

Elettronica In

www.elettronica.in.it

€ 7,50 - Bimestrale, anno XXXI - n. 282 - DICEMBRE 2024 / GENNAIO 2025

Android auto wireless con Raspberry Pi



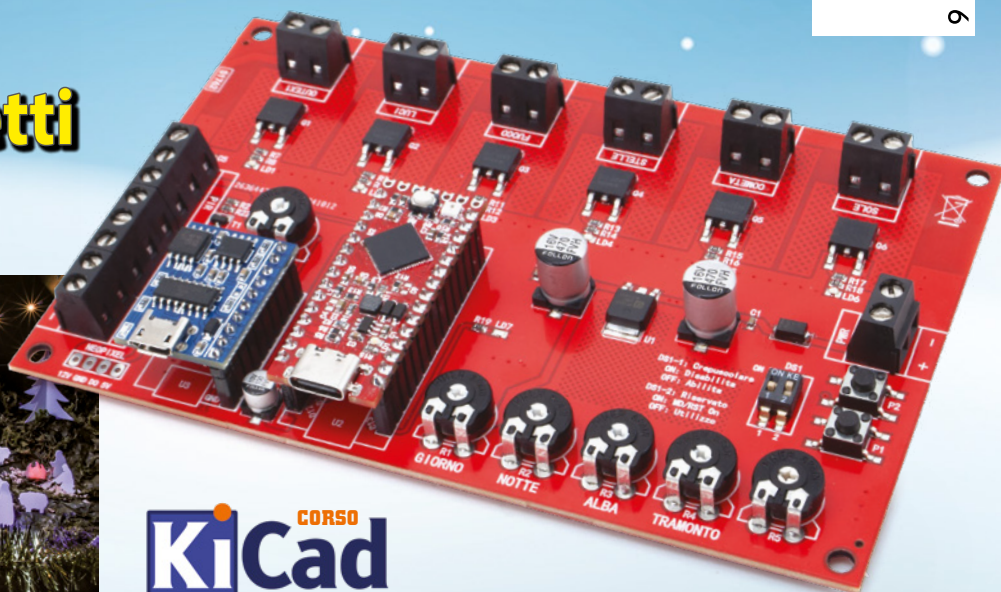
- ▶ Dissuasore per cani a ultrasuoni
- ▶ Regolare l'illuminazione con LoRa
- ▶ Circuito RC con Arduino

Scheda di controllo effetti per presepe



Smart Farm con ESP32

- ▶ Interruttore crepuscolare
- ▶ Rilevatore facciale con AI MediaPipe
- ▶ NANO R4: grande tecnologia in piccolo formato
- ▶ Modulo PIR a relé
- ▶ LC METER in-circuit



KiCad CORSO

Cerca, trova, proteggi! Localizzatori satellitari per ogni necessità!

Localizzatori GPS/GSM/GPRS per tutte le esigenze in grado di monitorare in tempo reale l'auto, la moto, la bicicletta, i bagagli e tanto altro.

LOCALIZZATORE 2G GPS/GSM/GPRS



**IL PIÙ
VENDUTO**

cod. FR864
€48,00

Dispone di base magnetica, ascolto ambientale e batteria da 1500mA. Servizio di localizzazione tramite App/ Piattaforma-Web/SMS e App dedicata alla configurazione. Grado di protezione IP65. Autonomia 3-5 giorni circa, in modalità tracking ad intervalli di un minuto.

LOCALIZZATORE GPS 4G LTE-2G - BATTERIA 3000 mA

**4G
LTE**



cod. FR851
€96,00



Prezzi IVA inclusa.

LOCALIZZATORE GPS 4G LTE-2G BATTERIA 1500 mA



Localizzatore GPS 4G di dimensioni compatte con grado di protezione IP65 che consente il monitoraggio in tempo reale di veicoli e persone tramite browser web o APP gratuita. Autonomia 2-4 giorni circa, in modalità tracking ad intervalli di un minuto, 5-7 giorni power saving mode, 10-15 giorni in modalità ibernazione.

**4G
LTE**

cod. FR865
€74,00

Ideale per l'uso in auto e personale (ad esempio per escursionisti), dotato di tracciamento in real-time, ascolto ambientale, tasto SOS, Geofence. Grado di protezione IP65. App dedicata per il tracciamento e la configurazione. Autonomia 5-7 giorni circa, in modalità tracking ad intervalli di un minuto, 15 giorni circa in modalità ibernazione. "Hibernation mode"

LOCALIZZATORE VEICOLARE GPS 4G LTE - 2G GSM/GPRS/EDGE



**4G
LTE**

cod. FR866
€79,00

LOCALIZZATORE GPS/GSM/Wi-Fi CON FUNZIONE SOS - 4G



Dispositivo di localizzazione satellitare GPS/GSM e Wi-Fi ad alta precisione. Facile e discreto nell'installazione sul veicolo, permette di localizzarlo con precisione in tempo reale. Viene fornito con una SIM a copertura mondiale attivabile gratuitamente, con 12 mesi di servizio rinnovabile alla scadenza.

cod. TRAKIMO4G
€169,00

Si installa in auto e sfrutta la connessione internet per monitorare la posizione del veicolo mediante App in cloud. Dotato di ascolto ambientale e batteria integrata, per funzionare anche in assenza della batteria del veicolo. È dotato di una batteria agli ioni di litio da 180mAh che offre un'autonomia necessaria da effettuare il ciclo di avvisi di allarme in assenza della tensione del veicolo.

**Informazioni dettagliate su
www.futurashop.it**



31 Anni di INnovazione

Con questo numero entriamo ufficialmente nel 31esimo anno di vita di Elettronica In, una rivista che ha partecipato alla storia dell'elettronica in Italia, raccontandone l'evoluzione. Fin dal primo numero, Elettronica In ha sempre puntato ad offrire contenuti innovativi e stimolanti per i lettori, accompagnandoli in un percorso di crescita e scoperta. Dalla sua nascita, la rivista ha seguito l'evoluzione del mondo della tecnologia, passando dai semplici circuiti e progetti hobbistici fino ad arrivare alle tecnologie di oggi, come l'Internet of Things e l'intelligenza artificiale. Dopo tutti questi anni, possiamo dire con orgoglio che

Per noi, Elettronica In è più di una semplice rivista: è il nostro contributo per creare il futuro.

ogni articolo, progetto e corso è stato realizzato con la volontà di trasmettere passione e competenza.

Dai primi tutorial sulle basi dell'elettronica fino alle sfide moderne dell'automazione e della robotica, siamo stati sempre al fianco dei nostri lettori: appassionati di ogni età, principianti ed esperti, con l'obiettivo

di rendere l'elettronica accessibile a tutti. Abbiamo sempre collaborato attivamente anche con le scuole, già prima che prendessero piede le competizioni di robotica, di cui siamo stati in passato sponsor e promotori. Oggi, la nostra collaborazione è focalizzata sull'apprendimento e l'innovazione, con la presentazione di prodotti STEM, come nel progetto di Smart Farm di questo numero, realizzato con ESP32.

Per noi, Elettronica In è più di una semplice rivista: è il nostro contributo alla costruzione di un futuro fatto di persone che non smettono mai di imparare, innovare e sognare.

E queste persone siete anche voi che, abbonandovi, diventate parte di questo viaggio, continuando a imparare, a esplorare nuove idee e a coltivare la passione che ci unisce. Buona lettura e buon divertimento con l'elettronica!

ON LINE

 elettronica.in

SI RINGRAZIA

Futura Elettronica
Mouser



In copertina:
per le festività natalizie
vi proponiamo
una scheda elettronica
che permette di
controllare le luci
e la musica
del vostro presepe.


Boris Landoni
 boris.landoni@elettronica.in

Elettronica In

Anno XXXI n. 282
DICEMBRE 2024 / GENNAIO 2025

EDITORIALE

Direttore Responsabile

Boris Landoni
 boris.landoni@elettronica.in

Redazione

Stefano Garavaglia, Paolo Gaspari,
Davide Scullino, Gabriele Daghetta
 redazione@elettronica.in

Ufficio Pubblicità

Monica Premoli (0331-752668)
 monica.premoli@elettronica.in

Ufficio Abbonamenti

 Tel. 0331-752668
 abbonati@elettronica.in

Direzione, Redazione, Pubblicità

FUTURA GROUP srl - Divisione Editoriale
via Adige 11 - 21013 Gallarate (VA)
 Tel. 0331-752668

Abbonamenti

Annuo 6 numeri Euro 39,00

Annuo 6 numeri digitale Euro 29,00

Le richieste di abbonamento vanno inviate a:

FUTURA GROUP srl
via Adige 11, 21013 Gallarate (VA)

 Tel. 0331-752668

oppure tramite il sito

<https://futura.net.it/abbonamenti>

Distribuzione per l'Italia

SODIP srl
via Bettola 18 - 20092 Cinisello Balsamo (MI)
 Tel. 02-660301  Fax 02-66030320

Stampa

ROTO3 Spa - Via Turbigo, 11/b
20022 CASTANO PRIMO (MI)



Rivista registrata presso il Tribunale di Milano
con il n. 245 il 03/05/1995.

Prezzo di copertina Euro 7,50.

Gli arretrati nei formati cartaceo e digitale (pdf) sono acquistabili
sul sito della rivista al prezzo di Euro 7,50.

Poste Italiane Spa - Spedizione in abbonamento Postale - D.L.
353/2003 (conv. in L. 27/02/2004) art. 1 comma 1 - DCB Milano.

Futura Group srl è iscritta al Registro Operatori della
Comunicazione n. 23650 del 02/07/2013.

Impaginazione ed immagini sono realizzati in DeskTop Publishing
con programmi Adobe InDesign e Adobe Photoshop per Windows.

Tutti i contenuti della Rivista sono protetti da Copyright.

Ne è vietata la riproduzione, anche parziale, la traduzione e più in
generale la diffusione con qualsiasi mezzo senza l'autorizzazione
scritta da parte dell'Editore. I circuiti, i firmware e i software
descritti sulla Rivista possono essere realizzati solo per uso
personale, ne è proibito lo sfruttamento a carattere commerciale
e industriale. Tutti possono collaborare con Elettronica In.

L'invio di articoli, materiale redazionale, programmi, traduzioni,
ecc. implica da parte del Collaboratore l'accettazione dei compensi
e delle condizioni stabilite dall'Editore (per maggiori informazioni
scrivere a redazione@elettronica.in). Manoscritti, disegni e
foto non richiesti non verranno in alcun caso restituiti. L'utilizzo
dei progetti e dei programmi pubblicati non comporta alcuna
responsabilità da parte della Società Editrice.

© 2024 Futura Group srl

**Futura
Group**
Edizioni



Contenuti

01

RUBRICHE

- 01 Editoriale
- 05 T&T Tips & Tricks
- 156 Scienza & Tecnologia

Progetto di Copertina



PRESEPE CON NANO R4

102

HOME PROJECT

Dissuasore per cani a ultrasuoni

**11**

DIDATTICA

Smart Farm con ESP32

**17**

WIRELESS

Regolare la luminosità con LoRa

**29****11**

ARTICOLI

HOME PROJECT

11 Dissuasore per cani a ultrasuoni

Irradia nell'ambiente onde ultrasoniche a frequenze tali da tenere lontane alcune specie animali indesiderate, garantendo la sicurezza sia degli animali che delle persone senza causare danni.

DIDATTICA

17 Smart Farm con ESP32

Realizziamo una fattoria in miniatura completamente automatizzata che utilizza il potente modulo ESP32 per connettività Wi-Fi e Bluetooth, ideale per progetti IoT.

WIRELESS

29 Regolare la luminosità con LoRa

Utilizziamo la tecnologia LoRa per regolare con precisione l'intensità delle lampade LED, offrendo un controllo preciso e privo di interferenze rispetto ai tradizionali sistemi a infrarossi.

DIDATTICA

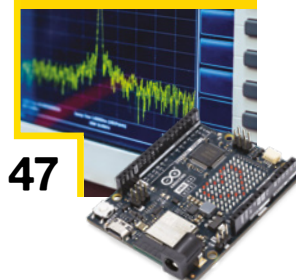
47 L'elettronica con Arduino: il circuito RC

Lo studio di un circuito RC, fondamentale per chi si avvicina al mondo dell'elettronica, è complicato senza un oscilloscopio. Arduino ci semplifica il lavoro grazie al plotter seriale.

HOME PROJECT

55 Interruttore crepuscolare

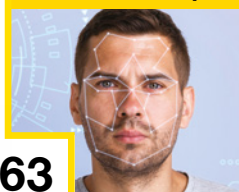
Comanda l'accensione di una o più lampade al calare della sera, spegnendole quando torna a fare giorno.

DIDATTICA**Circuito RC
con Arduino**

47

HOME PROJECT**Interruttore
crepuscolare**

55

**INTELLIGENZA
ARTIFICIALE****Rilevatore facciale
con AlMediaPipe**

63

IN AUTO**Android auto con
Raspberry Pi**

75

**INTELLIGENZA ARTIFICIALE**

63

**Rilevatore facciale
con AI MediaPipe**

Analizziamo un software capace di riconoscere le espressioni del viso e tra esse, lo stato di chiusura degli occhi per generare un allarme acustico.

IN AUTO

75

**Android auto wireless
con Raspberry Pi**

Come trasformare un Raspberry Pi in un dongle wireless per Android Auto, offrendo un'integrazione avanzata tra smartphone e veicolo, migliorando navigazione, musica e messaggistica.

TOP PROJECT

89

**NANO R4: grande tecnologia
in piccolo formato**

La board NANO R4 sviluppata da Futura Elettronica è una versione compatta della scheda Arduino UNO R4 minima. Mantiene lo stesso processore e le principali caratteristiche.

GADGET

102

Presepe con NANO R4

Centralina di gestione delle luci e delle musiche per un presepe sempre al passo con l'evoluzione tecnologica.

HOME PROJECT

113

Modulo PIR a relé

Perfettamente compatibile con i frutti per impianti elettrici BTicino Living Light e Vimar Arké, attiva un relé miniatura al suo interno quando rileva l'approssimarsi di persone nell'ambiente.

**I PROGETTI DI **

125

LC METER in-circuit

Misuriamo la capacità e l'induttanza direttamente sul circuito.

DIDATTICA

137

Conoscere e usare KiCad

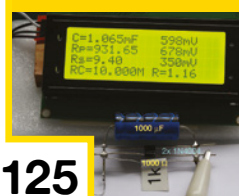
Impariamo ad utilizzare uno strumento di progettazione elettronica tanto potente quanto versatile, che consente di partire dall'idea ed arrivare alla scheda finita. Prima Puntata.

TOP PROJECT**NANO R4: tecno-
logia in miniatura**

89

HOME PROJECT**Modulo PIR
a relé**

113

******LC METER
in-circuit**

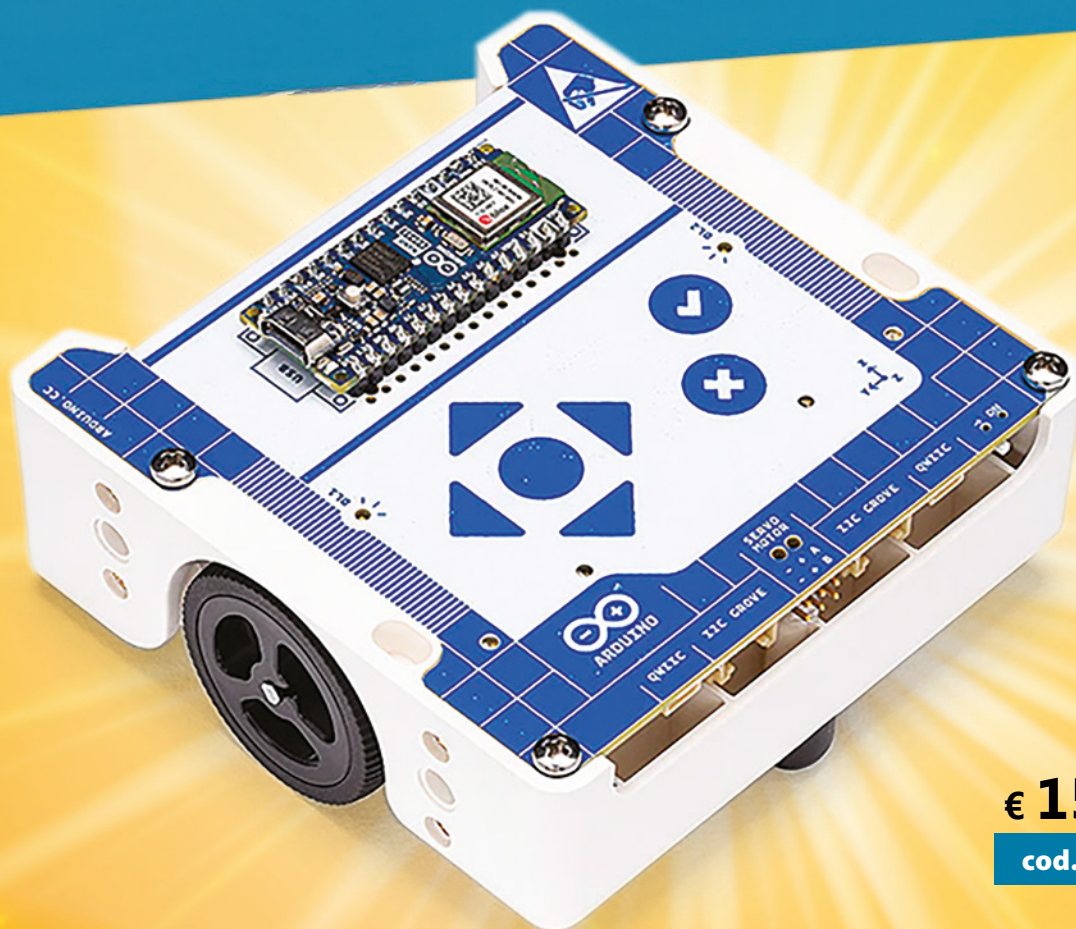
125

DIDATTICA**Conoscere
e usare KiCad**

137



ALVIK Robot



Prezzo IVA inclusa.

€ 159,00

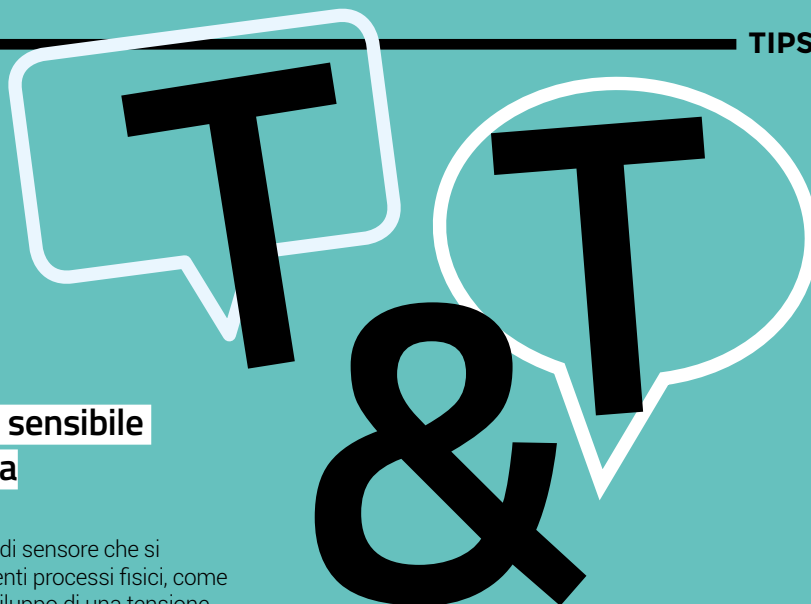
cod. ALVIK

Arduino **Alvik** è un robot educativo progettato per avvicinare studenti, insegnanti e appassionati di tecnologia al mondo della programmazione e della robotica. Grazie alla sua semplicità, **Alvik** è perfetto per chiunque, dai bambini agli adulti, indipendentemente dal livello di esperienza.

Questo robot supporta diversi linguaggi di programmazione, tra cui la programmazione a blocchi, MicroPython e il linguaggio Arduino, rendendo l'apprendimento della programmazione un'esperienza divertente e coinvolgente.

Alvik include un pacchetto di 11 lezioni pratiche che esplorano attività di robotica e progetti innovativi. Questi corsi aiutano a sviluppare il pensiero critico e le capacità di risoluzione dei problemi, rendendo la programmazione accessibile e stimolante per tutti. Scegliere Arduino **Alvik** significa optare per un'esperienza educativa versatile che promuove la creatività e l'innovazione, trasformando il processo di apprendimento in un'avventura appassionante.





Il resistore sensibile alla chimica

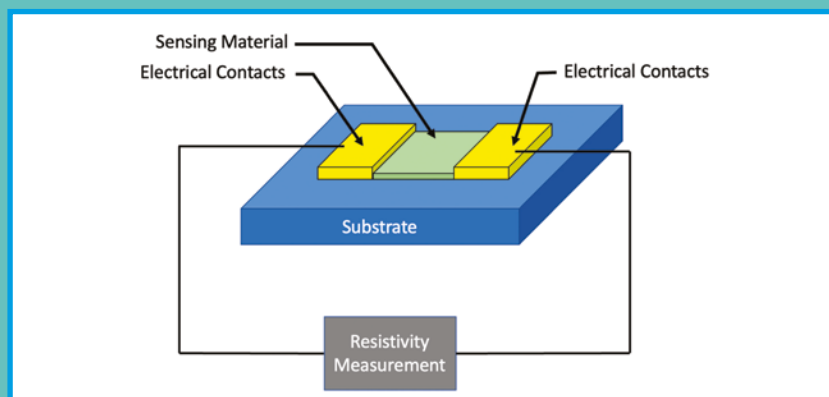
Esistono vari tipi di sensore che si basano su differenti processi fisici, come ad esempio lo sviluppo di una tensione o la variazione di resistenza elettrica e capacità. Tra quelli a variazione di resistenza possiamo citare i fotoresistori, la cui resistività varia in base all'esposizione luminosa e i termistori, sensibili alla temperatura. Esistono poi dispositivi più recenti come i chemiresistori, il cui comportamento è legato all'interazione chimica anziché fisica, come nei due precedenti.

I chemiresistori sono materiali conduttori con una resistenza/conduktanza intrinseca che può subire cambiamenti in risposta alle variazioni delle caratteristiche chimiche dell'ambiente circostante. Il termine chemiresistore è stato coniato nel 1985 dopo aver riscontrato che la resistività della ftalocianina di rame cambiava quando esposta a vapori di ammoniaca. I chemiresistori sono dei trasduttori che si basano sull'interazione chimica tra la sostanza da rilevare (analita) e il materiale di rilevamento che può essere basata su legami covalenti, legami a idrogeno o ancora sul riconoscimento molecolare. I più comuni materiali chemiresistivi sono i semiconduttori basati su ossido di metallo, i polimeri conduttivi, le nanoparticelle metalliche e i nanomateriali a base di carbonio come grafene e nanotubi di carbonio. Questa resistenza può essere misurata utilizzando sensori semplici ed economici. Un'architettura di base del chemiresistore è costituita da uno strato del materiale di rilevamento che collega due elettrodi o riveste un set di elettrodi interdigitati su substrato di allumina, vetro, SiO₂/Si o substrato flessibile. In genere, gli elettrodi sono realizzati in metalli conduttivi come platino e oro per fornire un buon contatto ohmico con il materiale di rilevamento che a sua volta controllerà il trasferimento di carica tra gli elettrodi. La maggiore quantità di materiale a contatto con le dita in un elettrodo interdigitato aumenta

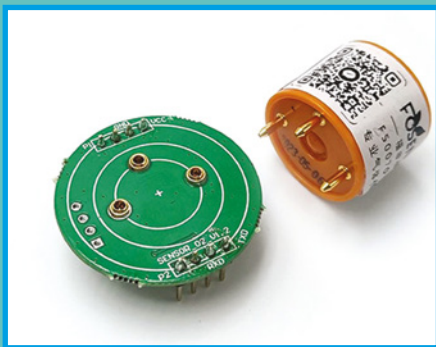
Tips & Tricks sono i nostri suggerimenti in pillole e i piccoli trucchi riguardanti sia l'hardware che il firmware, che sperimentatori, tecnici e progettisti elettronici vorrebbero conoscere per sviluppare meglio la propria passione: quella per l'Elettronica applicata. Ecco quelli che abbiamo preparato questo mese per voi.

la conduttività complessiva, rendendo possibile l'uso di materiali sensibili con elevata resistività di offset. Tuttavia, spesso richiedono tecnologie di microfabbricazione come serigrafia e fotolitografia a seconda della risoluzione richiesta. Per quanto riguarda le variazioni di resistenza, dipenderà dal carattere dell'interazione con la sostanza da rilevare, portando al rilevamento binario della presenza/assenza dell'analita (rilevamento qualitativo) o all'indicazione precisa della concentrazione dell'analita (rilevamento quantitativo). In entrambi i casi, i sensori chemiresistivi sono comunemente impiegati in sistemi per operazioni singole o in array di sensori come "lingue elettroniche" o "nasi elettronici".

Negli ultimi anni, la tecnologia dei chemiresistori è stata applicata nello sviluppo di sensori per numerose applicazioni grazie alla loro capacità di fornire informazioni accurate in tempo reale tramite dispositivi miniaturizzati con un consumo minimo di elettricità. Un esempio sono i sensori di gas, che rilevano la concentrazione di gas misurando la variazione di resistenza tra gas ed elettrodi. Considerando che le prestazioni di rilevamento dei materiali chemiresistivi dipendono anche dalle loro proprietà microstrutturali, i materiali nanostrutturati, più specificamente le nanofibre (NF), sono molto interessanti considerando il loro elevato rapporto superficie/volume, l'elevata porosità e le caratteristiche facilmente regolabili come



↑ Schematizzazione del chemiresistore.



↑ Sensore di gas a chemiresistore.

composizione, struttura, morfologia e diametro. Le NF polimeriche sono le più tradizionali e, quando composte da polimeri conduttori (CP), sono comunemente utilizzate nei chemiresistori per le loro eccellenti proprietà elettroniche e fisico-chimiche sintonizzabili. In grado di rispondere a un'ampia gamma di analiti polari e non polari, il meccanismo di rilevamento dipenderà dall'interazione tra l'analita e il CP. L'interazione analita-CP è stata discussa come sei diversi tipi di interazione che, alla fine, causeranno tutti cambiamenti nel livello di drogaggio, conduttività e trasporto di elettroni nello scheletro polimerico coniugato. La varietà di possibilità di interazione è un vantaggio principalmente nelle applicazioni in cui è necessario rilevare più di un analita. Un ottimo esempio è stato presentato da Liu et al. (2012a, b), in cui le NF di polianilina drogata con acido cloridrico puro (d-PANI) sono state applicate come materiale per chemiresistori per il rilevamento di CO in atmosfera di idrogeno in concentrazioni basse fino a 1 ppm. La risposta rapida e selettiva è stata possibile grazie alle diverse interazioni di legame di H₂ e CO con d-PANI, dove H₂ si lega con atomi di azoto protonati (gruppi amminici) e CO con atomi di azoto non protonati, riducendo la resistenza elettrica aumentando i portatori di carica nello scheletro d-PANI. Indipendentemente dal meccanismo, i chemiresistori basati su nanofibre polimeriche conduttrici (CPNF) sono generalmente rapidi e reversibili rispetto ad altri materiali. Inoltre, la selettività può essere facilmente modulata dalla funzionalizzazione post-sintesi. Complessi metallici, nanofibre inorganiche o particelle e nanosheets, e derivati del carbonio sono i candidati più altamente valutati per la funzionalizzazione post-sintesi di CPNF. Ad esempio è stata segnalata la modulazione della resistività e della sensibilità di grafene/nanofibre polimeriche

aggiungendo tetrafenilporfirina. Anche la temperatura (60–70 °C) ha giocato un ruolo significativo nella sensibilità, contribuendo a migliori prestazioni delle fibre con porfirine, mentre le fibre senza porfirina avevano una sensibilità ai COV influenzata dalla temperatura.

Numerazione ottale, binaria ed esadecimale

Una grandezza elettrica analogica può assumere infiniti valori entro due margini, quindi per ottenere una tensione variabile ci vorrebbe un'infinità di transistor che, commutando uno alla volta, uno in successione all'altro, fanno crescere l'ampiezza e viceversa. L'alternativa è rappresentare il valore in formato numerico, ma anche qui, immaginando di dover andare da 0 a 100, significherebbe utilizzare 100 transistor. Se i computer dovessero trattare i valori numerici usando la numerazione decimale, dovrebbero essere grandi dieci volte quello che sono; ecco perché si è utilizzata la logica binaria, che consiste nel rappresentare tutti i numeri in base due, il che permette di utilizzare due condizioni: 1 e 0. Uno significa attivo e 0 non attivo; nel caso della numerazione, uno vuol dire che la posizione corrispondente va contata e 0 che, la stessa, vale 0. Perché è stata utilizzata la logica binaria è presto detto: due condizioni elettriche sono facilmente producibili e, cosa più importante, riconoscibili con ampia tolleranza da circuiti estremamente semplici; ad esempio, usando dei comuni transistor, l'elemento in conduzione (saturazione) è attivo e interdetto (tensione sulla sua base nulla o inferiore a quella di soglia) è non attivo. Quindi, all'ingresso di uno stadio a BJT la presenza di tensione sopra gli 0,6 V si può interpretare come livello uno e l'assenza di differenza di potenziale (o meno di 0,6 V) come stato zero. Per questo motivo, nella logica binaria si associa 1 al livello (o stato) alto di tensione e lo 0 al livello (o stato) basso. Ma come si conta con due soli livelli logici? Ebbene, si fa come con il sistema decimale: in quest'ultimo, andando da destra a sinistra, le cifre assumono un valore che è pari a multipli, ovvero potenze di dieci: la prima cifra vale 10 elevato alla 0 (1), la seconda 10 alla 1 (10), la terza dieci alla seconda (100), la quarta 10 alla terza (1.000) e via di seguito. Quindi, 230 significa la somma di 0, 3 per 10 elevato alla 1 e 2 per dieci elevato alla 2. Nella numerazione binaria le cose sono analoghe, solo che le posizioni dei numeri

corrispondono a potenze di due e, inoltre, i numeri sono solo 1 e 0 (due, appunto) invece di dieci come nella numerazione decimale; quindi, la prima posizione a destra vale 2 alla 0 (1), la seconda 2 alla 1 (2), la terza 2 alla seconda (4), la quarta 2 alla terza (8), la quinta 2 alla quarta (16) ecc. Per ricavare il valore decimale di un numero, bisogna ricordare che quando c'è zero non bisogna sommare il valore della posizione in cui questo si trova, mentre occorre considerare i valori delle posizioni aventi 1. Per fare un esempio, 110 vale la somma di 0, 2 e 4, ossia corrisponde al 6 decimale; 1001 vale 1+0+0+8, quindi 9. Oltre ai sistemi in base 10 e 2, esistono altri metodi di rappresentazione, usati per effettuare operazioni matematiche o rappresentare in forma ridotta grandi quantità di dati binari; resta inteso che l'elettronica digitale funziona solo con 1 e 0, quindi in binario e che quindi tutti i dispositivi digitali (porte logiche, encoder e decoder, contatori, CPU, memorie ecc) lavorano con due soli valori. La numerazione ottale ha, come fa intendere il suo nome, base otto; quindi prevede otto valori e le posizioni da destra a sinistra valgono come le potenze di otto: 8 alla zero (1), 8 alla uno (8), 8 alla seconda (64) ecc. Per ottenere il corrispondente decimale basta moltiplicare ogni numero per il valore rappresentato dalla cifra in cui si trova e sommare i risultati dei singoli prodotti; quindi, ad esempio, 10 ottale corrisponde a 0+8 = 8. 11 ottale vale 1+8 = 9, mentre 20 vale 16+0=16 e 22 corrisponde a 16+2=18. Quanto alla numerazione esadecimale, è molto importante perché la vedrete spesso in alcune rappresentazioni

Decimale	Binario	Esadecimale
0	0	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

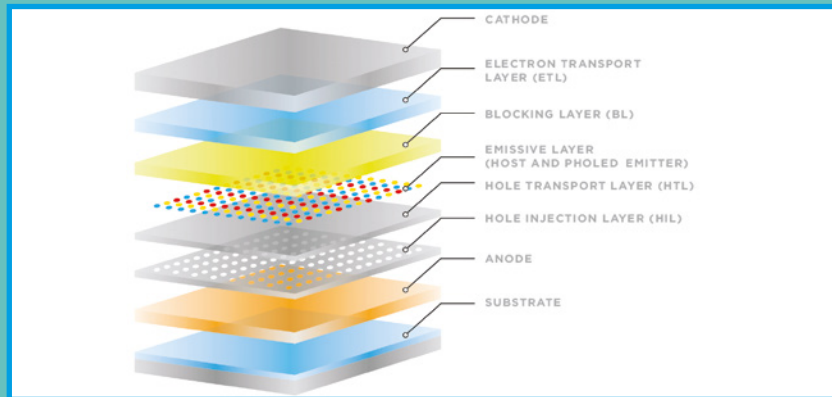
↑ Corrispondenza tra numerazione decimale, binaria ed esadecimale per numeri da 0 a 15.



delle locazioni di memoria o di valori in memoria; è comoda perché in alcuni casi si riducono interi byte in valori esadecimali per ottimizzare lo spazio occupato in memoria. Il formato esadecimale ha base 16, quindi ogni posizione da destra a sinistra assume il valore di una potenza di 16: si parte dalla destra, occupata da 16 alla 0 (1), poi si ha 16 alla 1 (16), 16 al quadrato (256) e via di seguito; anche in questo caso, il corrispondente valore decimale si ottiene moltiplicando il numero di ogni cifra per il valore corrispondente e sommando i risultati. La cosa interessante della logica esadecimale è che ha ben 16 valori; ora, non esistendo altrettanti numeri decimali per rappresentare una singola cifra, si va da 0 a 9 e, oltre si usano le lettere A, B, C, D, E, F, che valgono, rispettivamente, 10, 11, 12, 13, 14, 15. Ad esempio, 10 esadecimale corrisponde a $1+16=17$, mentre 20 esadecimale vale $0+32$; 21 esadecimale vale $1+32$, quindi 33. Questo quando si usano solo numeri; ma esistono anche rappresentazioni come 0F, che vale $15+0=15$, oppure AB, che vale $11+(10 \times 16)$ e quindi 171. Notate che in molti casi, quando un numero viene espresso in formato esadecimale ma non si può dedurlo dalla sua espressione (perché non ha lettere ma solo numeri), tra parentesi si mette accanto la scritta Hex. Ad esempio, un valore come 10 potrebbe benissimo essere decimale (varrebbe 10) o binario (varrebbe 2); se si intende come esadecimale, basta scrivere 10 Hex o 10 hex. È chiaro che se la rappresentazione del numero ha delle lettere, risulta evidente come sia esadecimale e sia quindi inutile utilizzare il suffisso Hex.

FR4, ovvero la vetronite

Un po' tutti siamo abituati ad associare il circuito stampato alla vetronite, che è il materiale con cui più frequentemente si realizzano i PCB, da quelli amatoriali ai più professionali sfornati dall'industria elettronica. Eppure quando si parla di vetronite non si intende un'unica cosa, perché sebbene il supporto su cui si applicano le ramature dei circuiti stampati è ciò che tecnicamente viene chiamato FR4 (dove FR sta per Flame Resistant e 4 per il tipo di materiale, che si deve alla NEMA...) ed è sostanzialmente una base di fibra di vetro impregnata di resina epossidica contenente un composto ritardante la fiamma, l'industria sforna vari tipi di tale materiale, che si distinguono per temperatura massima di lavoro



↑ Struttura del pixel di un display PHOLED (OLED fosforescente).

(prima della deformazione e della bruciatura) ma anche per rigidità dielettrica e robustezza meccanica. Esiste una classificazione internazionale basata attualmente sulla norma IPC4101, cui sono assoggettati i tipi di vetronite in commercio. La norma è poi stata evoluta nelle sue release successive.

Ad oggi sono classificate due classi di base della vetronite, ossia la FR4.0 e la FR4.1: nella prima il materiale ritardante la fiamma è bromina, mentre nella seconda, creata per ottemperare alle direttive RoHS, vengono utilizzati materiali non alogeni.

In tutti i casi, l'impregnante è costituito da una miscela di resine epossidiche e impregnanti secondari inorganici (il cui peso deve essere inferiore al 50% delle epossidiche) composti da talco, biossido di titanio, silice, carbonato di calcio, argilla ecc. in base alle caratteristiche desiderate di costante dielettrica, capacità tra due strati ramati, resistenza all'incendio, robustezza meccanica, umidità trattenuta ecc.

OLED fosforescenti

Negli smartphone e nelle TV i display OLED stanno soppiantando gli LCD, però in termini di durata e luminosità non sono l'ideale. Una loro evoluzione, chiamata PHOLED (Phosphorescent OLED) blu colmerà tali lacune.

Per comprendere perché un PHOLED sia così importante bisogna sapere che gli OLED convertono direttamente l'energia elettrica in fotoni. Hanno più strati, tra cui un catodo, un anodo e uno strato emissivo.

Quando la corrente attraversa un OLED, gli elettroni lasciano il catodo e le lacune lasciano l'anodo; si ricombinano nello

strato emissivo centrale, unendosi in eccitoni e rilasciando luce o calore mentre decadono.

A seconda di come si allineano gli stati di spin delle molecole, questo processo può generare due tipi di eccitoni: singoletti e tripletti. I singoletti sono in genere un terzo dei tripletti, ma ciò è un problema, perché negli OLED tradizionali, cioè fluorescenti, solo i singoletti emettono fotoni, mentre i tripletti rilasciano la loro energia sotto forma di calore.

Ciò riduce l'efficienza, perché gli OLED trasformano solo il 25% dell'energia elettrica in luce. Aggiungendo metalli pesanti al mix di composti organici che compongono gli OLED, gli stati di spin cambiano e persino gli eccitoni tripletti emettono luce.

Il risultato è un OLED fosforescente, potenzialmente efficiente al 100%. Gli OLED rossi fosforescenti esistono da circa due decenni; il verde è disponibile da un decennio ed il blu è in sviluppo, perché il suo gap di energia più elevato richiede l'utilizzo di materiali speciali. Ad oggi i display PHOLED si basano su subpixel rossi e verde, mentre il blu è OLED.

Web Forum e supporto tecnico

Hai un problema con uno dei circuiti pubblicati? Vorresti effettuare una modifica? Accedi al nostro Web Forum dove i nostri tecnici (ma anche gli altri lettori) ti aiuteranno a chiarire qualsiasi dubbio di natura tecnica. Collegati a:

www.elettronicaonline.it/webforum



ABBONAMENTO
ANNUALE



Abbonati a Elettronica In!
Realizza i **Progetti** descritti sulla rivista,
rimani aggiornato con i nostri **Corsi**,
approfondisci la conoscenza delle
tecniche e dei componenti
più avanzati con i **Tutorial**
e le nostre **News**.



Visita: www.elettronica.in.it

6 NUMERI
A SOLI

€29

ANZICHÉ € 45,00

ABBONAMENTO ANNUALE PER RIVISTA DIGITALE

In **OMAGGIO** avrai
il **10% DI SCONTO** per i tuoi ordini
su **Futurashop.it**

COME ABBONARSI

→ **on-line**

nella pagina "**Abbonati o rinnova**"
del nostro sito www.elettronicaain.it

→ **e-mail**

scrivendo a
abbonati@elettronicaain.it

→ **telefono**

telefonando a
+39-0331-752668

NUOVO FORMATO 160 PAGINE CON PIÙ CONTENUTI

ABBONAMENTO ANNUALE PER RIVISTA **CARTACEA**

In **OMAGGIO** avrai
il **10% DI SCONTO** per i tuoi ordini
su **Futurashop.it**

GRATIS 1 VOLUME A SCELTA della collana
"L'elettronica per tutti" oppure

BUONO SCONTO DA 5€ valido 12 mesi
da utilizzare su **Futurashop.it** nelle categorie "Libri" o "Riviste"

La possibilità di **LEGGERE GRATUITAMENTE**
la versione digitale di ELETTRONICA IN

6
NUMERI
A SOLI

€39

ANZICHÉ € 45,00

IL SALOTTINO DEL LIBRO TECNICO

Ultimi libri presentati



€ 13,90

cod. S5745



€ 16,99

cod. GRAMANPE



€ 17,90

cod. S5741

Prezzi IVA inclusa.

Scoprili tutti su futurashop.it



€ 4,99

cod. S5526



€ 15,90

cod. S5691



€ 12,90

cod. S5722

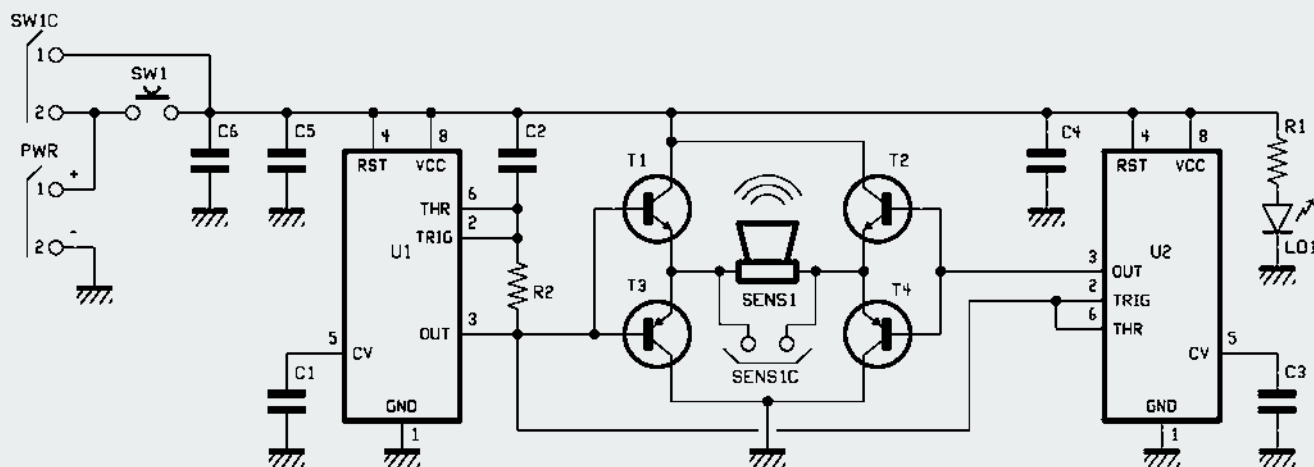
DISSUASORE PER CANI A ULTRASUONI



a cura della **REDAZIONE**

Irradia nell'ambiente onde ultrasoniche a frequenze tali da tenere lontane alcune specie animali indesiderate, garantendo la sicurezza sia degli animali che delle persone senza causare danni.

Può capitare di trovarsi in situazioni molto particolari nelle quali occorra tenere lontani gli animali da aree o da apparati per l'incolumità loro e delle persone o di altri animali, ovvero anche solo per essere al riparo da continui latrati; in casi del genere, fermo restando che il contrasto a fenomeni come il randagismo spetta alle autorità competenti, è possibile fare ricorso all'elettronica per implementare contromisure che senza causare danno ai quadrupedi raggiungono comunque lo scopo. Tra queste, una molto semplice consiste nel produrre e irradiare nell'ambiente circostante, o nella direzione di possibile provenienza, degli ultrasuoni, che animali come cani e gatti percepiscono provando un leggero fastidio, tale da indurli ad



allontanarsi; tale sensibilità riguarda anche animali di taglia più piccola, fino a giungere agli insetti come le zanzare.

Il progetto che vi proponiamo in queste pagine configura proprio tale soluzione: si tratta di un generatore di segnale a ultrasuoni, poco sotto i 40 kHz, oltre che di un trasduttore specifico pilotato tramite uno stadio a ponte.

Il tutto verrà meglio descritto durante l'analisi circuitale del progetto.

SCHEMA ELETTRICO

Il nostro dispositivo è composto idealmente da tre blocchi, che sono un generatore di impulsi rettangolari, facente capo ad un timer 555, un invertitore di tale segnale implementato da un secondo integrato 555 e da uno stadio di potenza a ponte che serve a fornire alla capsula piezoelettrica la tensione e la corrente occorrenti a irradiare gli ultrasuoni a una distanza di diversi metri. Per farvi comprendere come funziona il circuito dobbiamo per prima cosa spiegare come funziona il 555, perché due stadi determinanti funzionano basati su di esso, sebbene in una configurazione diversa: il primo stadio genera l'onda rettangolare a frequenza ultrasonica come blocco principale, utilizzando il timer 555 (siglato U1 nello schema elettrico) in modalità di multivibratore astabile; nel secondo stadio, invece, sfruttiamo solo il flip-flop e i comparatori senza alcuna temporizzazione. La funzione di questo secondo 555 è ottenere impulsi di tensione in opposizione di fase rispetto a quelli prodotti dall'uscita dell'U1.

Il timer 555 contiene due comparatori aventi in comune un partitore di tensione a scala di resistenze che polarizza, all'operazionale superiore l'ingresso invertente e a quello inferiore, il non-invertente; dei comparatori vengono resi accessibili dall'ester-

no gli ingressi non-invertenti di quello superiore, corrispondente al piedino 6 (THR, ossia Threshold) e l'invertente di quello superiore, corrispondente al piedino 2 (TRI, ossia Trigger).

Notate che per "superiore" intendiamo il comparatore che si trova in alto nello schema a blocchi interno proposto nella **Fig. 1** (quindi riceve il potenziale di riferimento più alto) mentre definiamo "inferiore" quello in basso, ossia quello che riceve dal partitore multiplo la tensione di riferimento più bassa.

Il nodo tra la prima e la seconda resistenza del partitore a scala di riferimento dei due comparatori (alimentato, come il resto degli stadi interni al 555, dal piedino 8) viene portato all'esterno tramite il piedino 5 (CONT, che sta per Control Voltage) e ciò permette di alterare le tensioni di riferimento dei comparatori, cosa che in questo progetto non è richiesta, tant'è che ci limitiamo a filtrare il piedino 5 di ciascuno dei 555 così da rendere il più stabile possibile ciascuno dei potenziali di riferimento. Le uscite dei comparatori entrano una nel reset (quella del comparatore superiore, che arriva su R) e nel SET (comparatore inferiore) di un flip-flop di tipo RS, che è un circuito logico la cui uscita diretta (Q) va a livello alto (corrispondente a circa il potenziale del piedino 8) quando il SET è a livello alto, ovvero assume lo zero logico se SET è posto a circa zero volt. L'uscita presenta un comportamento esattamente opposto quando ad essere stimolato è l'ingresso reset, il quale, posto a livello logico alto, resetta il flip-flop, intendendo con ciò che ne azzerà lo stato di Q.

Il flip-flop ha anche un'uscita complementata (/Q), la cui condizione logica è sempre l'inverso di quella Q; nel 555 è collegata, tramite una resistenza, alla base di un transistor NPN configurato ad open collector il cui emettitore è connesso a massa e il

collettore al piedino DIS (Discharge, 7).

Il transistor può commutare una corrente di collettore di 200 mA, utilizzato in modo sink (ad assorbimento di corrente). L'uscita diretta, Q, è invece collegata al piedino OUT (3) dell'integrato attraverso un buffer interno push-pull, capace di erogare un massimo di 200 mA.

Il negativo di alimentazione, vale a dire la massa di riferimento del 555 corrisponde al piedino 1, mentre al 4 è connesso il reset generale (R1) a logica invertita del flip-flop: questo piedino permette di resettare forzatamente il circuito dall'esterno attraverso l'applicazione di un livello logico basso, ma nel nostro caso la relativa funzionalità è stata disattivata collegando il piedino 4 al positivo di alimentazione (piedino 8).

La massima frequenza di funzionamento del 555 configurato come stabile è 500 kHz, ma nel nostro caso ci fermiamo a poco meno di 40 kHz.

Ciò detto, vediamo che il timer U1 è configurato come astabile, seppure in un modo atipico che ci serve per avere l'uscita in avvio a un certo livello di tensione: c'è un condensatore tra i piedini 4, 8 e 2, 6, oltre a un resistore tra questi pin e l'uscita (piedino 3). Tale configurazione permette di asservire la temporizzazione direttamente allo stato d'uscita del 555. Il condensatore di temporizzazione collegato tra i piedini 2,6 e il positivo di alimentazione, è l'elettrolitico C2.

Per comprendere come funziona l'astabile ci appoggiamo allo schema a blocchi di **Fig. 1** ed immaginiamo di alimentare l'integrato con C2 scarico e avente quindi una tensione tra le armature di 0 volt: inizialmente i piedini 2 e 6 sono al potenziale di alimentazione, pertanto l'uscita del comparatore

inferiore è a livello basso; ciò determina il livello zero sull'ingresso SET del flip-flop e lo stesso sull'uscita Q (la /Q si porta ad uno logico mandando in saturazione il transistor collegato al DIS, che in questa configurazione non è usato in quanto a gestire carica e scarica del condensatore C2 provvede direttamente l'uscita del 555).

Il livello basso sul piedino 3 determina la carica del condensatore C2 attraverso la corrente che fluisce nella resistenza R2, allorché a un certo punto i piedini 2 e 6 si trovano a un potenziale che prima scende sotto quello di riferimento del comparatore superiore (pari a $2/3$ della tensione di alimentazione del 555) e dopo si porta anche a livello di quello dell'inferiore (pari ad $1/3$ dell'alimentazione del 555). Quando il potenziale sui piedini 2, 6 dell'U1 scende sotto la soglia di commutazione del comparatore superiore, l'uscita di quest'ultimo si porta a livello basso liberando l'R del flip-flop, ma nulla cambia fin quando esso non scende al disotto della tensione di soglia del comparatore inferiore, che è pari a $1/3$ di quella applicata al piedino 8.

Ora l'uscita del suddetto comparatore si porta ad uno logico e porta con sé il SET del flip-flop, la cui uscita passa a livello alto determinando la commutazione da zero a livello alto dello stato di uscita del 555.

In questa condizione il condensatore C2 inizia a scaricarsi attraverso il circuito di uscita dell'U1, determinando la crescita del potenziale sui piedini 2 e 6 fino a superare in un primo momento la soglia di commutazione del comparatore inferiore e poi quella del comparatore superiore; la scarica avviene fino a $2/3$ della tensione applicata tra i piedini 1 e 8 del 555, quindi il ciclo ricomincia.

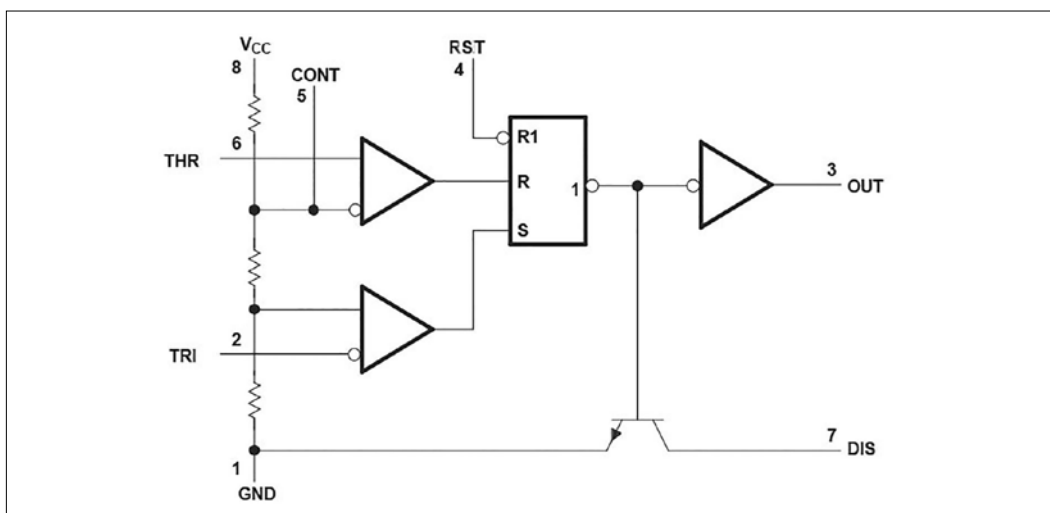


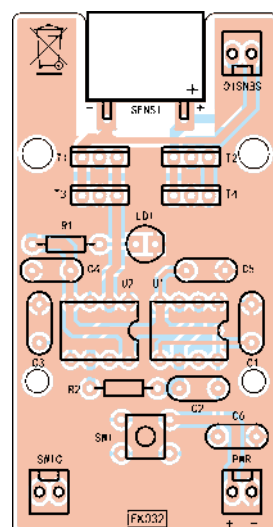
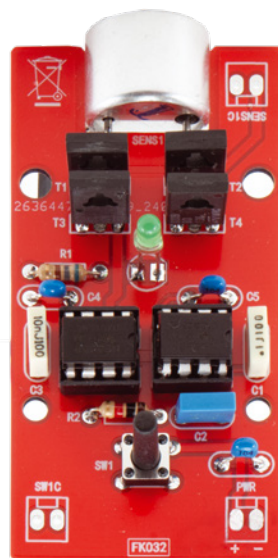
Fig. 1
Schema a
blocchi del
timer NE555.

Elenco Componenti:

R1: 680 ohm
 R2: 1 kohm
 C1: 100 nF 63 VL poliestere
 C2: 18 nF 63 VL poliestere
 C3: 10 nF 63 VL poliestere
 C4, C5, C6: 100 nF ceramico
 T1, T2: BD135
 T3, T4: BD136
 SW1: Microswitch 17 mm
 LD1: LED verde 3 mm
 SENS1: Capsula TX ultrasuoni 40 kHz
 U1, U2: NE555
 PWR: Clip per batteria 9V

Varie:

- Zoccolo 4+4 (2 pz.)
- Circuito stampato FK032 (35x69 mm)



Per effetto di ciò abbiamo l'uscita OUT dell'integrato che si mantiene a livello basso (circa zero volt) nel periodo che serve al condensatore affinché la tensione ai loro capi passi da 1/3 a 2/3 della tensione di uscita del 555, che poi coincide più o meno con quella di alimentazione e a livello alto nel tempo che trascorre da quando C2, in fase di scarica, registra un calo della tensione ai propri capi da 2/3 della tensione di alimentazione ad 1/3. I tempi delle commutazioni dipendono dal valore di C2 ed R2

$$t_h = 0,693 \times R2 \times C2$$

Tale durata coincide con la carica del condensatore. Invece la pausa dura un tempo pari a:

$$t_l = 0,693 \times R2 \times C2$$

e coincide con la scarica dell'elettrolitico.

La durata del periodo T è pari a $t_h + t_l$ e il duty-cycle vale:

$$dc = t_h / T$$

Essendo la carica e la scarica del condensatore C2 operate attraverso la stessa resistenza, si può dire che il duty-cycle vale il 50%, sebbene in pratica la carica sia un po' più veloce della scarica (impulso più lungo della pausa).

Queste sono le caratteristiche del segnale rettangolare fornito dall'U1, che piloterà un lato del ponte H a transistor NPN e PNP (circuitazione a simme-

tria complementare) cui è demandato il compito di amplificare il segnale ed in particolare la tensione che pilota la capsula a ultrasuoni; lo stesso segnale che arriva al lato sinistro del ponte (H-bridge) triggerà il successivo NE555, così da ottenere all'uscita di quest'ultimo un segnale in isofrequenza e opposto di fase, vale a dire un livello logico basso quando l'uscita dell'U1 è allo stato alto e viceversa. Tale segnale piloterà il lato destro del ponte ad H, così da fare in modo che quando T1 sarà in conduzione lo sarà anche T4, mentre T2 e T3 rimarranno interdetti e viceversa all'inversione dello stato logico alle uscite dei due 555.

Il segnale di uscita dell'U1 entra nei piedini 2 e 6 dell'U2, in modo da pilotarne gli ingressi liberi dei due comparatori: in presenza dello stato logico alto, vale a dire circa il potenziale di alimentazione, il comparatore superiore ha l'uscita a livello alto mentre quello inferiore (il cui ingresso fa capo al piedino 2) pone la propria uscita a zero logico; viceversa, quando U1 fornisce lo zero logico, il comparatore superiore dell'U2 ha l'uscita a zero e quello inferiore fornisce lo stato alto.

Nel primo caso il piedino 3 dell'U2 è a livello basso, mentre nel secondo è a zero; vedete quindi che il secondo 555 funziona sostanzialmente da inverter logico.

Concludiamo con l'alimentazione del circuito, che si alimenta tra i contatti PWR (Vcc è il positivo e GND il negativo) ai quali va applicata una tensione continua di valore compreso fra 5 e 9 volt, prelevata da una fonte in grado di erogare almeno 80 mA.

Il condensatore elettrolitico C6 serve a filtrare sia

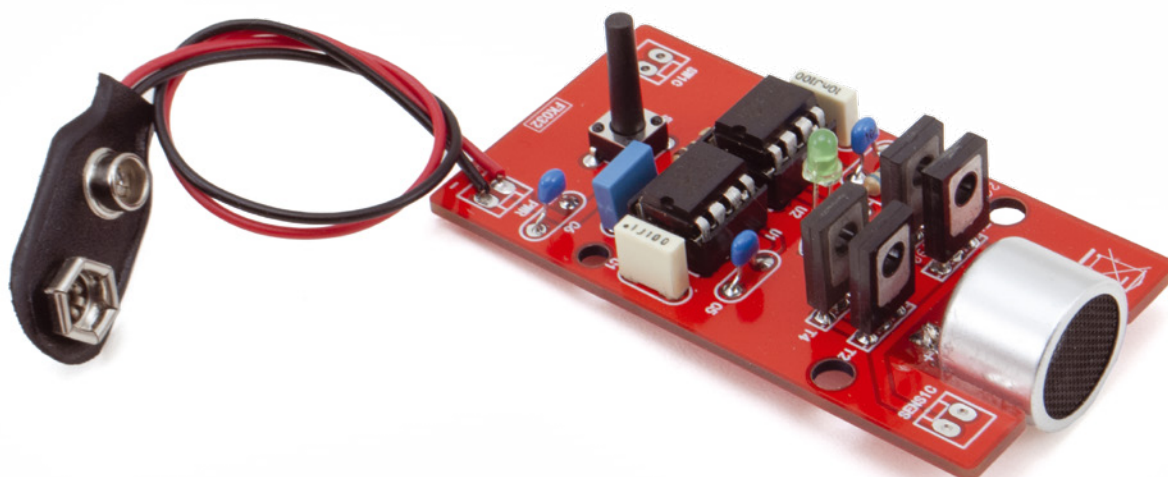


Fig. 2
Il segnale
ultrasonico
viene irradiato
da una capsu-
la piezoelet-
trica.

i disturbi introdotti sulla linea di alimentazione sia l'eventuale ripple generato da un alimentatore da rete, mentre C4 e C5 sono montati vicino alle linee di alimentazione dei due 555 per evitare che la pulsazione dei transistor del ponte e il conseguente carico sull'alimentazione propagino disturbi tali da alterare il segnale di uscita dei timer, impedendo il corretto funzionamento dell'intero circuito.

REALIZZAZIONE PRATICA

Passiamo adesso alle note costruttive: per il progetto abbiamo previsto il montaggio su circuito stampato a doppia ramatura, che abbiamo disegnato allo scopo e del quale rendiamo disponibile sul nostro sito www.elettronica.in.it il file gerber da utilizzare con il servizio PCBPRODUCTION disponibile dal sito ftrlink.it/pcb.

Una volta ottenuto il circuito stampato, potete procedere al montaggio dei componenti, tutti a foro passante (THT = Through Hole Technology), quindi facilmente gestibili anche da chi ha poca esperienza.

Iniziate il montaggio con le resistenze, gli zoccoli dip 4+4 contatti per i due 555, i condensatori (tutti non polarizzati) il pulsante e i quattro transistor, prestando attenzione a orientarli come indicato, ovvero tutti con il lato metallico rivolto alla capsula a ultrasuoni.

Passate quindi al LED (per il corretto orientamento ricordate che il catodo è l'elettrodo dal lato smussato del contenitore) e alla capsula piezoelettrica ad ultrasuoni, che andrà montata nella cava del circuito stampato, saldandone i terminali dal lato

componenti come si vede nelle foto del prototipo. Per l'orientamento dei componenti polarizzati fate riferimento al piano di montaggio visibile in queste pagine.

Per l'alimentazione potete utilizzare una presa volante per pile da 9 volt, di quelle a strappo, i cui fili rosso e nero dovreste infilare e stagnare nelle piazzole rispettivamente + e - PWR.

Una volta completato il montaggio e verificatane l'esattezza, il circuito è pronto per l'uso, dato che non richiede alcuna operazione di taratura preliminare; è vero che dovete regolare il trimmer per variare a piacimento la velocità di scorrimento del punto luminoso, ma il gadget funziona qualunque sia la posizione. Per far funzionare il circuito fornitegli un'alimentazione continua a tensione di valore compreso fra 5 e 9 volt. □



COSA OCCORRE ?

Codice	Descrizione	Prezzo € IVA inclusa
FK032	Scacciacani ultrasonico	8,00
555	IC Timer 555	0,60
SC-701	Contenitore plastico con vano batteria 9V	2,50

IL MATERIALE VA RICHIESTO A:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - www.futurashop.it

Smart Eco-Friendly House Kit

Kit Educativo STEM con guida illustrata ed esempi di programmazione

EDUKIT 16



Prezzo IVA inclusa.

€ 89,00

cod. EDUKIT16

Realizza una Smart House in miniatura completamente automatizzata, mediante l'utilizzo simultaneo di sensori e attuatori, che possono agire in piena autonomia o essere gestiti mediante una tastiera ed un display LCD; il cuore del progetto è una board compatibile con Arduino UNO, con eccellenti caratteristiche, denominata KS0486 Keystudio Plus.

CON
GUIDA
ILLUSTRATA
DI 56 PAGINE
IN ITALIANO



SMART FARM CON ESP32

del **PROF. MICHELE MENNITI**



Realizziamo una fattoria in miniatura completamente automatizzata che utilizza il potente modulo ESP32 per connettività Wi-Fi e Bluetooth, ideale per progetti IoT.

Sul numero 280 abbiamo presentato un kit progettato dalla Keyestudio, lo "Smart Eco-Friendly House Kit" e basato su una board Arduino compatibile. In questo articolo vedremo un nuovo kit dello stesso produttore, denominato "ESP32 Smart Farm", questa volta si tratta di una fattoria in miniatura completamente automatizzata, basata sempre sull'utilizzo di sensori e attuatori, che possono agire in piena autonomia o essere gestiti mediante un Web server o una specifica APP per smartphone; il cuore del progetto è una board compatibile con l'IDE di Arduino, e basata sul modulo ESP32, che è un SoC (System On a Chip) Wi-Fi e Bluetooth da 2,4 GHz progettato da Espressif Systems,

per applicazioni mobili, elettroniche indossabili e Internet-of-Things (IoT).

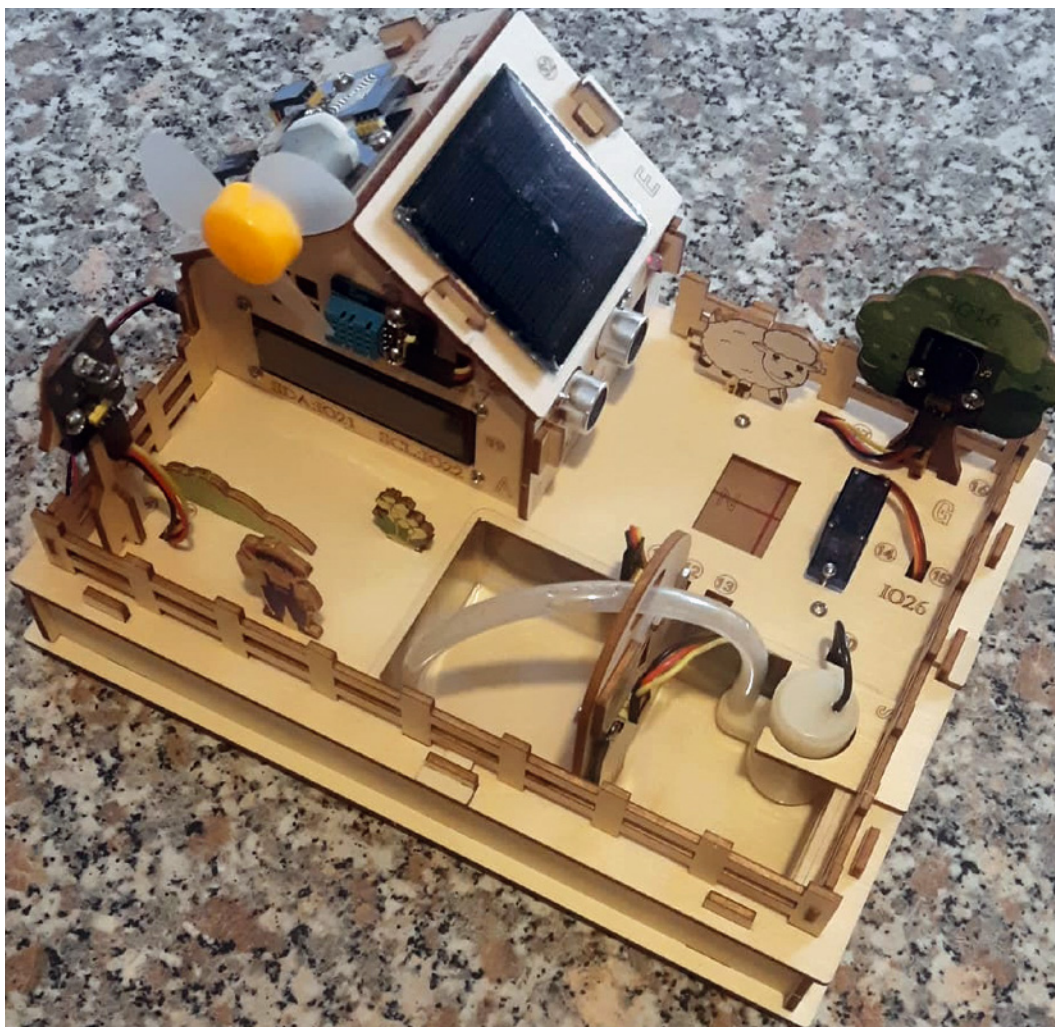
Il produttore del kit ha realizzato una pagina Wiki ricca di contenuti utili per far funzionare tutto. Il kit viene fornito in una scatola con l'immagine della villetta completa funzionante; per avere un'idea del risultato finale riportiamo la foto di quella montata da noi (**Fig. 1**).

Una volta aperta la confezione troviamo sei pannelli in balsa (il tipico legno da modellismo) contenenti tutte le parti necessarie per la realizzazione della struttura, già intagliate, che si possono staccare con una leggera pressione sui pezzi stessi; un ulteriore pannello in balsa con elementi colorati; un pannello e una ruota dentata in plexiglass; la meccanica è completata da viti, dadi, distanziali, tubi, vaschette, una ventola in plastica, cavetti jumper per i collegamenti, e alcuni piccoli attrezzi, indispensabili per il montaggio.

L'elettronica è composta da:

- Board KS5016 ESP32 Plus con cavetto USB C;
- Display LCD 16x2 con interfaccia I2C;
- Modulo PIR;
- Modulo Sensore Ultrasuoni;
- Modulo LED bianco;
- Modulo Fotoresistore;
- Modulo Sensore di temperatura e umidità;
- Modulo pulsante con cappuccio;
- Modulo motorino con ventola;
- Modulo Sensore di vapore/pioggia;
- Modulo Sensore dell'umidità del suolo;
- Modulo Sensore di livello dell'acqua;
- Modulo relè 5V;
- Modulo Buzzer passivo;
- Pompa elettrica per l'acqua;
- Servomotore 9G 180°;
- Pannello solare con LED RGB;
- Portabatterie 6 x AA.

Fig. 1
La ESP32 Smart
Farm montata.



La board e tutti i vari moduli e accessori sono di ottima qualità, facilmente identificabili dalle serigrafie e con le numerose foto presenti nella Wiki. Il montaggio richiede tempo e la giusta attenzione, sulla WIKI sono presenti sia un video che una pagina con tutte le fasi del montaggio ma, considerata la nostra esperienza diretta, abbiamo deciso di preparare un manuale in italiano ricco di immagini, con la descrizione di ogni singolo passaggio. In questo articolo di presentazione ci dedichiamo interamente alla parte software e a tal fine riporteremo i collegamenti dei singoli moduli alla board, in modo da poterli testare tutti facilmente prima di montarli definitivamente nella struttura. Oltre al manuale di montaggio abbiamo preparato un manuale d'uso della Smart Farm, sempre in italiano e anch'esso ricco di immagini, nel quale abbiamo descritto tutti gli sketch, approfondendo alcuni elementi utili per una migliore comprensione dell'uso dei moduli.

Il produttore ha creato una specifica sezione GitHub per poter scaricare sketch e librerie per i vari test; inoltre, come dicevamo, è stata realizzata anche una eccellente Wiki, dalla quale si accede a tutte le altre sezioni predisposte per il montaggio e i test del kit, ma saranno utili solo per qualche approfondimento, in quanto il manuale che forniremo in dotazione al kit conterrà tutte le necessarie istruzioni ed anche diversi tips e suggerimenti ricavati dalla nostra esperienza di montaggio e uso. La versione IDE consigliata è la 2.3.2 (l'ultima nel momento in cui abbiamo testato il kit), alla quale occorrerà aggiungere le librerie scaricate da GitHub. Vediamo i necessari passaggi per l'installazione della parte software.

L'IDE conviene copiarla direttamente nella root dell'hard disk, in caso contrario potrebbero sorgere problemi di lunghezza dei percorsi durante la copia. Prima di aprire l'IDE occorre collegare la board mediante il cavo USB-C in dotazione al kit e verificare che il driver del chip CH340 venga caricato correttamente, cosa che avverrà se in precedenza sul PC sono state usate altre board con il convertitore USB-seriale basato su questo componente. In caso contrario il sistema di Windows chiederà di indicare la posizione del driver, che potrà essere scaricato dal seguente link: <https://www.arduined.eu/ch340-windows-10-driver-download/> dove sono presenti anche le informazioni per la sua corretta installazione (Fig. 2).

Una volta che la board è correttamente collegata e configurata si può aprire l'IDE, andare in File-Preferences ed inserire il link della scheda ESP32

CARATTERISTICHE TECNICHE

della board ESP32

Tensione di lavoro 5V	Frequenza WiFi 2,4 GHz
Tensione di alimentazione (batteria) 9V	Protocollo Bluetooth v 4.2 Full Standard
Massima corrente in uscita ≤ 1,5 A	Dimensioni 255 x 185 x 130 mm ±1%
Consumo massimo ≤ 10 W	Temperatura di lavoro -10 °C ÷ +65 °C

(https://espressif.github.io/arduino-esp32/package_esp32_index.json) negli URL del gestore schede aggiuntive (Fig. 3).

Chiudere le Preferenze, cliccare sull'icona Boards Manager e scrivere *esp32* nel campo ricerca, quindi installare le board "esp32 by Espressif Systems"

Fig.2
Installazione
del driver
della board.



CH340 Windows 10 driver download

Download latest version of verified & working Windows 10 driver for CH340, CH340G USB to Serial converter. Tested with Arduino, ESP2866, Wemos D1 and other boards.

[CH340 driver for Windows 10](#)

Driver details

CH340, CH340G & CH341 (USB-SERIAL chip) V3.5
Driver Version = 2019-01-30, 3.5.2019.1

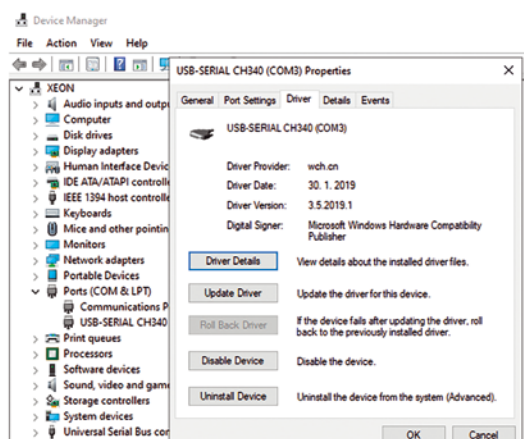
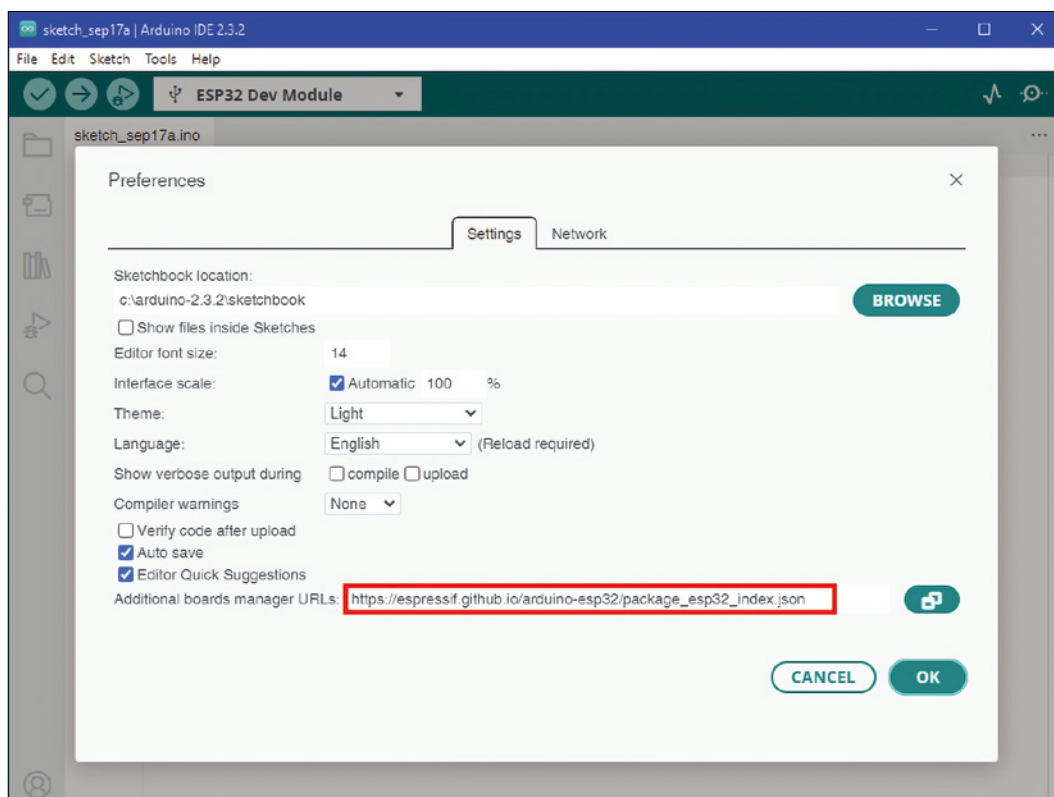


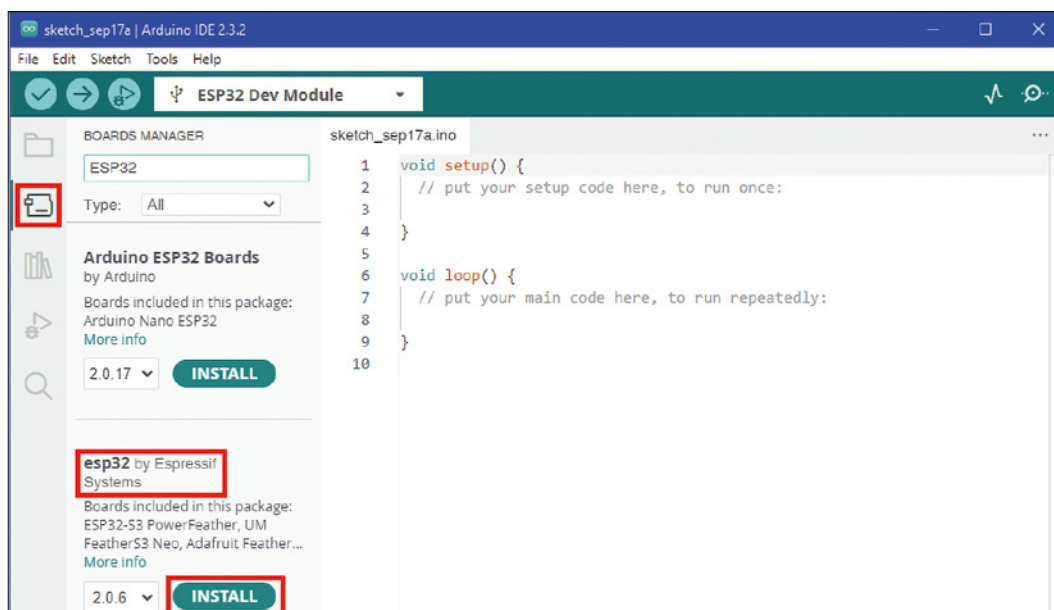
Fig. 3
Aggiornamento
Gestore schede
aggiuntive.



Nota bene: occorre fare ben attenzione a selezionare la versione **2.0.6** (Fig. 4), invece di quella proposta per default, altrimenti alcuni sketch non saranno compilati correttamente e daranno errore! Altre importanti indicazioni sono state inserite nel

manuale d'uso. Dopo l'aggiunta delle librerie si potrà iniziare con l'esecuzione degli sketch realizzati dal produttore. Poiché nel manuale abbiamo aggiunto immagini molto chiare dei collegamenti dei singoli moduli, si può anche pensare di

Fig. 4
Installazione board
esp32.



testarli rapidamente a banco, prima di montare la struttura della Smart Farm. Nella **Fig. 5** abbiamo riportato l'elenco completo degli sketch scaricabili da GITHUB.

Apriamo l'IDE e selezioniamo la board ESP32 Dev Module e la COM alla quale essa è collegata, quindi possiamo iniziare i nostri test, caricando gli sketch, uno per volta, e controllando che tutto funzioni correttamente. Naturalmente, nel caso in cui non si riscontri l'effetto descritto, occorrerà controllare prima di tutto i collegamenti e poi eventuali condizioni fisiche e ambientali della struttura, in base al tipo di sensore o attuatore sotto test; per esempio il sistema composto dal relè e pompa dell'acqua non funzionerà se la vasca nella quale è immersa non conterrà una quantità d'acqua sufficiente. Di seguito descriviamo alcuni sketch.

1.1 Blink: test del modulo LED montato nel cortile davanti alla Smart Farm e collegato al pin digitale io27 (**Fig. 6**); è il classico blink di Arduino, adattato a questa board, non appena viene caricato il LED bianco inizia a lampeggiare al ritmo di un'accensione al secondo.

Tutti gli sketch sono da considerare come suggerimenti d'uso, come si può ben comprendere osservando lo sketch alternativo (**1.1aSOS**) da noi predisposto, nel quale abbiamo simulato il classico segnale visivo di richiesta d'aiuto secondo il codice Morse (SOS): tre lampeggi brevi, tre lunghi, ancora tre brevi e poi una pausa, quindi ricomincia; il semplice codice è riportato nel **Listato 1**.

2.2 Light-Control-System: test del modulo fotore-sistore, montato sulla falda sinistra del tetto della Smart Farm e collegato al pin analogico IO34, in abbinamento al modulo LED, che consiste nell'accendere automaticamente il LED quando la luce ambientale scende al di sotto di una certa soglia, che in questo esempio è fissata a 800 ma che naturalmente può essere variata a piacimento. Oltre a vedere l'effetto immediato del funzionamento mediante l'accensione o spegnimento del LED, si può controllare il valore di lettura del fotore-sistore attivando il Serial Monitor (velocità 9600 baud) (**Fig. 7**).

3.3 Buzzer-Tone: test del modulo buzzer passivo montato sull'albero nel cortile e collegato al pin digitale IO16 (**Fig. 8**); lo sketch utilizza il comando **tone** che permette di inviare al buzzer un segnale con frequenza precisa. Vengono emessi inizialmente tre singoli suoni, corrispondenti alle note

1.1Blink	6.2Ultrasonic-Sensors
1.2PWM	6.3Intelligent-Feeding-System
1.3Button	7.1DHT11
1.4Self-Locking-Button	7.2LCD1602
1.5Lighting-System	7.3Motor
2.1Photocell-sensor	7.4Temperature-Control-System
2.2Light-Control-System	8.1Soil-Humidity-Sensor
3.1PIR-Motion-Sensor	8.2Soil-Humidity-Testing-System
3.2Passive-Buzzer	9.1Water-Level-Sensor
3.3Buzzer-Tone	9.2Water-Level-Testing-System
3.4Buzzer-Music.ino	10.1Water-Pump
3.5Alarm-System	10.2Auto-irrigation
4.1Steam-Sensor	11.1WiFi-HTML-HELLOWORLD
4.2Rainwater-Detection-System	11.2WiFi-HTML-Smart-Farm
6.1Servo	12.1APP-Smart-Farm

Fig. 5
Collegamenti e test sensore PIR.

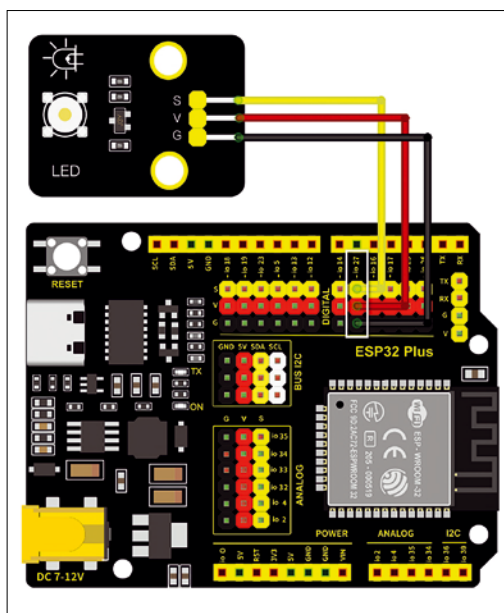


Fig. 6
Collegamento del modulo LED.

musicali Do-Re-Mi della quinta ottava; successivamente viene emessa una serie rapidissima di 80 suoni a frequenza crescente, e subito dopo la stessa serie ma a frequenza decrescente, quindi il ciclo ricomincia.

Chi volesse eseguire qualche test con le note musicali, nel manuale d'uso troverà una tabella con tutte le frequenze corrispondenti alle note principali delle ottave dalla zero alla settima.

3.5 Alarm-System: questo codice fa uso di tre moduli: LED, sensore PIR e buzzer passivo e simula un semplice sistema di allarme. Dopo aver caricato lo sketch nella board occorre aprire il Serial Monitor (velocità 9600 baud). In condizioni di riposo la spia del PIR è accesa, il LED ed il buzzer sono spenti, sul Serial Monitor esce il messaggio **"No one"**; nell'i-

▼ Listato 1 Sketch 1.1aSOS

```
#define LED_BUILTIN 27 //LED pins

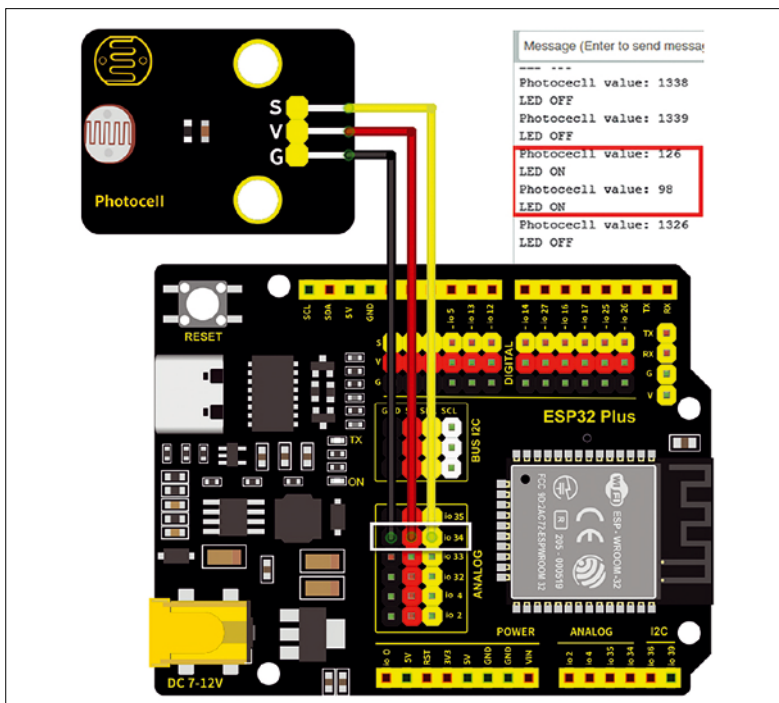
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  for (int i = 1; i<=3; i++)
  {
    digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
    delay(250); // wait for a second
    digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
    delay(250);
  }
  delay (500);

  for (int i = 1; i<=3; i++)
  {
    digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
    delay(500); // wait for a second
    digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
    delay(500);
  }
  delay (500);

  for (int i = 1; i<=3; i++)
  {
    digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
    delay(250); // wait for a second
    digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
    delay(250);
  }
  delay (2000); // wait for a second
}
```

Fig. 7
Accensione del
LED controllata dal
Fotoreistore.



stante in cui viene rilevata una presenza la spia del PIR si spegne, il LED inizia a lampeggiare ed il buzzer emette un segnale sonoro modulato come allarme, mentre sul Serial Monitor il messaggio cambia in "**Someone**" (Fig. 9).

4.2 Rainwater-Detection-System: test del sensore di vapore/pioggia montato sulla falda sinistra del tetto della Smart Farm e collegato al pin analogico IO35. Questo codice usa anche il buzzer passivo. Dopo aver caricato lo sketch nella board occorre aprire il Serial Monitor (velocità 9600 baud). Vengono fissate tre soglie di lettura dei valori; da 800 a 2000, da 2001 a 4000, oltre i 4000 (fino a 4096 massimo). In condizioni normali il Serial Monitor indica il valore 0 e il buzzer è spento. Appena il sensore percepisce una quantità di vapore o umidità (sotto forma di condensa) o di acqua tale da superare il valore 800 viene emessa una serie ripetuta di tre note con frequenza di 200 Hz, se il valore supera i 2000 ma non i 4000 le note assumono la frequenza di 400 Hz, infine oltre il

valore 4000 la frequenza diventa di 600 Hz. In tal modo dal tipo di suono che si sente nel buzzer si può ricavare un'indicazione di massima della situazione ambientale. Naturalmente si può agire sia sul valore dell'istruzione **tone** per ottenere frequenze diverse, che sui valori delle tre soglie (Fig. 10).

6.3 Intelligent-Feeding-System: questo sketch usa il servomotore montato nel cortile di fronte l'albero e collegato al pin digitale IO26 e il modulo sensore ultrasuoni montato sulla parete destra della Smart Farm e collegato ai pin digitali IO12 e IO13 (Fig. 11). Questo sensore funziona emettendo una frequenza ultrasonica mediante il comando del pin Trigger e ricevendo il segnale di ritorno, dopo aver raggiunto un ostacolo, sul pin Echo; il tempo trascorso tra l'invio del segnale e la ricezione dello stesso permette di calcolare la distanza dell'ostacolo.

Dopo aver caricato lo sketch nella board occorre aprire il Serial Monitor (velocità 9600 baud) che istantaneamente indicherà la misura di 9 cm o maggiore (da intendere come assenza di ostacoli nel cortile).

Nel momento in cui si mette un oggetto davanti al sensore ad ultrasuoni, in modo da rilevare una misura tra 2 e 7 cm, istantaneamente si attiverà il servomotore facendo aprire la botola presente sul pavimento del cortile; rimuovendo l'oggetto il sensore rileverà una misura di valore uguale o maggiore di 9 cm e la botola si richiuderà.

7.4 Temperature-Control-System: questo sketch utilizza il sensore di umidità e temperatura (montato sulla parete anteriore della Smart Farm e collegato al pin digitale IO17), il motore con la ventola (posto sulla falda sinistra del tetto della Smart Farm e collegato ai pin digitali IO18 e IO19) e il display LCD (che si trova sulla parete anteriore della Smart Farm ed è collegato alla linea del BUS I²C) (Fig. 12).

Dopo il caricamento dello sketch nella board, i valori di umidità e temperatura vengono mostrati sul display LCD; se la temperatura ambientale supera i 25 °C viene acceso il motore con la ventola come sistema di raffreddamento.

Il motore può girare in senso antiorario, mediante l'istruzione **ledcWrite(1, xxx)**; nella quale xxx è un valore tra 0 (motore fermo) e 256 (massima velocità), oppure in senso orario tramite l'istruzione **ledcWrite(3, xxx)**; nella quale xxx è un valore tra 0 (motore fermo) e 256 (massima velocità).

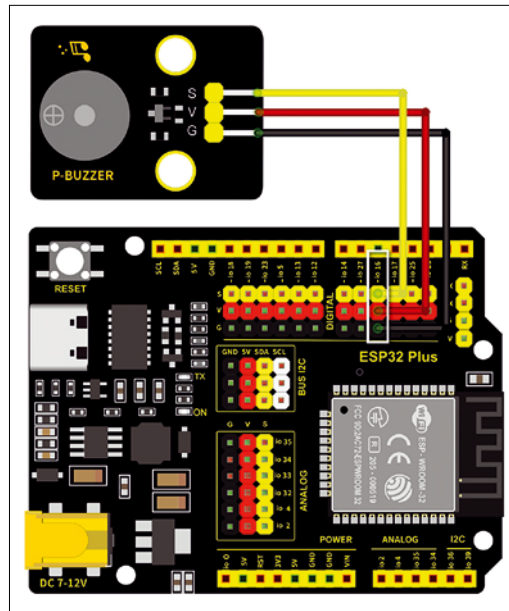


Fig. 8
Collegamento
del modulo
Buzzer pas-
sivo.

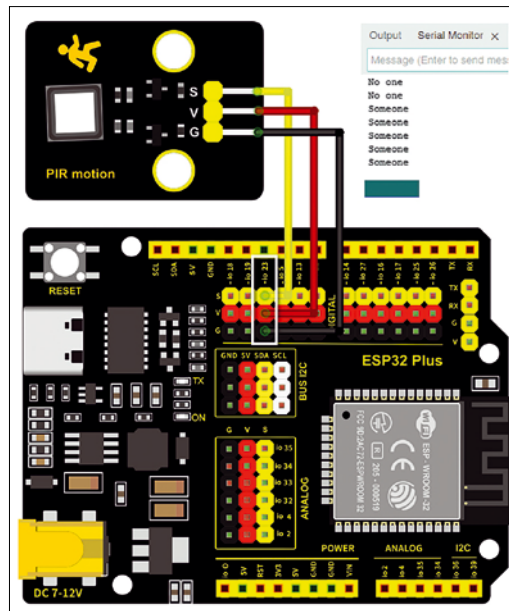


Fig. 9
Collegamen-
to e test del
modulo PIR.

Naturalmente, una volta decisa la direzione bisogna valorizzare la relativa istruzione e impostare a 0 quella che gestisce la direzione opposta. Inoltre, se si prevede un repentino cambio di direzione, è sempre opportuno fare arrestare prima il motore settando a zero entrambe le istruzioni, e impostando un ritardo di almeno 200 ms, per poi riavviare il motore nella direzione opposta, altrimenti si potrebbe avere un forte assorbimento in corrente, che potrebbe resettare la board ESP32 ed anche danneggiare il motore.

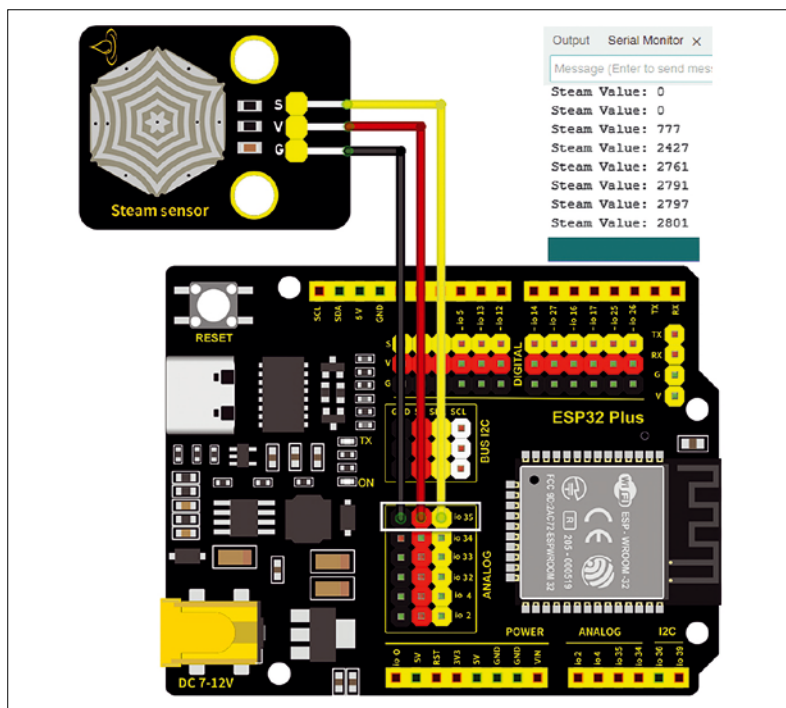


Fig. 10
Collegamento e test del modulo sensore vapore/pioggia

Naturalmente è possibile modificare la temperatura di soglia, portandola dai 25 °C attuali a quella desiderata, maggiore o minore, modificando la riga 46 dello sketch.

10.2 Auto-irrigation: questo sketch gestisce un sistema più complesso di quanto visto finora, in quanto gli elementi coinvolti sono: il sensore di livello dell'acqua, situato all'interno della vasca più esterna (prima vasca) e collegato al pin analogico IO33, il sensore di umidità del suolo, situato all'interno della vasca più interna (seconda vasca) e collegato al pin analogico IO32, il relè (localizzato sulla base inferiore della struttura, nei pressi della vasca esterna e collegato al pin digitale IO25) di comando della pompa dell'acqua, il pulsante (montato sulla parete posteriore della Smart Farm e collegato al pin digitale IO5) ed il display LCD.

Nella **Fig. 13** mostriamo i collegamenti dei due sensori, naturalmente nel manuale saranno visibili anche tutti gli altri.

Lo scopo di questo test è quello di controllare costantemente l'umidità del terreno presente nella seconda vasca ed il livello dell'acqua contenuta nella prima vasca, mostrando i rispettivi valori sul display LCD; in caso di basso livello di valore in una o entrambe le vasche scatta un allarme sonoro; l'allarme è disattivabile con la pressione del pulsante.

Nota bene: affinché questo esperimento funzioni correttamente la vasca dell'acqua inizialmente deve essere riempita almeno per 1/3 della sua capienza, altrimenti la pompa non partirà; se

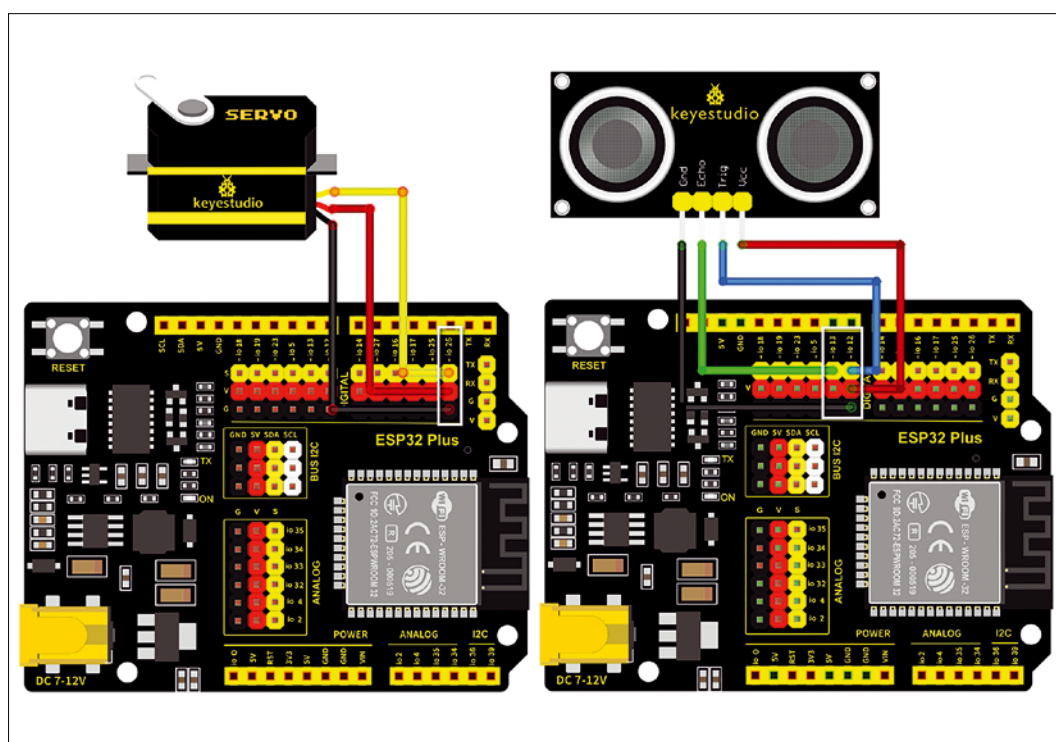


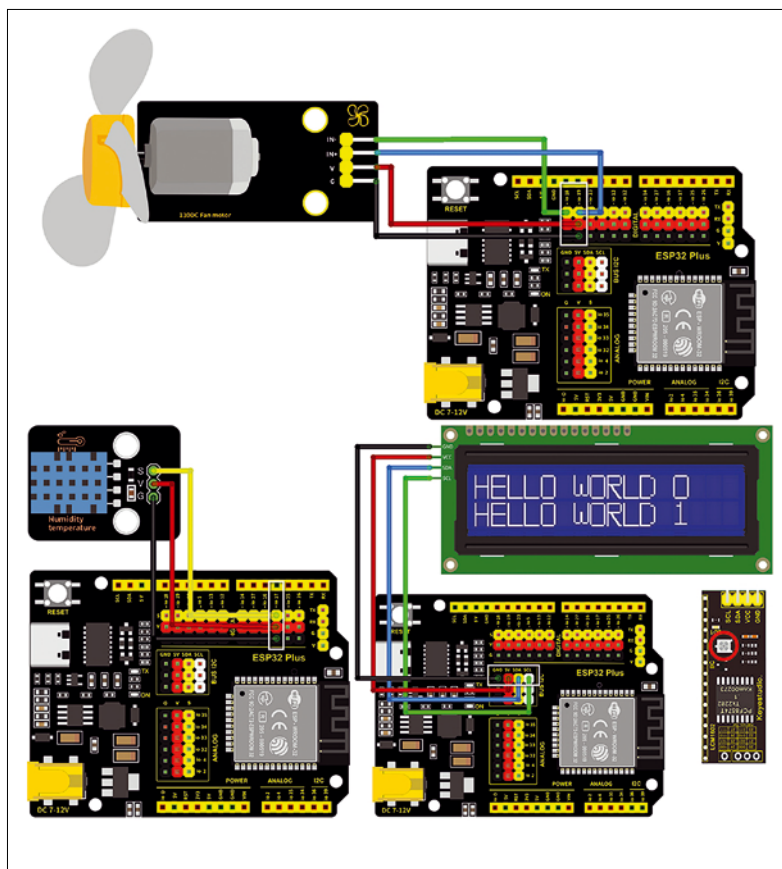
Fig. 11
Collegamenti del servomotore e del sensore ultrasuoni.



dovesse partire ma senza inviare acqua nel tubo, significa che internamente è vuota, in questo caso occorrerà riempire completamente la vasca (fino a 4-5 mm dall'orlo); quando la pompa funzionerà correttamente si potrà anche abbassare il livello dell'acqua nella vasca. Dopo aver caricato lo sketch nella board occorre aprire il Serial Monitor (velocità 9600 baud); il Serial Monitor serve per mostrare lo stato del pulsante di attivazione/disattivazione dell'allarme sonoro. Il display LCD mostrerà subito i valori del livello dell'acqua e dell'umidità del suolo, rilevati dai rispettivi sensori nelle vasche di appartenenza (**Fig. 14**).

Il codice prevede queste quattro routine cicliche:

- 1 – Lettura dello stato della variabile legata al pulsante: inizialmente è 0, premendo il pulsante una prima volta diventa 1, ripremendolo ridiventa 0 e così via; il valore della variabile incide sugli allarmi: se è 0 funzionano ed il buzzer emette i toni, se è 1 il buzzer viene disattivato. Il valore della variabile viene mostrato sul Serial Monitor;
- 2 – Controllo del livello di umidità del suolo nella seconda vasca: se il livello è minore o uguale a 500 vengono emessi tre toni di allarme (a patto che la variabile del pulsante sia uguale a 0);
- 3 – Controllo del livello di acqua nella prima vasca: se il livello è minore o uguale a 500 vengono emessi tre toni di allarme (a patto che la variabile del pulsante sia uguale a 0);
- 4 – Controllo umidità del suolo e livello dell'acqua: se l'umidità del suolo è minore o uguale a 500 e il livello di acqua è maggiore o uguale a 1000 viene attivato il relè che controlla la pompa dell'acqua in modo che la vasca del terreno



riceva sufficiente acqua per portare la lettura del sensore di umidità ad un valore maggiore di 500.

I due prossimi sketch servono per la gestione completa della Smart Farm e naturalmente sfruttano

Fig. 12
Collegamenti motore, sensore DHT11 e display LCD.

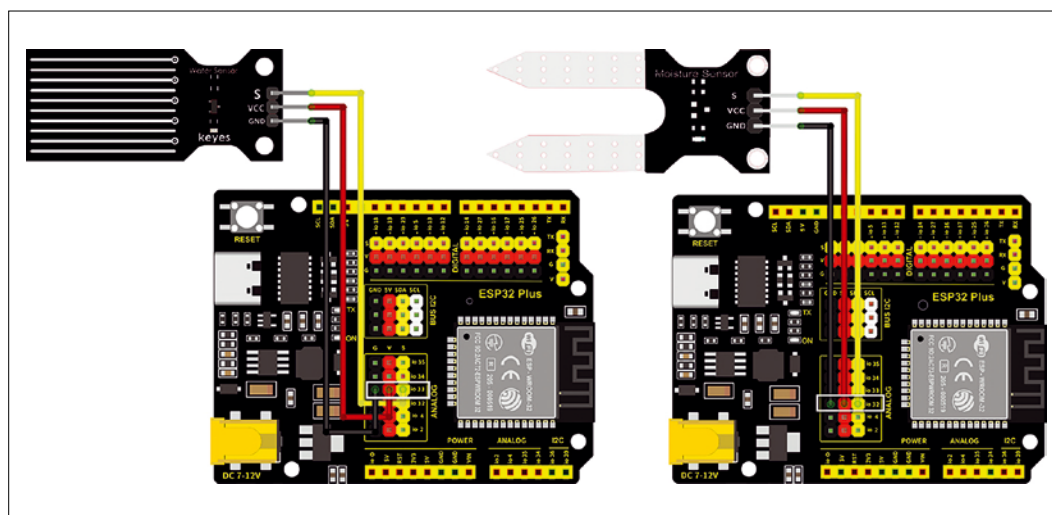


Fig. 13
Collegamenti dei sensori di livello acqua e umidità del suolo.



Fig. 14
Indicazione dell'umidità del suolo e del livello dell'acqua sul display LCD.

la dotazione Wi-Fi della board ESP32; la differenza sostanziale consiste nel fatto che il primo utilizza un Web server mentre il secondo necessita di un'APP creata appositamente sia per Android che per IOS. Naturalmente è necessaria la presenza di una rete Wi-Fi domestica alla quale si dovrà collegare la board ESP32 fornendo le credenziali.

11.2 WiFi-HTML-Smart-Farm: questo sketch fa uso di quasi tutte le librerie caricate in fase di installazione, oltre ad alcune già in dotazione all'IDE di Arduino.

Prima del caricamento sulla board occorre aggiungere i dati della propria rete Wi-Fi nelle due righe 33-34, predisposte per SSID e Password.

Lo scopo dimostrativo è quello di una gestione completa della Smart Farm tramite Web Server; sono coinvolti i sensori: umidità e temperatura, vapore/pioggia, umidità del suolo e livello dell'acqua, la pompa dell'acqua ed il relè, il fotoresistore, il LED, il servomotore e il motore con la ventola. Dopo il caricamento dello sketch bisogna aprire il Serial Monitor (velocità 9600 baud) sul quale verrà mostrato il messaggio di collegamento alla rete Wi-Fi con l'indirizzo IP ottenuto. A questo punto bisogna aprire un browser, digitare lo stesso indirizzo e si aprirà una pagina con un riquadro/display che mostrerà in tempo reale i valori di temperatura e umidità ambientale, la percentuale

TEST HTML ESP32

(2) WhatsApp

Non sicuro http://192.168.1.16

Sensor Data

Temperature:25°C
Humidity:49%rh
WaterLevel:33%
Steam:75.29%
Light:3167
SoilHumidity:27%

LED

Fan

Feeding

Watering

11.1WiFi-HTML-HELLOWORLD.ino

```

1  #include <WiFi.h>
2  #include <WebServer.h>
3
4  const char* ssid = "..."
5  const char* password = "..."

```

Output

Serial Monitor

Message (Enter to send me)

Connecting to WiFi...
Connecting to WiFi...
Connecting to WiFi...
Connected to WiFi
192.168.1.16
Web server started

Fig.15
Collegamento al Wi-Fi e Web server per il controllo della Smart Farm.

del livello dell'acqua nella prima vasca, la percentuale di vapore/pioggia, la luminosità ambientale e la percentuale di umidità del suolo rilevata nella seconda vasca (**Fig. 15**).

Sotto il riquadro dei dati sono presenti quattro pulsanti:

- LED: ad ogni pressione il LED cambia stato, passando da spento ad acceso e viceversa;
- Fan: permette di avviare la ventola, una nuova pressione la arresta;
- Feeding: apre la botola nel cortile, una nuova pressione la richiude;
- Watering: avvia la pompa dell'acqua trasferendola dalla vasca esterna a quella interna.

12.1 APP-Smart-Farm: questo programma è molto simile al precedente, la differenza sostanziale consiste nel fatto che in questo caso la Smart Farm viene gestita mediante un APP Android (ma esiste anche la versione IOS) invece che tramite il Web server. L'APP si chiama **"IOT farm"** ed è scaricabile dal Play Store.

Bisogna prima di tutto caricare lo sketch sulla board e quindi aprire il Serial Monitor (velocità 9600 baud), sul quale si vedranno le operazioni di connessione alla rete Wi-Fi, occorre prendere nota dell'indirizzo IP.

A questo punto si può aprire l'APP ed inserire l'indirizzo IP nel campo in alto della schermata che appare sullo smartphone, quindi spostare l'interruttore (il pallino posto accanto all'indirizzo) verso destra in modo da collegare il cellulare con la Smart Farm, il messaggio **"Successfully Connected"** indicherà che il collegamento è andato a buon fine (contestualmente uscirà un messaggio di conferma sul Serial Monitor), in caso contrario occorrerà ricontrollare l'indirizzo IP inserito. La schermata indicherà in tempo reale tutti i valori rilevati dai sensori coinvolti: temperatura e umidità dell'aria, umidità del suolo (seconda vasca), luce ambientale, livello dell'acqua (prima vasca), percentuale di vapore/pioggia (**Fig. 16**).

Nella schermata dell'APP sono presenti anche 5 pulsanti, ognuno con una specifica funzione on/off:

- LED: accende o spegne il LED;
- Watering: avvia la pompa dell'acqua ad ogni pressione;
- Fan: accende o spegne il motore con la ventola;
- Feeding: apre o chiude la botola nel cortile;
- Music: diffonde una musicchetta tramite il buzzer passivo.

In definitiva questo kit, oltre ad essere un ottimo passatempo, è anche un eccellente strumento

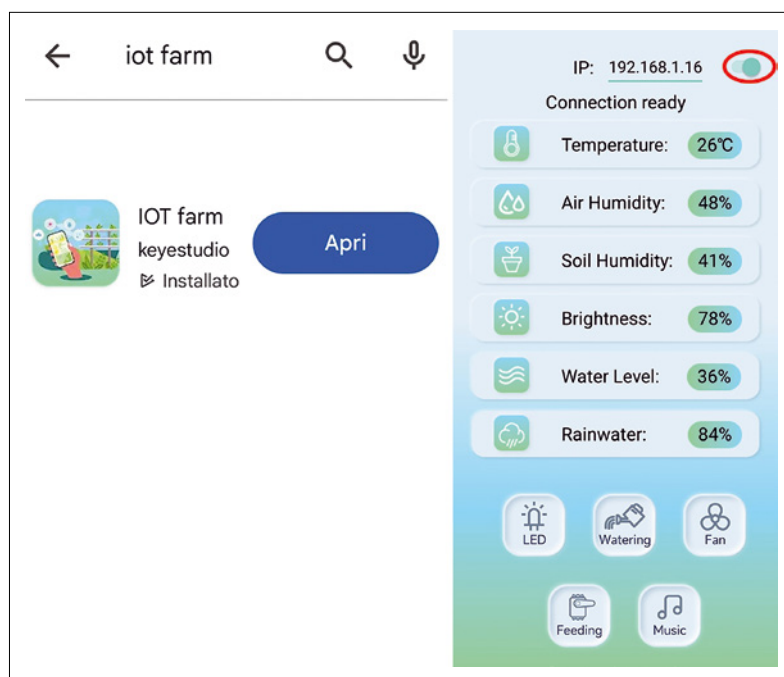


Fig. 16
APP per il controllo della Smart Farm.

didattico, che permette di prendere dimestichezza sia con il mondo Smart che con l'uso del Wi-Fi in ambiente IDE Arduino.

Termina qui la nostra presentazione di questo interessantissimo kit che permetterà di entrare nel mondo della gestione wireless di un ambiente smart, con possibilità di sperimentare e apprendere l'utilizzo dei molti moduli presenti nel kit. Il produttore ha preferito non inserire uno sketch di automazione completa, in modo da lasciare all'utente la possibilità di scrivere il proprio software in base alle esigenze personali; sarà sufficiente concatenare tra loro il codice dei vari sketch singoli e gestirli opportunamente.

Non ci resta che augurarvi buon divertimento! ■

COSA OCCORRE ?		
Codice	Descrizione	Prezzo € IVA inclusa
EDUKIT15	Kit Smart Farm con ESP32 e manuale	129,00
IL MATERIALE VA RICHIESTO A:		
Futura Elettronica, Via Adige 11, 21013 Gallarate (VA) Tel: 0331-799775 - www.futurashop.it		

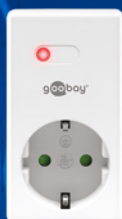
Tutto sotto RADIO CONTROLLO

Set presa da rete radiocomandata a tre canali espandibile con telecomando



Ideale per controllare dispositivi elettrici con un carico massimo di 1100 watt. Offre la possibilità di aggiungere ulteriori trasmettitori e abbinare fino a tre prese, ciascuna associata ai tre canali operativi del trasmettitore. Particolarmente indicato per i punti difficili da raggiungere negli spazi interni, offrendo comodità e facilità d'uso. Il sistema è espandibile.

cod. WT3PRTX
€ 22,90



Telecomando
addizionale per set
WT3PRTX

cod. WT3PR
€ 14,90



Presa radiocomandata
addizionale per set
WT3PRTX

cod. WT3TX
€ 9,00

Sistema di radiocontrollo a 433 MHz

Trasmettitori a 1-2-4 canali e ricevitori a 12VDC e 220VAC operanti sulla banda dei 433 MHz. Si basano sulla tecnologia RF che fornisce un segnale stabile e a lunga distanza. I canali sono tra di loro indipendenti e consentono, oltre al controllo del carico, di eliminare qualsiasi interferenza reciproca. Bassissimo consumo in modalità stand-by. Possono essere impiegati per il controllo di luci, per l'attivazione/disattivazione di impianti d'allarme, ecc.

RICEVITORE

2 canali 220 VAC

Cod. EMY220RX2

Euro 16,00

2 canali 12 VDC

Cod. EMY12RX2

Euro 16,00

4 canali 12 VDC

Cod. EMY12RX4

Euro 19,00

TRASMETTITORE

1 canale

Cod. EMYTX1

Euro 11,00

TRASMETTITORE

2 canali

Cod. EMYTX2

Euro 12,00

TRASMETTITORE

4 canali

Cod. EMYTX4

Euro 13,00

RICEVITORE

1 canale 220 VAC

Cod. EMY220RX1

Euro 13,00

1 canale 12 VDC

Cod. EMY12RX1

Euro 13,00



Kit radiocomandi

Kit radiocomandi da esterno

2CH – 433MHz

12V – 24V – AC/DC

cod. SUPRK002

€ 30,00



Kit radiocomandi da esterno

2CH – 433MHz

220Vac



cod. SUPRK004

€ 44,00

Kit radiocomandi da interno

2CH – 433MHz

12V – 24V DC

cod. SUPRK001

€ 24,00



Prezzi IVA inclusa.

WIRELESS

REGOLARE LA LUMINOSITÀ CON LoRa

di TIZIANO ROSSI



Utilizziamo la tecnologia LoRa per regolare con precisione l'intensità delle lampade LED, offrendo un controllo preciso e privo di interferenze rispetto ai tradizionali sistemi a infrarossi.

In un'epoca in cui l'efficienza energetica e la personalizzazione dell'ambiente domestico diventano sempre più importanti, la gestione tradizionale dell'illuminazione attraverso sistemi a infrarossi mostra significative limitazioni, tra cui la necessità di visibilità diretta e la suscettibilità a interferenze. Questo articolo affronta tali sfide introducendo un innovativo sistema di controllo dell'intensità luminosa per lampade LED a media potenza. La tecnologia LED ha una maggiore efficienza di una qualsiasi lampada tradizionale, pertanto svilupperemo un progetto che permetta la variazione dell'intensità luminosa, senza intaccarne l'efficienza, tenendo presente le caratteristiche che vengono riportate dal costruttore. La variazione dell'intensità luminosa dei LED potrebbe

essere controllata da un potenziometro lineare: questa è una soluzione, ma abbiamo preferito virare su un controllo a distanza (più pratico e comodo).

Una possibilità è quella di utilizzare un telecomando IR ma, come accennato all'inizio, funzionante solo in condizioni in cui sia possibile avere il raggio IR diretto al ricevitore. Tenendo presente le considerazioni sopra riportate abbiamo preferito optare per una tecnologia LoRa (Long Range), sia per la trasmissione che per la ricezione, superando questa limitazione in quanto può lavorare in qualsiasi condizione ambientale ci dovessimo trovare (vicino, lontano e con barriere architettoniche).

Si tratta di una tecnologia di comunicazione wireless punto-punto che sfrutta la banda libera

utilizzata in Europa a 868Mhz, per cui non ha necessità che vi sia un campo libero.

Esistono in commercio vari moduli ricetrasmittitori LoRa a 868Mhz che possono fare al caso nostro, ma, per questioni di prezzo, facilità di comando e piccole dimensioni, abbiamo optato per una breakout chiamata RFM95W (d'ora in poi per comodità la chiameremo, modulo LoRa). Deciso cosa utilizzare per comandare la fonte di illuminazione a LED resta da affrontare come mantenerne intatte le caratteristiche, anche quando abbiamo la necessità di variare l'intensità luminosa.

A questo proposito la tecnica migliore è comandare la variazione dell'intensità luminosa tramite la modulazione della larghezza di impulso PWM (Pulse Width Modulation).

Per approfondire il perché si utilizza la tecnica di modulazione della larghezza di impulso vi rimandiamo all'approfondimento che trovate nell'appro-

PULSE WIDTH MODULATION

Diversamente dall'illuminazione a incandescenza o a fluorescenza tradizionale, non è facile variare la luminosità dei LED conservando la qualità della luce.

Per ottenere ciò ci dobbiamo avvalere del controllo dei LED tramite il "dimmeraggio" (parola non di uso corrente). Viene tradotta dall'inglese DIMMER (VARIATORE), ma, nonostante ciò, viene utilizzata nel linguaggio comune per, appunto, definire una variazione: in questo caso, per il controllo della variazione dell'intensità luminosa.

Una tecnica affermata è quella di ridurre la luminosità del LED utilizzando la modulazione della larghezza di impulso (PWM) che alimenta i LED. Si tratta di una frequenza stabile, ma con la variazione della larghezza dell'impulso ON E OFF. La premessa di base è che durante il ciclo "ON" (di attivazione del treno PWM), il

LED opera in condizioni di corrente/tensione dirette ottimali. Di conseguenza, la qualità della luce è alta e la luminosità percepita è linearmente proporzionale al ciclo di lavoro del treno PWM. Le recenti linee guida sulla progettazione dell'illuminazione suggeriscono che la frequenza di lavoro debba essere stabile e superiore a 80/100 Hz se si vogliono evitare effetti a lungo termine sulla salute, determinati dal così detto "sfarfallio".

Esistono diversi articoli che approfondiscono quale sia la corrente ottimale per ottenere la massima efficienza dei LED (espressa in LUMEN per watt e KELVIN), ma, a questo proposito, è meglio attenersi a quanto viene riportato nel manuale del produttore (vedi esempio la TABELLA 1 sotto riportata).

Fino a pochi anni fa, l'inten-

sità del fascio di luce che una lampada era in grado di emettere veniva comunemente indicata in termini di WATT, ovvero l'unità di misura della potenza per eccellenza. La tendenza era che ad un wattaggio superiore corrispondesse necessariamente una "quantità di luce" superiore. Oggi, con l'arrivo della tecnologia LED nelle nostre case, questo mito è stato sfatato, e proprio in virtù di ciò in tutti i moderni cataloghi di illuminazione compare il parametro "Lumen".

Cosa indica il Lumen?

Misura l'intensità di un flusso luminoso, artificiale o naturale che sia. In parole povere, ci indica la quantità di luce emessa dalla nostra lampadina e ci permette di scegliere il prodotto adatto alle nostre esigenze, considerando fattori come l'altezza dal suolo, l'ampiezza della porzione illuminata e persino l'atmosfera

che vogliamo ricreare. Esistono principalmente tre modi con i quali i lumen vengono indicati:

1. lm/m = viene di solito usato per le strisce LED o altri prodotti di lunghezza notevole ed indica quanti lumen per ogni metro di lunghezza del prodotto vengono emessi. Ad esempio, una striscia da 1.200 lm/m emetterà 1.200 lumen per ogni metro di lunghezza. Con 5 metri di striscia avremo quindi 6.000 lumen complessivi;
2. lm/W = viene usato per la maggior parte dei prodotti LED ed indica quanti lumen per ogni Watt di potenza del prodotto vengono emessi. Questo valore indica anche l'efficienza di un prodotto. Infatti, maggiore è il numero di lumen che un prodotto è in grado di emettere, maggiore sarà la sua luminosità a parità di potenza. Ad esempio, un faro con 10 W di potenza e con una efficienza di 120 lm/W emetterà

sito riquadro in queste pagine.
Non ci resta che raccogliere tutte le informazioni trattate e metterle in pratica

SCHEMA A BLOCCHI

Utilizzando il **trasmettitore** come un telecomando, quindi, portatile, dobbiamo necessariamente alimentarlo con una batteria che, nel nostro caso, si compone di una batteria a Litio ricaricabile (una ICR14500). Considerando che la tensione erogata da una batteria a Litio può variare da 3V (completamente scarica) a 4.2V (massima tensione di carica), per rispettare le caratteristiche di alimentazione del modulo LoRa (U3), che vanno da un minimo di 1.8V a un massimo di 3.7V, abbiamo introdotto un regolatore di tensione di 3.3V (U2), in modo da avere una tensione stabile che ci permette di controllare l'alimentazione sia in fase di ricarica della batteria, sia quando alimentato direttamente dalla batteria stessa.

Il controllo del modulo LoRa (disponibile sul sito *futurashop.it* con codice: RFM95-868) avviene tramite il microcontrollore Atmega328 (U1).
Il display LCD ci serve per visualizzare il dato numerico, espresso in percentuale, dell'intensità luminosa dei LED. L'aumento o la diminuzione di questo dato avviene premendo singolarmente i due mini-tasti S1 e S2. Lo schema a blocchi del **ricevitore** si compone di un ingresso di alimentazione (J2), che può variare da 9V a 12V DC (importante è mantenersi entro questo range di alimentazione). I regolatori di tensione U3 (5V) e U4 (3.3V), ci forniscono rispettivamente l'alimentazione per il modulo PWM e l'alimentazione del modulo LoRa (U2) che, come spiegato in precedenza, deve essere alimentato non oltre i 3.7V; inoltre, il regolatore da 3.3V alimenta il microcontrollore Atmega328, dedicato a controllare il modulo LoRa e alla generazione della forma d'onda per la modulazione PWM.

complessivamente 1.200 lumen mentre uno con un'efficienza di 144lm/w e la stessa potenza di 10W emetterà 1.440 lumen quindi il 20% in più del faro precedente con lo stesso identico consumo di energia;

3. lm= indica semplicemente la quantità di lumen emessi in totale da quella lampada LED, e

quindi la luminosità totale della lampada.

Cosa indicano i Kelvin?

Un altro punto su cui spesso ci si confonde riguarda un'altra unità di misura, che vedrete comparire ovunque e in generale su tutti i prodotti LED, ovvero il Kelvin (abbreviato con K). Questo parametro indica la tempera-

tura e, nel mondo dell'illuminazione, viene utilizzato per esprimere la tonalità di colore dei dispositivi.

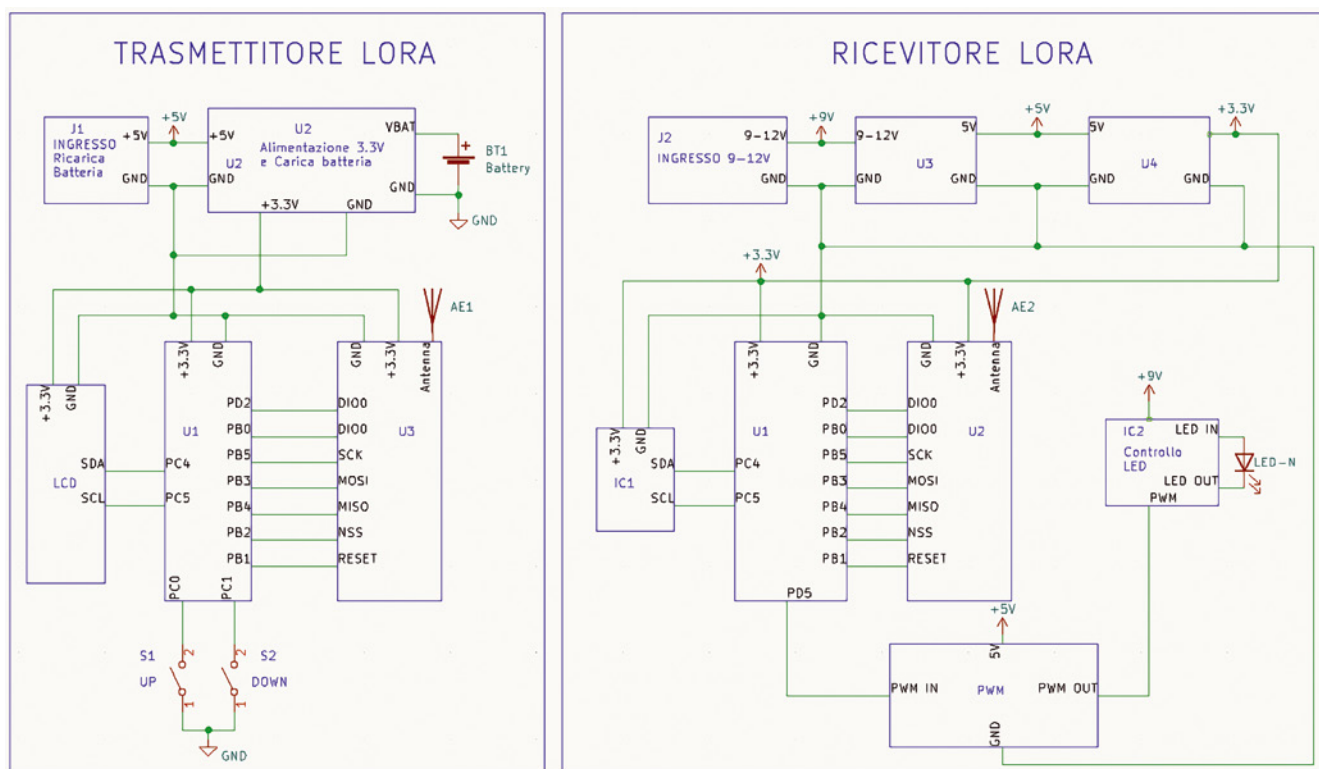
- Nel range compreso tra 2700 e i 3500 K, avremo una sfumatura di bianco tendente al rosso, definita come bianco "caldo", in grado di comunicare calore e accoglienza.
- Dai 4000 fino ai 4500 K è la volta

del cosiddetto bianco "naturale", una tonalità che cerca di essere la più neutra possibile e di replicare la luce solare.
6. Tra i 5000 e i 6500 K, il bianco assume una tonalità "glaciale" chiamata bianco "freddo" o bianco ghiaccio.

Tabella 1

NOMINAL CCT ^[1]	MINIMUM CRI ^[2]	LUMINOUS FLUX ^[3] (lm)		TYPICAL LUMINOUS EFFICACY (lm/W)	TYPICAL LUMINOUS FLUX (lm)	TYPICAL LUMINOUS EFFICACY (lm/W)
		MINIMUM	TYPICAL			
		120mA				
3000K	70	98	108	148	92	153
3500K	70	100	111	152	95	158
4000K	70	101	116	158	99	165
5000K	70	101	116	158	99	165
5700K	70	101	115	157	98	164
6500K	70	101	115	157	98	164





Schema a blocchi del trasmettitore e del ricevitore LoRa.

Al fine di mantenere in memoria il dato numerico ricevuto dal trasmettitore (che definisce l'intensità luminosa dei LED), anche quando spegniamo e riaccendiamo il ricevitore, abbiamo introdotto una memoria EEPROM (IC1).

Dato l'assorbimento dei LED, che, come vedremo in seguito, può aggirarsi sugli 80 mA cadauno, non possiamo collegare l'uscita della modulazione PWM, generata dal microcontrollore, direttamente ai LED ma, a questo scopo, abbiamo inserito un'interfaccia indicata come "PWM" e IC2.

IL TRASMETTITORE

Alimentato da una batteria (BT1) da 3.6V 800mA (cod: 5600-LIR14500FT) può essere ricaricata tramite la porta micro-USB (J1) collegata ad una comune presa USB 5V (Fig. 1).

Il led LD1, in combinazione con l'accensione del led LD3, indica che il trasmettitore è alimentato e che la batteria è in fase di ricarica. Lo spegnimento del led LD3 ci indica che la ricarica della batteria BT1 è completata.

La corrente di ricarica è controllata da U2 e determinata dal valore di R2. Applicando la formula riportata nel datasheet del costruttore ($I_{REG} = 1000V/R_{PROG}$), possiamo definire la corrente di ricarica della batteria, che normalmente si aggira

su un decimo della sua capacità.

Pur essendo alimentato a batteria, abbiamo dovuto introdurre uno stabilizzatore di tensione, composto da IC1 (3.3V). In questo modo avremo una tensione di alimentazione stabile che serve ad alimentare il microcontrollore (U1), il display LCD e il breakout RFM95W.

Il display OLED 128x32 0.91 pollici ci permette di visualizzare il dato trasmesso al ricevitore per aumentare o diminuire la modulazione PWM. Abbiamo suddiviso il display in due righe: nella riga superiore viene visualizzata una barra luminosa, la cui lunghezza è proporzionale al dato trasmesso al ricevitore tramite il modulo LoRa. Nella seconda riga, viene visualizzato il dato, trasmesso al ricevitore, come valore percentuale con a fianco la scritta "power", per cui uno 0% corrisponde alla minima intensità luminosa e, per ovvie ragioni, il 100% corrisponde alla massima intensità luminosa (Fig. 2).

Utilizzando un microcontrollore Atmega328 (U1) per comandare il display LCD e il modulo LoRa, la programmazione del microcontrollore potrebbe avvenire nell'ambiente di sviluppo (IDE) di Arduino, sfruttando le librerie necessarie per comandare la comunicazione LoRa e il display OLED da 0.91 pollici.

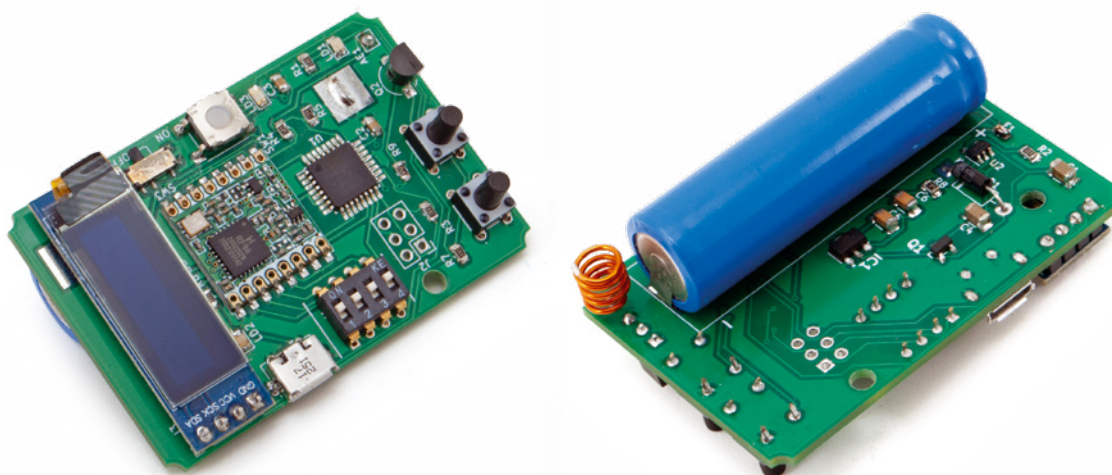


Fig.1
Il trasmettitore
montato.

schema elettrico TRASMETTITORE

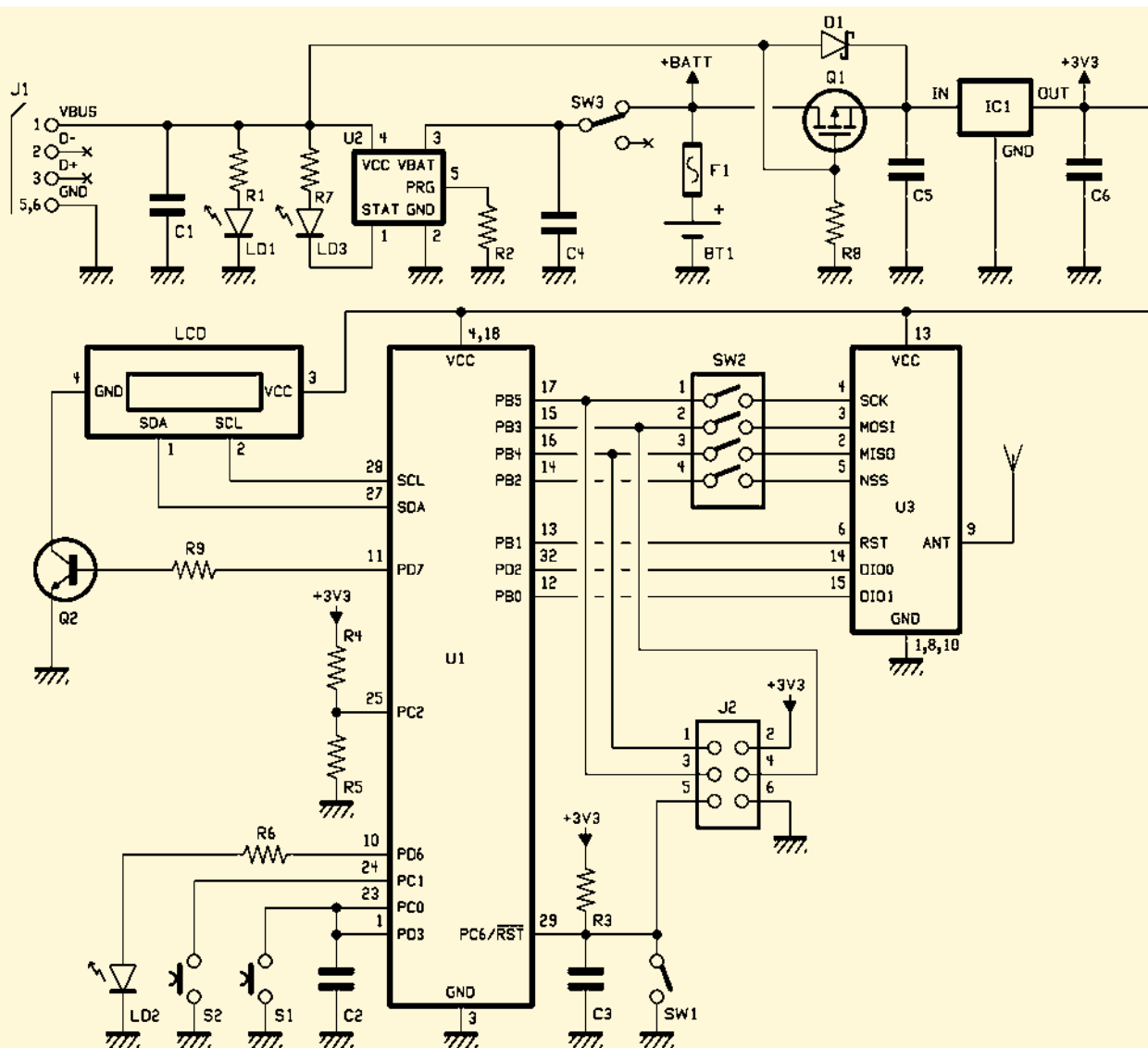


Fig.2
Il display indica
l'intensità luminosa
impostata.

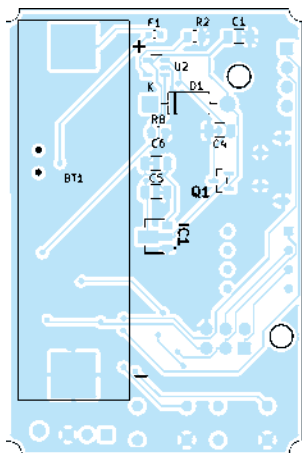
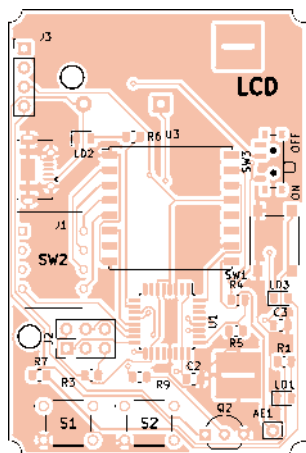
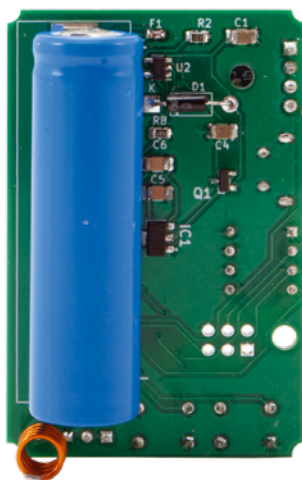
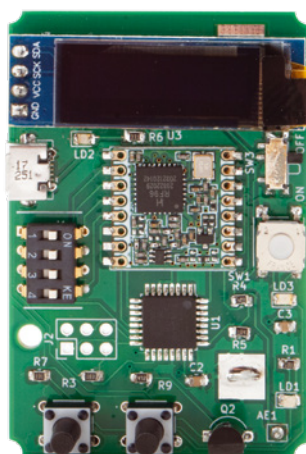


Per questo progetto abbiamo preferito utilizzare l'IDE di Microchip Studio, avvalendoci del programmatore MPLAB PICKIT 4 della Microchip per caricare lo sketch. Vedremo nell'apposito paragrafo come procedere alla programmazione. Il modulo LoRa, rappresentato dalla breakout RFM95W (U3), serve a trasmettere il dato che serve a variare il valore di modulazione PWM.

Oltre ai già citati microcontrollore Atmega328, modulo LoRa e display LCD non ci rimane altro che fare un'ultima panoramica di ciò che compone il trasmettitore, tracciando i punti più importanti che sono:

1. il dip switch SW2, collegato tra il BUS di comunicazione SPI del microcontrollore e il modulo LoRa. Oltre alla comunicazione tra il microcontrollore U1 e il modulo LoRa U3, la porta SPI viene utilizzata per il programmatore PICKIT4 collegato a J2, per cui per programmare il microcontrollore dobbiamo far sì che il BUS di comunicazione SPI sia isolato dal modulo LoRa. Mettendo in posizione OFF i quattro interruttori del dip switch SW2 si ottiene quanto premesso. Questo ci permette di programmare correttamente il microcontrollore senza rischiare eventuali errori nel caricamento dello sketch.

piano di montaggio TRASMETTITORE



Elenco Componenti:

R1, R6, R7: 330 ohm (0805)
R2, R3: 10 kohm (0805)
R4, R5, R8: 100 kohm (0805)
R9: 4,7 kohm (0805)
C1, C4: 4,7 μ F ceramico (1206)
C2, C3: 100 nF ceramico (0805)
C5, C6: 1 μ F ceramico (1206)
D1: 1N5819
Q1: DMP2160
Q2: BC547
U1: ATmega328P-A (MF1709)
U2: MCP73831
U3: RFM95W-868S2
IC1: AP7215-33YG-13
LD1: LED verde (0805)
LD2, LD3: LED rosso (0805)
S1, S2: Microswitch
SW1: Microswitch
SW2: Dip-switch 4 vie
SW3: Deviatore a slitta
LCD: Display OLED 0,91"
F1: Fusibile 500 mA (0603)
BT1: Batteria ICR14500 3.6V/600mA

Varie:

- Connettore microUSB
- Circuito stampato S1709 (59x40 mm)

- Ovviamente questa operazione è richiesta soltanto durante la fase di download dello sketch;
2. i due pulsanti S1 e S2 necessari per comandare l'aumento (S1) o la diminuzione (S2) della modulazione PWM, che, per comodità, chiameremo *power*. Questo si traduce nella trasmissione del dato che permetterà al ricevitore di gestire l'intensità luminosa dei LED;
 3. l'antenna (AE1): lavorando in ambienti interni non è necessario munirsi di un'antenna professionale. Andrebbe usata un'antenna tarata sulla lunghezza d'onda di 868 MHz (frequenza utilizzata in Europa) ma basta usare un semplice filo di rame lungo circa 8.6 cm, ovvero $\frac{1}{4}$ della lunghezza d'onda (34,56 cm/4);
 4. il LED LD2 che ci permette di sapere quando la batteria è scarica (3.35V). Il valore di tensione della batteria lo ricaviamo tramite i due resistori R4 e R5 collegati all'ingresso PC2 del microcontrollore.

Per preservare la carica della batteria e avere la maggiore autonomia del trasmettitore procederemo come segue:

- manterremo acceso il display il minor tempo possibile, ma sufficiente per la visualizzazione dell'incremento e del decremento del valore di "power". Per far ciò il display rimarrà acceso per 26 secondi (più che sufficiente per far ciò che serve). Raggiunto il tempo prefissato viene tolta tensione al display tramite Q2;
- un altro aiuto per risparmiare corrente ci viene dato da una funzione del modulo LoRa chiamata "sleep". Questa funzione ci permette di ridurre al minimo la corrente di assorbimento dello stesso.

A questo punto, da un consumo a pieno regime, che si aggira sui 15mA, utilizzando le due funzioni sopra elencate passiamo ad un consumo che si aggira sui 3mA.

- Il mini-pulsante S1: oltre ad avere la funzione di incrementare il dato denominato *power* permette di riattivare il display OLED e il modulo LoRa dalla fase di "sleep". Basta premerlo e rilasciarlo.

IL RICEVITORE

Considerando che la programmazione del microcontrollore Atmega328 (U1), come anche la funzione dei dip-switch (SW1) presenti nello schema del ricevitore (**Fig. 3**), avviene nello stesso modo del trasmettitore.

Siccome la modulazione PWM è la migliore soluzione per un corretto controllo dell'intensità luminosa dei LED, entreremo nei particolari del ricevitore tenendo a mente i valori sotto riportati, che sono:

- Frequenza PWM 120 Hz
- Corrente massima dei LED 1,3A
- Tensione d'ingresso 9/12V (l'alimentatore utilizzato lo potete trovare sul nostro sito con codice: 8220-PSS6E0920)

Come potete vedere dallo schema elettrico, l'alimentatore si compone di tre stadi: Il primo stadio ci fornisce l'alimentazione di ingresso, tramite J2, che serve ad alimentare i LED collegati a J3. Tenendo presente che la corrente massima che possiamo erogare è di 1,3A, in base alla corrente di controllo dei LED utilizzati (esempio: 80mA) potremo collegare in parallelo al massimo sedici LED; il secondo stadio tratta l'alimentazione dei 5V, fornita dallo stabilizzatore di tensione U3: ci permette di alimentare lo stadio di controllo della modulazione PWM, composto da Q2, Q3, Q4 ecc. (approfondiremo più avanti la sua funzione); per ultimo, lo stabilizzatore 3.3V che, come trattato in precedenza, serve per alimentare stabilmente U1, U2 e IC1. La presenza di una EEPROM (IC1), ci permette di memorizzare l'ultimo dato *power* ricevuto, che serve al controllo della modulazione PWM. Per ottenere la stessa intensità luminosa che si aveva prima dello spegnimento del ricevitore, il dato *power* viene caricato alla riaccensione.

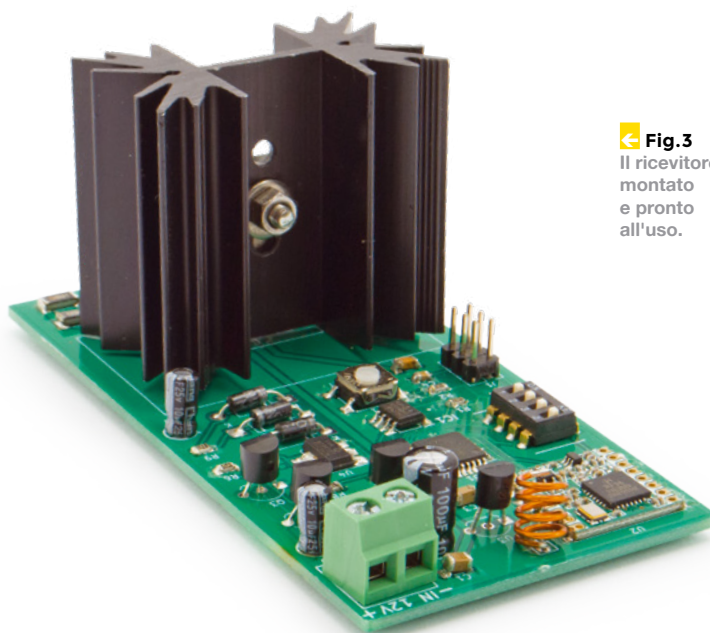
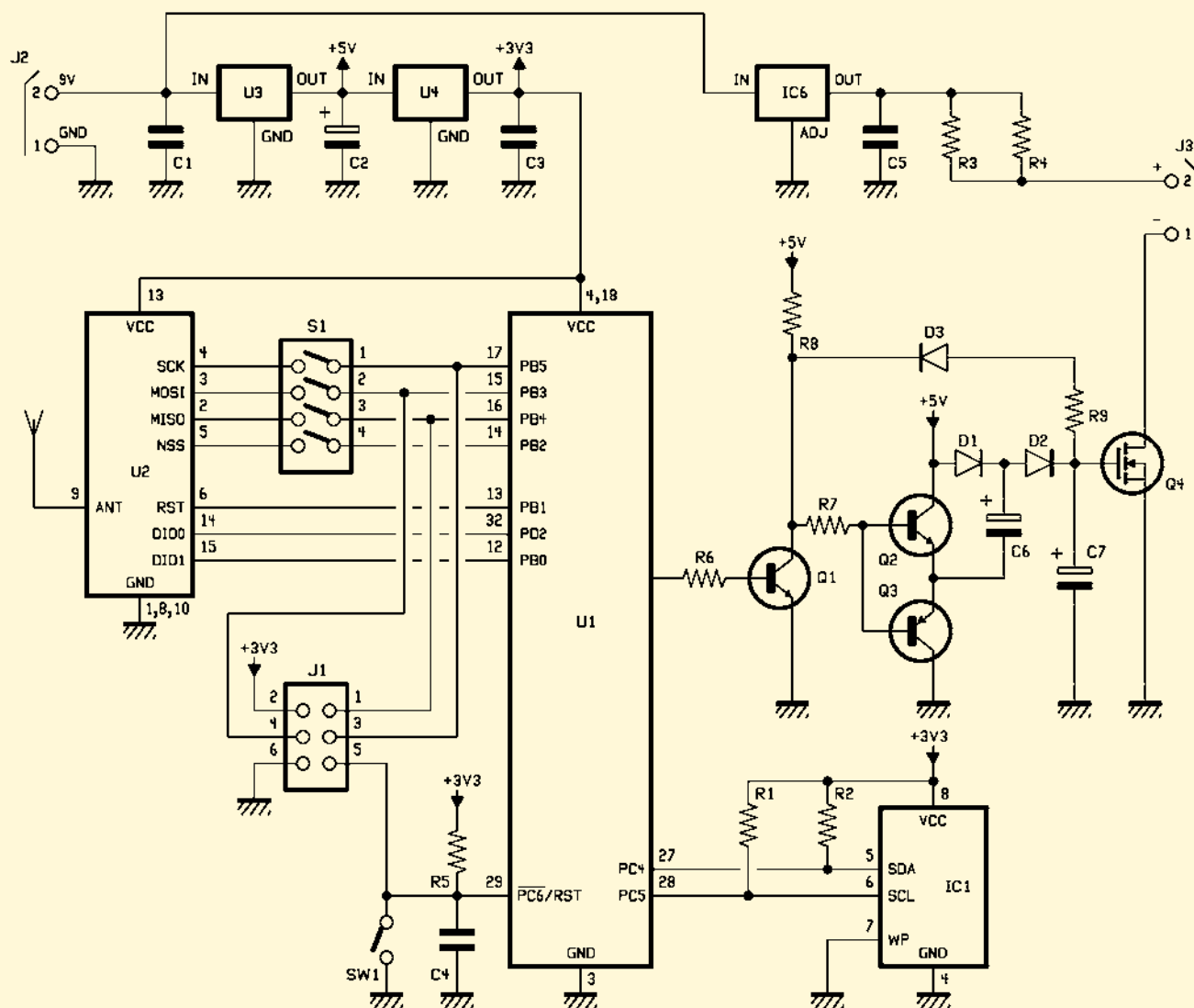


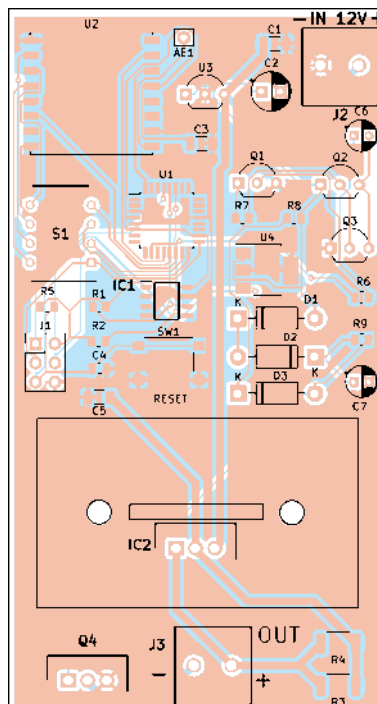
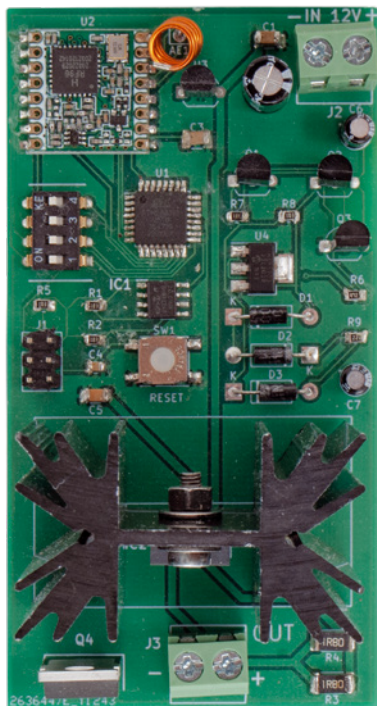
Fig. 3
Il ricevitore
montato
e pronto
all'uso.



La modulazione PWM viene generata dal microcontrollore Atmega328 (U1), utilizzando il TIMER 0 interno ad esso. All'uscita del pin PD5 troverete una forma d'onda con una frequenza di 120 Hz a 3.3V. Come già visto in precedenza, non possiamo controllare i LED direttamente tramite l'uscita PD5 del microcontrollore, ma abbiamo la necessità di un'interfaccia che possa controllare l'elevata corrente fornita ai LED e mantenere stabile la frequenza della modulazione PWM. Per far ciò ci siamo avvalsi dei componenti Q1, Q2, Q3 e Q4. La loro rete ci permette di controllare i LED, mantenendo una frequenza stabile e senza picchi spuri (vedi **Fig. 4**) che possono danneggiare il MOSFET (Q4) e il generatore di corrente, composto da IC2 che, in questo caso, viene utilizzato

come limitatore di corrente anziché di tensione. La corrente di controllo dei LED che, come ribadito, è di 1,3A, viene determinata dalle due resistenze R3 e R4. Abbiamo preferito mettere in parallelo due resistenze anziché utilizzarne una, in quanto alcuni valori bassi di resistenze SMD (Surface-mount Device), con potenze superiori ai 2W, sono difficili da reperire, per cui qualora vogliate cambiare il valore della corrente dei LED, avendo due resistenze in parallelo, avete maggiore flessibilità nella scelta. Uno sguardo particolare va rivolto al nostro limitatore di corrente IC2 (LM338): dotato di dissipatore, permettere di smaltire il calore in eccesso rispetto alla massima potenza dello stesso (1W).

piano di montaggio RICEVITORE



Elenco Componenti:

R1, R2, R5, R8: 10 kohm (0805)
 R3, R4: 1,8 ohm 2 W (2010)
 R6: 4,7 kohm (0805)
 R7: 100 ohm (0805)
 R9: 220 ohm (0805)
 C1, C3: 4,7 μ F ceramico (1206)
 C2: 100 μ F 25 VL elettrolitico ($\varnothing$ 5 mm)
 C4: 100 nF ceramico
 C5: 1 μ F ceramico (1206)
 C6, C7: 10 μ F 25 VL elettrolitico ($\varnothing$ 4 mm)
 D1, D2, D3 1N4007
 Q1, Q2: BC547
 Q3: BC557
 Q4: IRF540
 SW1: Microswitch
 S1 Dip-Switch 4 vie
 U1: ATmega328P-A (MF1710)
 U2: RFM95W-868S2
 U3: 78L05
 U4: MCP1703A-3002
 IC1: 24LC16B-I/SN
 IC2: LM338T_NOPB

Varie:

- Morsetto 2 poli passo 5,08 mm (2 pz.)
- Pin strip 2x3 vie Maschio passo 2,54 mm
- Circuito stampato S1710 (93x50 mm)

Siccome la tensione di ingresso può variare da un valore di 9V a 12V, nella situazione peggiore (12V) abbiamo una potenza dissipata pari a 8.6W, per cui è assolutamente necessario applicare un dissipatore.

Non è compito di questo articolo approfondire come calcolare il tipo di dissipatore da scegliere, ma vi basti pensare che per mantenere al di sotto dei 125°C la temperatura operativa massima della giunzione di IC2 (T_j), è necessario collegarlo ad un dissipatore di raffreddamento pari ad un valore di almeno 6°C/W o minore (minore è il valore, maggiore è la temperatura dissipata).

IDE MICROCHIP STUDIO

Come accennato in precedenza i programmi per il modulo trasmettitore e ricevitore (rispettivamente LoRaSender e LoRaReceiver), sono stati sviluppati grazie all'IDE di Arduino ma utilizzeremo l'IDE di Microchip Studio per la programmazione. Microchip Studio lavora come se fossimo in ambiente di sviluppo Arduino.

Per programmare il microcontrollore del trasmettitore e del ricevitore si potrebbe utilizzare la scheda di Arduino UNO, seguendo il tutorial che troverete al seguente indirizzo: <https://docs.arduino.cc/built-in-examples/arduino-isp/ArduinoToBreadboard/>, ma, per quanto ci riguarda, improponibile per microcontrollori con piedinatura TQFP montati su scheda.

Molto più funzionale è utilizzare un programmatore come:

- ATMEL-ICE(AVR) – funzionale ma un po' costoso. Può essere utilizzato direttamente nell'IDE di Arduino e solo per microcontrollori con architettura AVR
- MPLAB PICKit4 – programmatore della Microchip decisamente meno costoso e, soprattutto, più versatile.

Utilizzando il programmatore PICKit4, Microchip Studio ci permette di programmare una vasta gamma di microcontrollori della famiglia Microchip e con architettura AVR e SAM.

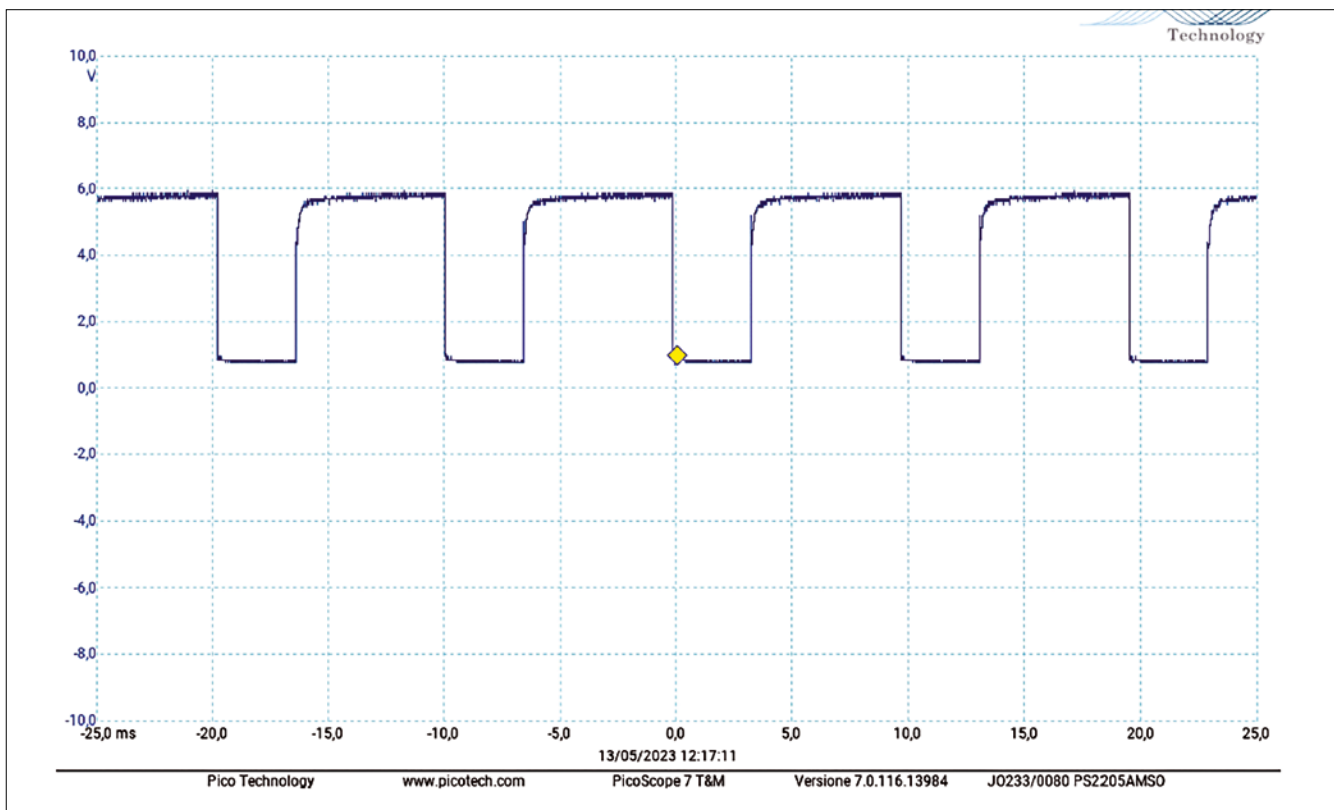


Fig.4
Modulazione
frequenza.

A questo punto installiamo il software Microchip Studio, che potete trovare nel sito Microchip al seguente indirizzo: <https://www.microchip.com/en-us/tools-resources/develop/microchip-studio>. Apritelo per caricare lo sketch creato nell'IDE di Arduino.

All'avvio vi troverete nella pagina di start (**Fig. 5**):

- cliccate su: "New Project" e vi comparirà la finestra come in **Fig. 6**;
- cliccate su: Create project from Arduino sketch;
- vi compariranno alcuni campi da compilare:

-Name: immettete il titolo del file da creare (LoRaSender o LoRaReceiver).

Automaticamente verrà compilato con lo stesso titolo il campo "Solution name"

-Location: è l'indirizzo dove verrà salvato il file "Name"

-Create new solution: lasciate invariato questo campo in quanto si tratta di un nuovo file.

Cliccate sul tasto OK per andare avanti.

La prossima finestra che vi comparirà è mostrata in **Fig. 7**.

Fig.5
Pagina apertura
software Microchip
Studio.



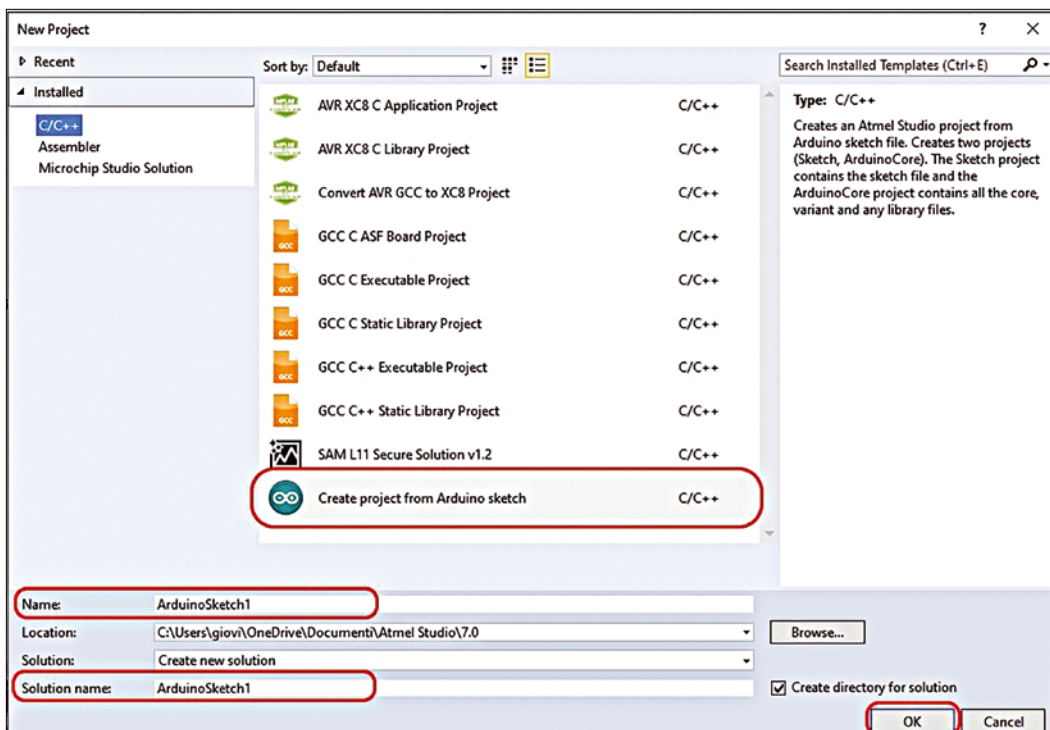


Fig. 6
Caricamento
Nuovo
Progetto.

Compilante anche in questo caso i campi come segue:

- Sketch file: cliccate sull'icona dei tre punti per accedere alla cartella dove avete memorizzato il file Sketch che abbiamo creato con l'IDE di Arduino.

Il file selezionato verrà utilizzato per creare il nostro programma nell'IDE di Microchip Studio;

- Arduino IDE patch: si tratta del percorso dove risiede il software di Arduino;

- Board: anche se poco importante, scegliamo la scheda Arduino Uno;
- Device: ovviamente scegliamo Atmega328.

Cliccando su "OK" verrà importato il file sketch di Arduino nell'IDE di Microchip Studio.

Come accennato in precedenza, l'ambiente di Microchip Studio ha le stesse funzioni dell'IDE di Arduino, ma con maggiori peculiarità, che vi invitiamo ad approfondire quando avremo terminato

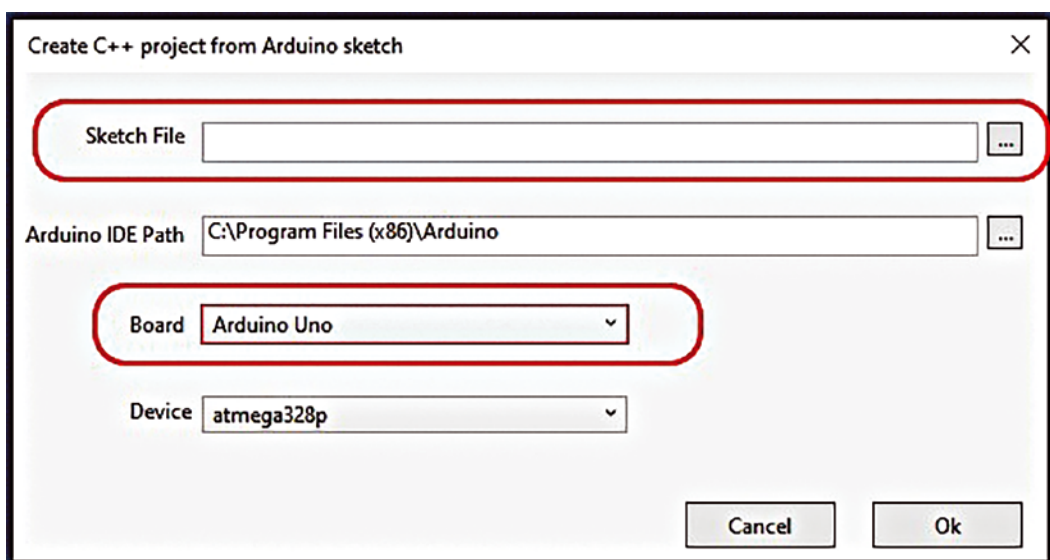


Fig. 7
Caricamen-
to sketch
Arduino

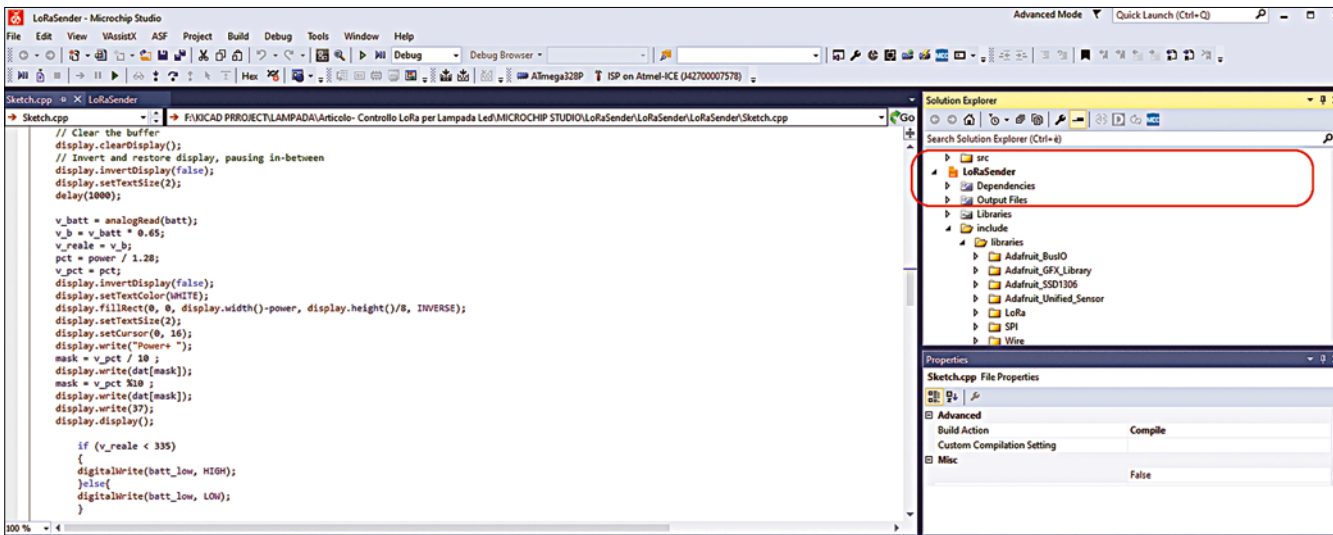


Fig. 8
Caricamento
librerie.

la verifica della funzionalità del file sketch, salvato nel nuovo ambiente di sviluppo.

Manca un ultimo passaggio affinché il nostro sketch funzioni: dobbiamo caricare le librerie necessarie.

Per farlo, cliccate con il tasto destro del mouse sulla finestra evidenziata in rosso (vedi **Fig. 8**); si aprirà un menu a tendina in cui troverete l'opzione "Add Arduino Library". Selezionatela, quindi scegliete dalla lista le librerie richieste per il vostro progetto.

Per LoRaSender, le librerie da selezionare sono:

- Adafruit_Sensor.h
- Adafruit_SSD1306.h
- LoRa.h

Per LoRaReceiver, selezionate:

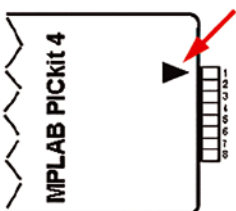
- Wire.h>
- SPI.h>
- LoRa.h>

NOTA

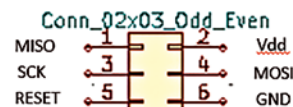
Per far sì che le librerie vi compaiano nella lista sopra citata dovete averle installate tramite le funzioni di caricamento librerie nell'IDE di Arduino. A questo punto abbiamo completato tutte le operazioni che ci permettono di caricare lo sketch nel nostro microcontrollore Atmega328.

Fig. 9
Collegamento
MPLAB Pickit 4.

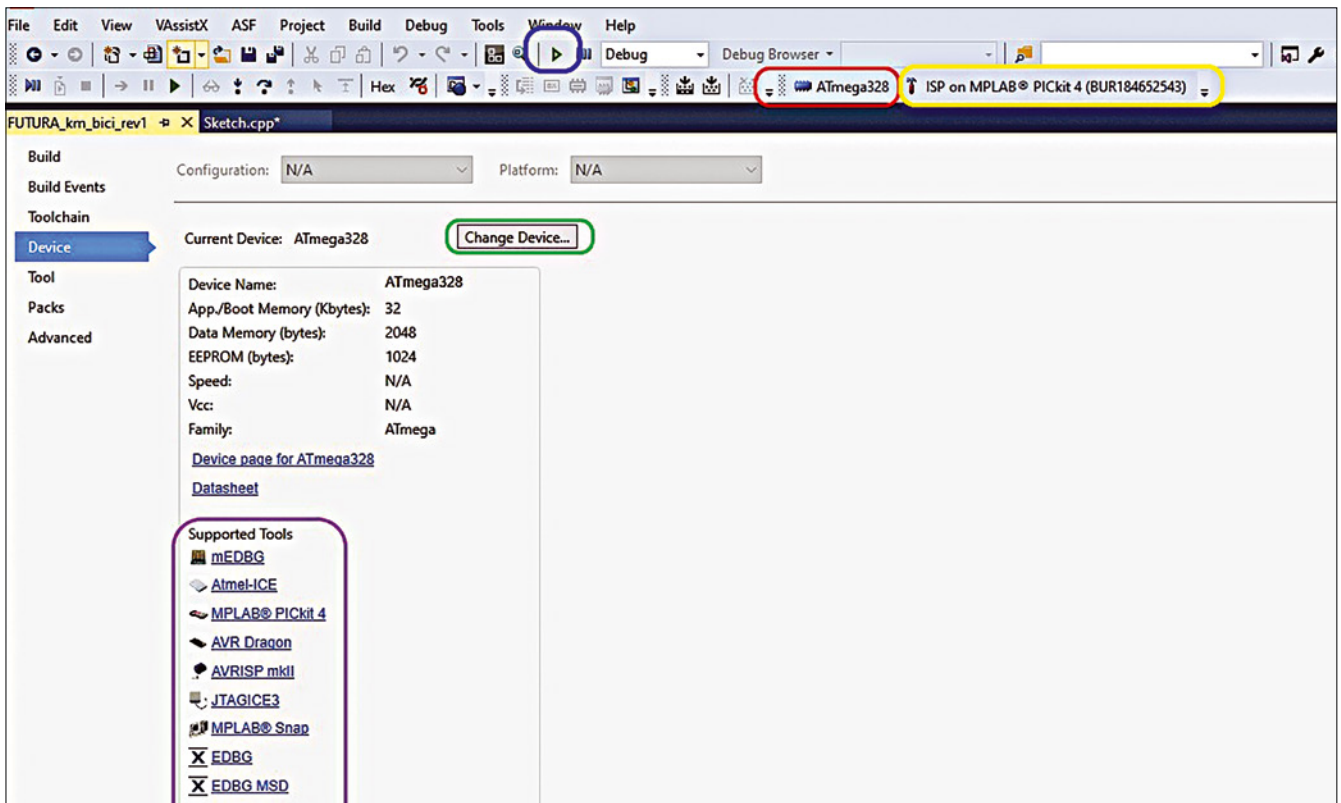
- Wire.h
- SPI.h
- Adafruit_GFX.h



MPLAB PICKIT 4 Pin Description*
Pin 1 = Unused for AVR
Pin 2 = Vdd
Pin 3 = GND
Pin 4 = MISO
Pin 5 = SCK
Pin 6 = RESET
Pin 7 = MOSI
Pin 8 = Unused for AVR



MPLAB PICKIT 4			DEBUG									
Connector	Pin #	Pin Name	ICSP (MCHP)	MIPS EJTAG	CORTEX SWD	AVR JTAG	AVR ISP(80W)	UPDI	UPDI	AV	DWIRE)	TPI
	1	TVPP	MCLR	MCLR	MCLR							
	2	TVDD	VDD	VIO_REF	VTG	VTG	VTG	VTG	VTG	VTG	VTG	VTG
	3	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND
	4	PGD	DAT	TDO	SWO	TDO	MISO	DAT	DAT	DATA		DAT
	5	PGC	CLK	TCK	SWCLK	TCK	SCK					CLK
	6	TAUX	AUX			RESET	RESET		CLK		dW	RST
	7	TTDI		TDI		TDI	MOSI					
	8	TTMS		TMS	SWDIO	TMS						



Per caricare il firmware ci serviamo del programmatore MPLAB PICKit4.

CARICAMENTO DEL FIRMWARE

Nelle due schede è presente un connettore a 6 PIN (J2 per il trasmettitore e J1 per il ricevitore) che utilizzeremo in accoppiata con il nostro programmatore MPLAB PICKit4 per caricare il firmware nel microcontrollore, collegando i pin come illustrato in **Fig. 9**.

Vi ricordiamo che per caricare il firmware nel microcontrollore dovete mettere in OFF il dip-switch chiamato SW2 nel board del trasmettitore e SW1 nel board del ricevitore.

Collegato l'MPLAB PICKit4 al microcontrollore procediamo come segue:

- selezioniamo il tipo di microcontrollore utilizzando cliccando sull'icona posta nella barra degli strumenti, e raffigurata da un integrato cerchiato in rosso (vedi **Fig. 10**);
- si aprirà una pagina che vi permette di scegliere il tipo di microcontrollore che stiamo utilizzando. Qualora non sia già selezionato cliccate sull'icona denominata "change device" cerchiata in verde;
- si aprirà un menu a tendina con una serie di

microcontrollori supportati da Microchip Studio; scegliete il microcontrollore Atmega328 e cliccate su "OK".

Nella stessa pagina viene riportata una lista di "programmatori" (vedi lista cerchiata in viola) supportati chiamata "Supported Tools". Cliccando sul programmatore di vostro interesse verrete indirizzati nella pagina WEB, dove potrete trovare maggiori dettagli e la documentazione a esso correlata.

Collegate il programmatore tramite la porta seriale mini USB e, se correttamente riconosciuto, selezionatelo cliccando sull'icona cerchiata in giallo (preventivamente aperta tramite il simbolo del martello). A questo punto abbiamo selezionato gli strumenti che ci servono per caricare il nostro firmware. Ora basta cliccare sull'icona rappresentata da una freccia verde (cerchiata in blu). Se avete selezionato correttamente tutti gli strumenti, inizierà il download del firmware nel microcontrollore.

Al termine del download, nella finestra di "output" in basso, vi dovete trovare la scritta "Build succeeded": significa che il download è andato a buon fine.

Fig. 10
Selezione device.

▼ Listato 1

```
//routine di accensione display *****
if (count == 100){
  digitalWrite(lcd_off, HIGH);
  delay(1000);
  //SSD1306_SWITCHCAPVCC = generate display voltage from 3.3V internally
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
  pct = power / 1.28;
  v_pct = pct;
  display.invertDisplay(false);
  display.setTextColor(WHITE);
  display.fillRect(0, 0, display.width()-power, display.height()/8, INVERSE);
  display.setTextSize(2);
  display.setCursor(0, 16);
  display.write("Power+ ");
  mask = v_pct / 10 ;
  display.write(dat[mask]);
  mask = v_pct %10 ;
  display.write(dat[mask]);
  display.write(37);
  display.display();
  count = 0;
  TIMSK1 = 0b00000001; //Mask INT OCF1A abilitato
  TCNT1 = 0;
}
```

SKETCH LORASENDER

Prima di analizzare lo sketch (scaricabile gratuitamente dalla pagina di presentazione di questa rivista) va fatta una piccola precisazione: nello sketch vedrete spesso comparire una funzione chiamata *power*, non è altro che il dato trasmesso per controllare la modulazione della larghezza di impulso (PWM) che, tradotto, determina l'intensità luminosa di LED.

La parte di SETUP dello sketch LoRaSender non viene trattata in quanto esiste già una breve spiegazione a fianco delle funzioni; inoltre, alcune funzioni sono parte della libreria ad essa correlata come, ad esempio, la gestione del display OLED SSD1306 (Adafruit_GFX.h, Adafruit_Sensor.h e Adafruit_SSD1306.h) e la trasmissione dei pacchetti LoRa (LoRa.h).

Come accennato, abbiamo un tempo limite di 26 secondi in cui il Display rimane acceso.

Per ottenere questo tempo utilizziamo il TIMER 1 a 16BIT interno al microcontrollore Atmega328. Raggiunto il valore di overflow del TIMER verrà richiamata la routine ISR (TIMER1_OVF_Vector). All'interno della routine, finché non saranno raggiunti una quantità di conteggi pari a 50 (cinquanta), definiti dalla variabile *-count-*, il display non verrà spento e il modulo LoRa non verrà messo in "sleep". La procedura di *wake_up* (risveglio) viene richiamata nel momento in cui si preme

e si rilascia il mini-pulsante S1 e, come potete immaginare, riaccenderà il display e "sveglierà" il modulo LoRa (**Listato 1**).

Il controllo che determina l'incremento o il decremento del dato chiamato *power* è dato dai mini-pulsanti S1 e S2. I comandi:

```
LoRa.beginPacket();
LoRa.print("A");
LoRa.print(power);
LoRa.endPacket();
```

ci permettono di trasmettere il dato *power* per tradurlo, dal ricevitore, in intensità luminosa. La lettera che anticipa il dato *power*, nel nostro caso la lettera "A", ci serve per trasmettere un dato che possa essere interpretato dal ricevitore come "start" di ricezione, permettendoci di filtrare eventuali interferenze radio che possano disturbare la ricezione del dato trasmesso.

Il calcolo dell'incremento o del decremento della barra, che indica la percentuale della variabile *power*, è:

```
power--;
pct = power / 1.28;
v_pct = pct;
```

Per incrementare la barra, e il valore in percen-



tuale ad esso correlato, dobbiamo sottrarre, anziché sommare, la variabile *power*. Questo è una conseguenza della funzione che controlla la lunghezza e l'altezza della barra, chiamata:

```
display.fillRect(0, 0, display.width()-counter, display.height()/8, INVERSE);
```

Tenendo conto che la matrice del display è un 128x32 pixel, la parte riguardante la lunghezza della barra viene determinata da *display.width()-counter* mentre lo spessore da *display.height()/8* per cui la lunghezza è determinata dalla sottrazione del valore di 128, mentre l'altezza della barra è determinata dal valore di 32 diviso per 8, per cui, nel nostro caso, risulta essere 4 pixel. Ultima annotazione riguarda il valore della variabile *power* espressa in percentuale: per far sì che i 128 pixel, inerenti alla larghezza del display, corrispondano al valore del 100% dobbiamo dividerlo per 1.28 (vedi **Fig. 5** cerchiato in arancio). Questo è il motivo per cui dobbiamo avere un primo valore float (pct - percentuale), per poi arrotondarlo in un valore intero (v_pct, valore percentuale). La parte riguardante il decremento del valore della barra è uguale alla funzione che abbiamo appena visto, con la sola differenza che anziché avere la variabile *power* sottratta (*power--*), avremo la stessa variabile incrementata (*power++*).

SKETCH LORARECEIVER

Il codice del ricevitore è molto semplice e breve

ma, per meglio comprenderne il senso lo abbiamo suddiviso in quattro macroaree.

Nel setup leggiamo l'ultimo dato di modulazione PWM ricevuto e memorizzato nella EEPROM IC, così da illuminare i LED con la stessa intensità impostata allo spegnimento.

Anche in questo caso la variabile che determina il dato di modulazione PWM è chiamata *power*. Molto importante è la parte:

```
TCCR0A = 0b00000000;
TCCR0B = 0b00000000;
TCCR0A = 0b10000011;
TCCR0B = 0b00000100;
OCR0A = 0;
```

in quanto determina come deve funzionare il TIMER 0, interno al microcontrollore Atmega328, per il controllo della modulazione PWM. Il dato della modulazione PWM viene memorizzato nella variabile *OCR0A* la quale determina il tempo che rimane in ON e il tempo che rimane in OFF la forma d'onda della modulazione di larghezza impulso PWM.

Onde evitare picchi incontrollabili di intensità luminosa dei LED, all'accensione del ricevitore abbiamo introdotto una piccola routine che viene eseguita solo quando si accende il ricevitore:

```
if (cnt == 0){
    Wire.beginTransaction(0x50);
    Wire.write(0x00);
    Wire.write(OCR0A = power);
```

Listato 2

```
int packetSize = LoRa.parsePacket(); //cerca di rilevare un pacchetto LoRa
if (packetSize) {                      //se viene ricevuto un pacchetto LoRa...

    while (LoRa.available()) {          //mentre è disponibile la connessione LoRa...
        lettera = (char)LoRa.read();
        power = (char)LoRa.read();
    }
    if (lettera == 65){                  //rilevazione lettera "A" = 65 decimale
        power = power * fatt;           //dato potenza illuminazione LED
        while (power != i){             //Trasmette il dato solo se c'è stata una variazione
            i = power;
            Wire.beginTransaction(0x50);
            Wire.write(0x00);
            Wire.write(OCR0A = power);
            Wire.endTransmission();
        }
    }
}
```

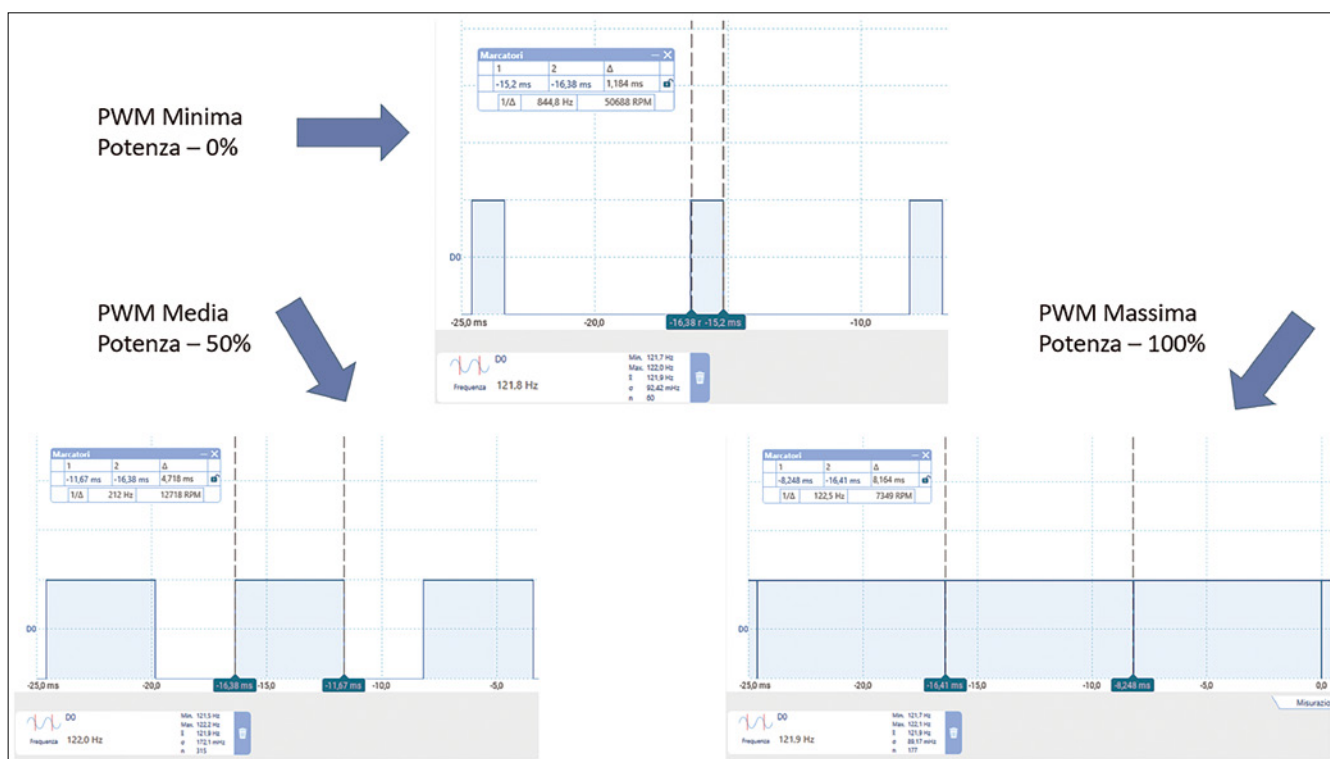


Fig. 11
modulazione della
larghezza di impulso
(PWM).

```
Wire.endTransmission();  
cnt++;
```

```
}
```

Nel **Listato 2** troviamo la parte che riguarda la ricezione del pacchetto LoRa, che determina il valore della variabile *power*. Siccome il valore di *power* va da 0 a 127, per avere un valore della variabile *OCROA* maggiore, ma non superiore a 255, (massimo valore di una variabile a 8 bit) abbiamo introdotto un fattore moltiplicativo pari a 1.71 (variabile "fatt = 1.71"), questo ci permette di avere una modulazione PWM con una forma d'onda minima, media e

massima come da **Fig. 11**. Siccome il modulo LoRa aggiorna continuamente il dato *power*, per ridurre al minimo le volte che memorizziamo il dato nella memoria EEPROM abbiamo introdotto un ciclo "while". In questo caso il dato della variabile viene memorizzato solo nel momento in cui il dato è diverso dal precedente, riducendo in questo modo al minimo il ciclo di scrittura/lettura della memoria.

CONCLUSIONI

Esiste una vasta gamma di LED di media potenza in commercio. Con questo articolo ci siamo soffermati su quelli che possono essere utilizzati per oggetti da illuminazione, dandovi modo di capire quale sia la scelta migliore per le vostre esigenze; inoltre abbiamo aggiunto la possibilità di comandare a distanza il vostro oggetto, utilizzando una tecnologia che non abbia limitazioni nella ricezione. Volutamente non abbiamo approfondito l'argomento sul funzionamento dei moduli LoRa in quanto già più volte affrontato con articoli mirati. A tale proposito vi consigliamo di visitare il nostro sito e cercare il corso LoRa pubblicato nel 2020 (vedi art. dal 246 al 251), per avere una maggiore panoramica sul funzionamento che potete ottenere da una semplice breakout come l'RFM95W. □



COSA OCCORRE ?

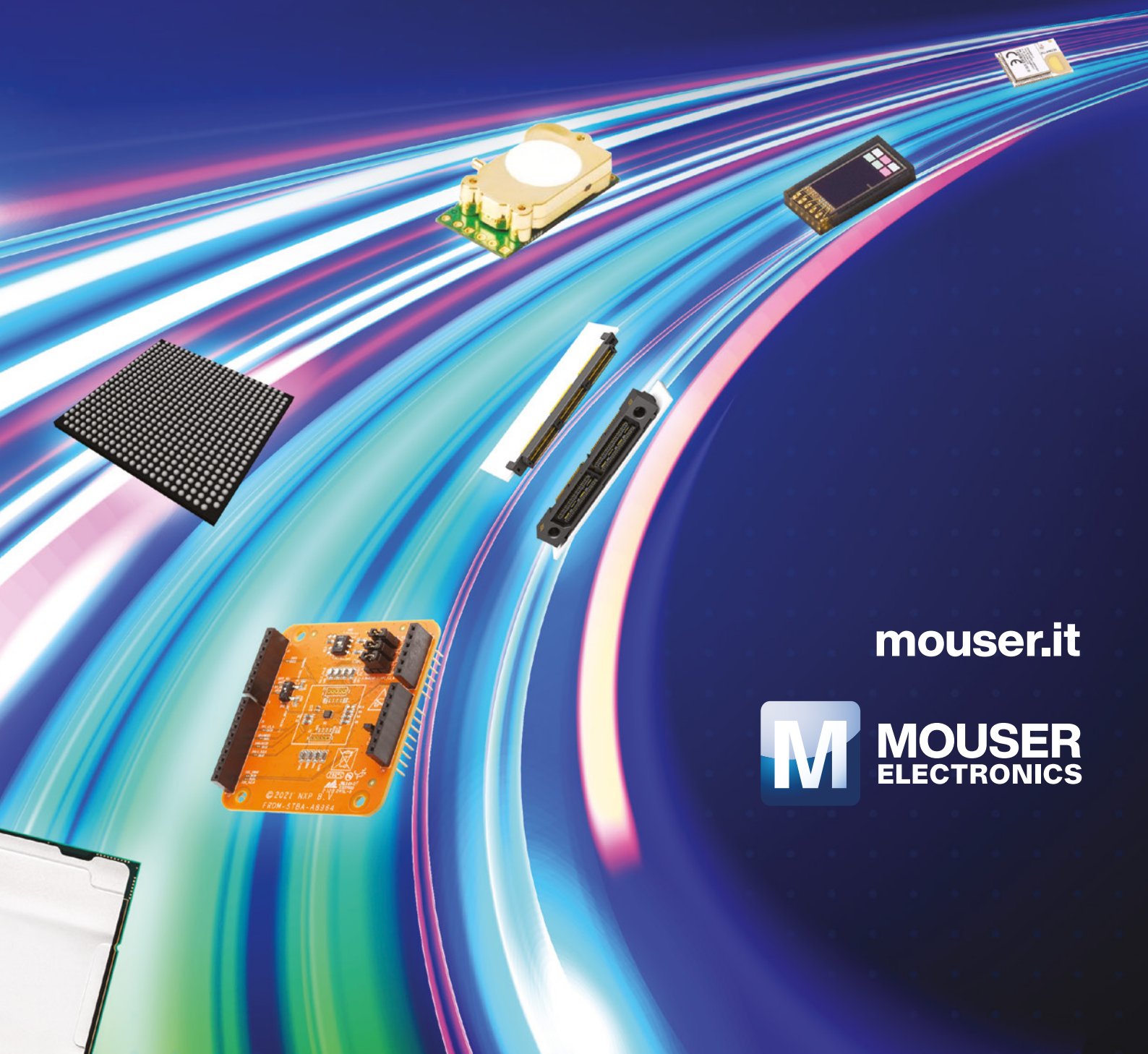
Codice	Descrizione	Prezzo € IVA inclusa
RFM95-868	Modulo RTX 868 MHz Long Range LoRa	13,50
I2COLED091	Display OLED I2C 0,91" 128x32 pixel	5,90
LIR14500FT	Batteria ricaricabile Li-ion 3,7V 750mAh	6,40

IL MATERIALE VA RICHIESTO A:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - www.futurashop.it

A tutta velocità

Affidatevi al leader nell'introduzione di nuovi prodotti™ per passare dall'idea al prototipo alla velocità della luce



[mouser.it](https://www.mouser.it)



**MOUSER
ELECTRONICS**

Monitor industriali a colori touch screen da 10"

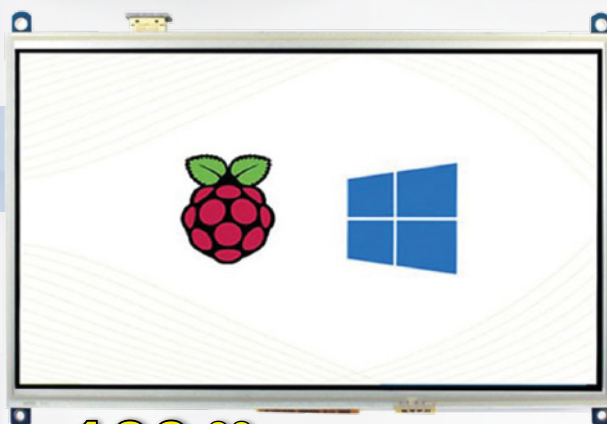


Cod. 10INCHHDMI2

€ 169,00

Display touch screen 10.1" capacitativo
con uscita audio – 1024×600 pixel

Connettività HDMI versatile, e tecnologia IPS per colori vividi. Touch a 10 punti per interazioni avanzate e incollaggio ottico per una maggiore leggibilità.
Compatibile con Raspberry Pi, PC e Game Console.
Alimentazione: 5 Vdc – 750~900 mA. Dimensioni: 236x144 mm.

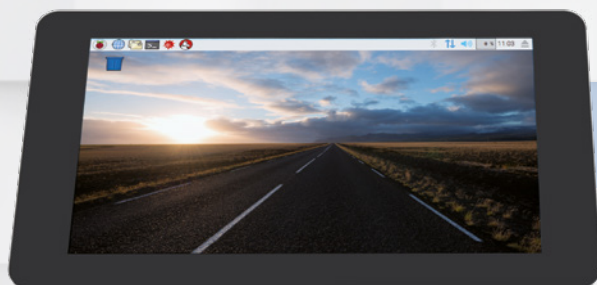


Cod. 10INCHRES

€ 129,00

Display touch screen 10" resistivo
1024×600 pixel con ingresso video HDMI

Particolarmente indicato per Raspberry Pi, può essere utilizzato anche come un monitor di un computer, proprio come qualsiasi altro schermo generico HDMI.
Alimentazione: 5 Vdc – 2 A. Dimensioni: 235x161x12 mm, peso: 495 grammi.



Cod. 10INCHHDMI

€ 185,00

Display a colori touch screen 10"
1280 x 800 pixel con case

Risoluzione di 1280x800 pixel. Particolarmente indicato per Raspberry Pi 3 / 2 / B + / A + / B. Il monitor viene fornito con una scheda di controllo dotata di ingresso HDMI, VGA e AV (CVBS).
Alimentazione: 5 Vdc – 300~750 mA. Peso: 1200 grammi.



Cod. 10INCH_HD

€ 198,00

Display touch screen 10.1" capacitativo
1920x1200 pixel con uscita audio e speaker

Display LCD capacitativo da 10,1 pollici con audio e 2 speaker integrati. La connettività HDMI consente il collegamento a una vasta gamma di dispositivi. Particolarmente indicato per schede Raspberry Pi, Jetson Nano, PC e console; è adatto per una vasta gamma di progetti e applicazioni, dallo sviluppo embedded o come display di alta qualità per il PC.
Alimentazione: 12 Vdc – 2A. Peso: 670 grammi.

Richiedi la quotazione per 5/10/20.... pezzi

L'ELETTRONICA CON ARDUINO: IL CIRCUITO RC

di PAOLO GUIDI



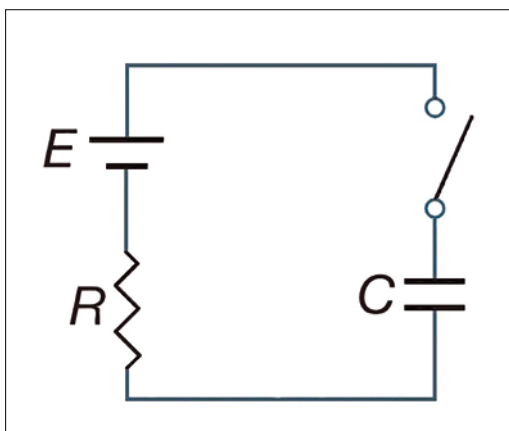
Lo studio di un circuito RC, fondamentale per chi si avvicina al mondo dell'elettronica, richiede una strumentazione adatta. Arduino semplifica il lavoro grazie al plotter seriale.

I

l circuito RC alimentato in continua (**Fig. 1**) è comunemente utilizzato sia in ambito elettrico che nella fisica. Analizzarne il comportamento significa andare oltre lo studio della carica e scarica del condensatore, che si carica chiudendo il tasto e si scarica aprendolo successivamente.

Nell'ambito della teoria dei sistemi rappresenta infatti l'esempio più classico di sistema del primo ordine al quale, dal punto di vista pratico, vengono fatte corrispondere diverse situazioni fisiche caratterizzate da un andamento di tipo esponenziale, crescente o decrescente. Una situazione fisica tipica che viene rappresentata con un andamento esponenziale, è ad esempio quella del riscalda-

→ **Fig. 1**
Circuito RC
alimentato
in continua.



mento e del raffreddamento di un forno che non avvengono immediatamente, con l'accensione o lo spegnimento del forno stesso, ma gradualmente e con modalità simili alla carica e alla scarica di un condensatore.

In **Fig. 2** vengono rappresentate le forme d'onda tipiche di un esponenziale crescente e di un esponenziale decrescente.

RICHIAMI TEORICI

Dal punto di vista matematico una funzione esponenziale presenta l'incognita (in questo caso la variabile tempo) all'esponente, essendo la base un numero reale diverso da uno; nel caso della carica e della scarica di un condensatore la base dell'esponenziale è il numero $e=2,718281\dots$ ovvero la costante tipica dei logaritmi neperiani. Per un condensatore (per ipotesi inizialmente scarico) che si sta caricando alla tensione massima E imposta dal generatore risulta la seguente

espressione analitica:

$$v(t) = E(1 - e^{-t/\tau})$$

Per un condensatore che ha raggiunto il valore massimo della tensione e che si sta scaricando per ritornare nello stato iniziale risulta la seguente espressione analitica:

$$v(t) = Ee^{-t/\tau}$$

Il prodotto RC viene definito costante di tempo ed indicato con τ (tau).

La costante di tempo rappresenta il tempo necessario per raggiungere il 63% del valore finale durante la fase di carica e il 37% del valore iniziale durante la fase di scarica; è quindi un indice della velocità con la quale si carica/scarica il condensatore.

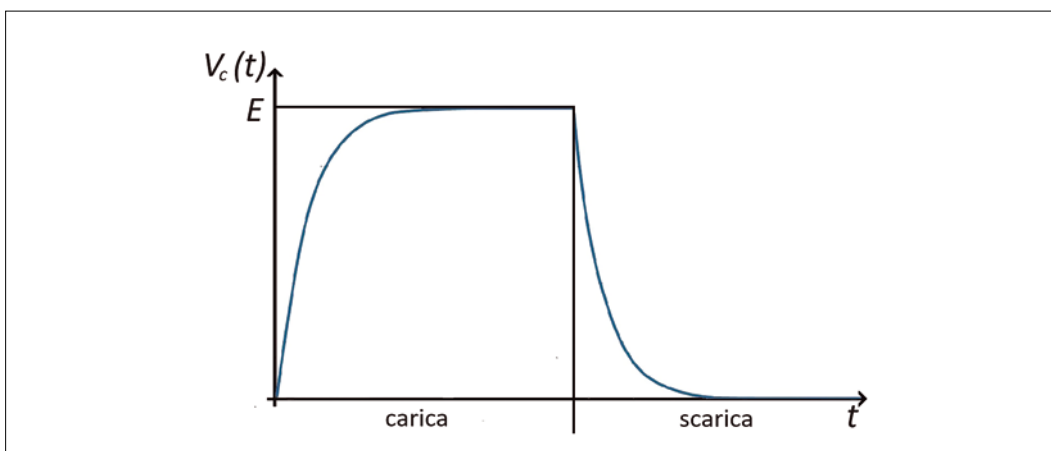
L'intervallo di tempo necessario per caricarsi o per scaricarsi completamente viene definito transitorio che per convenzione si considera esaurito al tempo $t=5\tau$. Il valore finale, o valore di regime, viene raggiunto al termine del transitorio. Le espressioni analitiche dei transienti di carica e scarica descritte in precedenza subiscono piccole modifiche se si considera che il condensatore sia inizialmente carico o parzialmente carico.

STUDIO DEL TRANSITORIO

Qui di seguito viene proposto lo studio del transitorio di un circuito RC senza ricorrere all'impiego della strumentazione tipica che è normalmente presente nei laboratori.

A livello hardware viene infatti fatto uso di una scheda Arduino e di una breadboard sulla quale viene montato il circuito.

→ **Fig. 2**
Esponenziale
crescente
e decrescente.



Come viene evidenziato in **Fig. 3**, la tensione che alimenta il circuito (5 V a livello alto e 0 V a livello basso) viene prelevata da un'uscita digitale, mentre la tensione da misurare (quella ai capi del condensatore) viene inviata a un ingresso analogico. Il polo negativo del condensatore elettrolitico è collegato al GND di Arduino.

Le forme d'onda sono visualizzate sul plotter seriale.

Lo schema elettrico complessivo utilizzato per questa esperienza viene riportato in **Fig. 4**. In questo caso, la resistenza è sostituita da un potenziometro da 10 k Ω , che permette di ottenere forme d'onda con diverse costanti di tempo. Il terminale centrale del potenziometro è collegato al positivo del condensatore.

I condensatori utilizzati sono elettrolitici, con una capacità più elevata; per aumentare ulteriormente la capacità, è possibile collegare più condensatori in parallelo (in questo esempio, si utilizzano due condensatori da 47 μ F e 100 μ F per ottenere così una capacità di circa 147 μ F).

Con questi componenti il valore massimo di costante di tempo ottenibile, tenendo presente la tolleranza dei medesimi, è di circa 1,5 s.

Lo sketch riportato nel **Listato 1** consente la visualizzazione di un ciclo completo di carica e scarica di un condensatore.

Nel *void setup* si avvia una fase iniziale di 5 s in cui il condensatore viene fatto scaricare completamente; tale fase viene segnalata con l'accensione

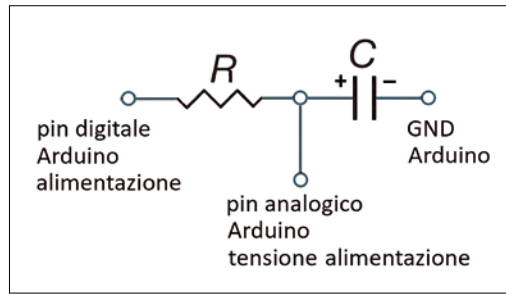


Fig. 3
Collegamento con i pin della scheda Arduino.

del LED di scarica. Nel *void loop* si distinguono una fase di carica seguita da una fase di scarica caratterizzate ciascuna da 250 letture di tensione prelevate ad intervalli di 10 ms (*ciclo for* con variabile contatore).

La fase di carica ha inizio portando a livello alto l'uscita D4, la fase di scarica portando la medesima a livello basso.

Vengono proposti tre casi in cui il potenziometro, regolato su valori differenti, permette di ottenere tre situazioni distinte, precisamente:

- primo caso: il potenziometro viene tarato sul valore minimo di resistenza ottenendo una costante di tempo tale che la carica e la scarica del condensatore non risultano apprezzabili;
- secondo caso: il potenziometro viene tarato su un valore di resistenza intermedio ottenendo una costante di tempo tale da consentire la carica e la scarica completa del condensatore;
- terzo caso: il potenziometro viene tarato sul

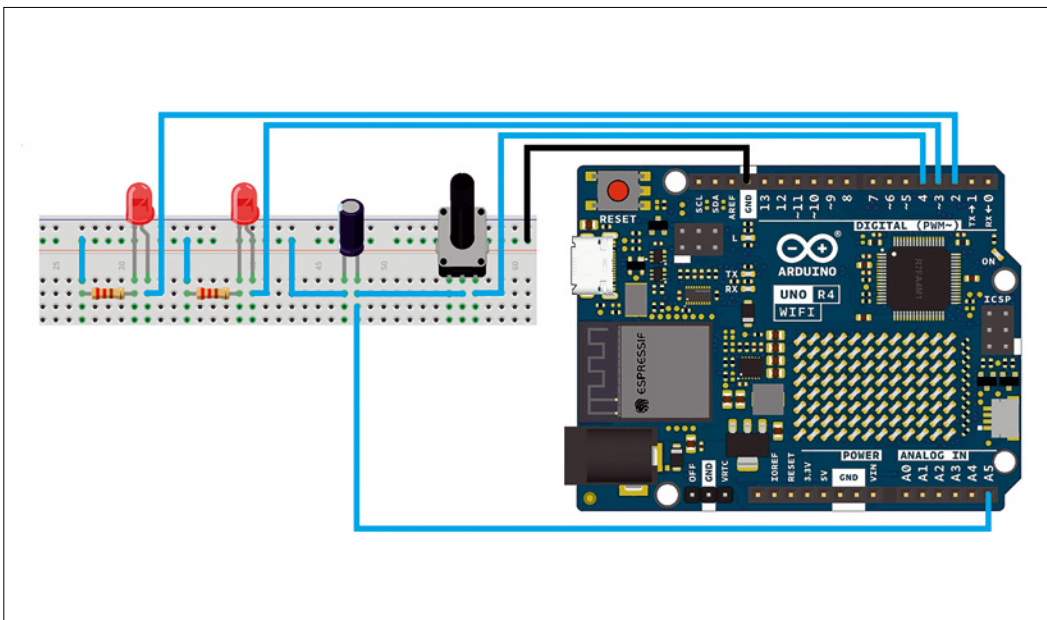


Fig. 4
Schema elettrico.

Listato 1

```
int i; // variabile contatore ciclo lettura
float v; // variabile tensione
void setup() {
  Serial.begin(9600); // imposta la velocità di trasmissione
  pinMode(2, OUTPUT); // configurazione delle uscite
  pinMode(3, OUTPUT);
  pinMode(4, OUTPUT);
  digitalWrite(4, 0); // il condensatore si scarica
  digitalWrite(3, 1); // LED scarica acceso
  delay(5000); // tempo attesa scarica completa
}
void loop() {
  digitalWrite(4, 1); // il condensatore si carica
  digitalWrite(2, 1); // LED carica acceso
  digitalWrite(3, 0); // LED scarica spento
  for (i = 0; i < 250; i++) { // ciclo di lettura
    v = analogRead(A5); // lettura tensione condensatore
    v = v * 5.0 / 1023; // adattamento valore di tensione
    Serial.print(v); // visualizzazione tensione condensatore
    Serial.print(' ');
    Serial.println(5); // visualizzazione livello alto
    delay(10); // ritardo tra due letture
  }
  digitalWrite(4, 0); // il condensatore si scarica
  digitalWrite(2, 0); // LED carica spento
  digitalWrite(3, 1); // LED scarica acceso
  for (i = 250; i < 500; i++) { // ciclo lettura
    v = analogRead(A5); // lettura tensione condensatore
    v = v * 5.0 / 1023; // adattamento valore di tensione
    Serial.print(v); // visualizzazione tensione condensatore
    Serial.print(' ');
    Serial.println(0); // visualizzazione livello basso
    delay(10); // ritardo tra due letture
  }
  digitalWrite(3, 0); // LED scarica spento
  while (1) {} // fine acquisizione
}
```

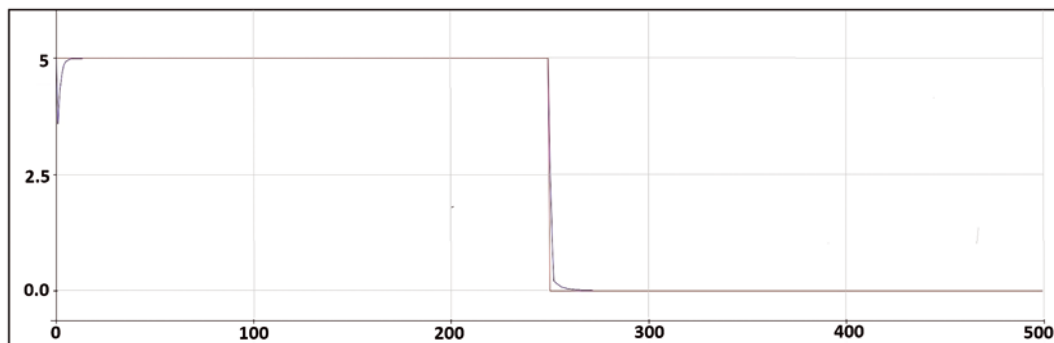
valore massimo di resistenza ottenendo una costante di tempo tale da non consentire la carica e la scarica completa del condensatore. Relativamente ai casi descritti nelle **Fig. 5, 6 e 7** vengono riprodotti i grafici della tensione ai capi del condensatore e delle tensioni di alimentazione visualizzati sul plotter seriale. Lo scorrere dei dati acquisiti può anche essere

visualizzato sul monitor seriale; la cadenza così veloce (10 ms tra un dato e quello successivo) non rende però agevole la lettura.

IMPIEGO DELLA MATRICE A LED DI R4 WI-FI

Se si utilizza la scheda R4-WIFI lo sketch precedente può essere modificato in modo tale da far accendere/spegnere gradualmente i LED della

Fig. 5
Grafico caso 1.



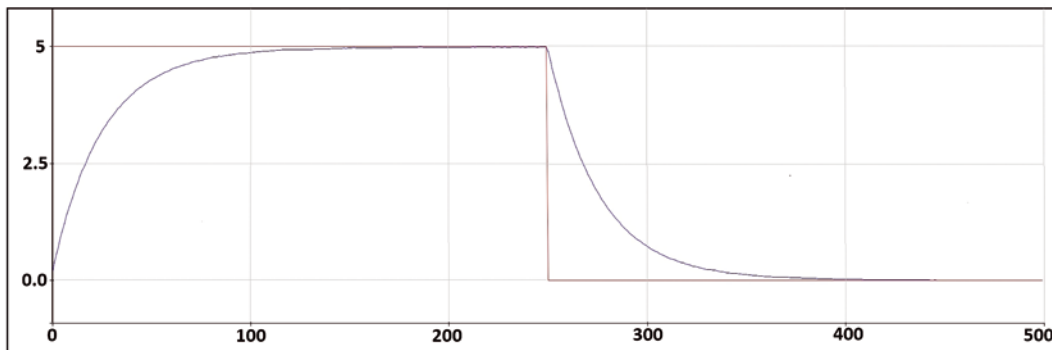


Fig. 6
Grafico
caso 2.

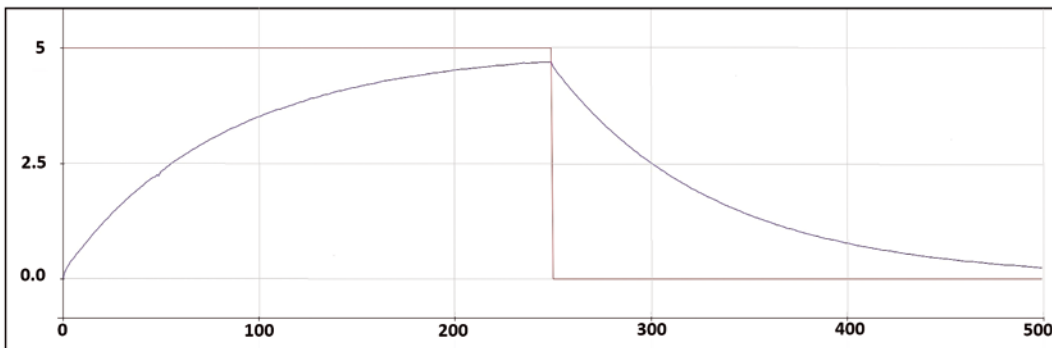


Fig. 7
Grafico
caso 3.

matrice in relazione allo stato di carica del condensatore; non vengono pertanto utilizzati i LED che segnalano le fasi di carica e scarica e nemmeno

vengono trasmessi i dati sul plotter o sul monitor seriale. Nell'esempio che segue si è scelto di mantenere sempre accese le prime due colonne (stato

Listato 2

```

#include "Arduino_LED_Matrix.h"           // include la libreria LED_Matrix.h

ArduinoLEDMatrix matrix;                  // crea l'oggetto matrix
int i;                                    // variabile contatore ciclo lettura
float v;                                  // variabile tensione

void setup() {
  matrix.begin();                          // inizializza l'oggetto matrix
  Serial.begin(9600);                      // imposta la velocità di trasmissione
  pinMode(4, OUTPUT);                     // configura l'uscita digitale alimentazione
  mat0();                                  // richiama matrice stato iniziale
  digitalWrite(4, 0);                      // il condensatore si scarica
  delay(5000);                             // tempo attesa scarica completa
}

void loop() {
  digitalWrite(4, 1);                      // il condensatore si carica
  for (i = 0; i < 250; i++) {              // ciclo di lettura
    v = analogRead(A5);                    // lettura tensione condensatore
    v = v * 5.0 / 1023;                    // adattamento valore di tensione
    delay(10);                             // ritardo tra due letture
    matC();                                 // richiama matrice fase di carica
  }
  digitalWrite(4, 0);                      // il condensatore si scarica
  for (i = 250; i < 500; i++) {            // ciclo lettura
    v = analogRead(A5);                    // lettura tensione condensatore
    v = v * 5.0 / 1023;                    // adattamento valore di tensione
    delay(10);                             // ritardo tra due letture
    matS();                                 // richiama matrice fase di scarica
  }
  while (1) {}                             // fine acquisizione
}

```

Listato 3

```
int i; // variabile contatore ciclo lettura
float v0, v, vct, vcf; // variabili tensione
float tao; // variabile costante di tempo
unsigned long t0, t; // variabile tempo iniziale e tempo corrente
void setup() {
    Serial.begin(9600); // imposta la velocità di trasmissione
    pinMode(4, OUTPUT); // configurazione uscita alimentazione
    Serial.println("scarica iniziale del condensatore");
    Serial.println("attendere 10 s.....");
    digitalWrite(4, 0); // il condensatore si scarica
    delay(10000); // tempo attesa scarica completa
    v0 = analogRead(A5); // lettura valore tensione iniziale
    v0 = v0 * 5.0 / 1023; // adattamento valore di tensione
    Serial.println("stato carica condensatore");
    Serial.print(v0, 3);
    Serial.println("V");
    Serial.println("inizio fase di carica");
    Serial.println("il circuito viene alimentato a 5 V");
}
void loop() {
    digitalWrite(4, 1); // il condensatore si carica
    t0 = millis(); // imposta il tempo iniziale
    t = millis() - t0;
    for (i = 0; i < 600; i++) { // ciclo di lettura
        t = millis() - t0; // valuta il tempo corrente
        v = analogRead(A5); // lettura tensione condensatore
        vct = v * 5.0 / 1023; // adattamento valore di tensione
        if (vct > 3.130 && vct < 3.170) { // confronto valore corrente-tensione di soglia
            tao = t;
        }
        if (i == 599) { // confronto valore corrente-tensione di soglia
            vcf = vct;
        }
        delay(10);
    }
    Serial.print("costante di tempo: ");
    Serial.print(tao / 1000, 3); // visualizza costante di tempo
    Serial.println("s");
    Serial.println("");
    Serial.print("valore finale alla fine del transitorio: ");
    Serial.print(vcf, 3); // visualizza valore finale
    while (1) {} // fine acquisizione
}
```

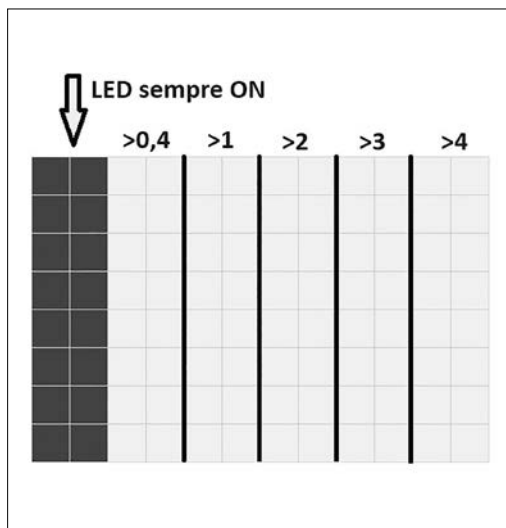


Fig. 8
Schema per l'accensione dei LED della matrice.

iniziale); durante la fase di carica si accenderanno a scatti di due le colonne successive fino a quando tutti i LED risulteranno accesi; durante la fase di scarica è previsto lo spegnimento dei LED in ordine inverso fino al ripristino dello stato iniziale. Per poter visualizzare i LED della matrice si deve includere la libreria *Arduino_LED_Matrix.h* e inizializzare l'oggetto denominato *matrix*. Nel corso del *void loop* vengono richiamati tre sottoprogrammi (*mat0*, *matS* e *matC*) che determinano l'accensione dei LED nello stato iniziale e nelle fasi di carica e scarica. All'interno del codice vi sono dei sottoprogrammi dove vengono impostate (serie di *if*) le soglie di tensione e la matrice corrispondente alla soglia raggiunta (il codice completo è scaricabile dal sito della rivista).

RESISTENZA [KΩ]	COSTANTE DI TEMPO [s]	VALORE FINALE [V]
9,5	1,479	4,917
6,3	0,975	4,985
4,5	0,692	4,995
2,7	0,409	5,000
1,3	0,189	5,000

Tabella 1
Risultati delle prove.

Dallo schema di **Fig. 8** risulta in particolare che:

- per $v > 0,4$ V si accendono i LED delle colonne 3 e 4;
- per $v > 1$ V si accendono quelli delle colonne 5 e 6;
- per $v > 2$ V si accendono quelli delle colonne 7 e 8;
- per $v > 3$ V si accendono quelli delle colonne 9 e 10;
- per $v > 4$ V si accendono quelli delle colonne 11 e 12.

Durante la fase di scarica i LED si spengono in ordine decrescente di soglia. La parte principale dello sketch viene proposto nel **Listato 2**.

CALCOLO DELLA COSTANTE DI TEMPO

Scopo di questo terzo esempio non è, come nei due casi precedenti, quello di monitorare la carica e la scarica di un condensatore ma il calcolo della costante di tempo e il valore finale al termine del transitorio di carica.

Dopo una fase iniziale in cui il condensatore viene fatto scaricare completamente, il circuito viene alimentato dando inizio alla fase di carica.

La costante di tempo si determina individuando l'istante di tempo in cui la tensione ai capi del condensatore raggiunge un valore pari a circa 3,15 V che rappresenta il 63% del valore finale.

Per determinare l'istante di tempo si deve utilizzare l'istruzione `millis()` memorizzando in una variabile l'istante in cui viene alimentato il circuito (*valore iniziale*) e valutando ad ogni ciclo di misura il *tempo corrente* che si ottiene sottraendo a `millis()` il valore iniziale. Poiché si è deciso di effettuare un ciclo di 600 misure a distanza di 10 ms l'una dall'altra, il valore finale al termine del transitorio viene calcolato quando viene effettuata l'ultima misura ovvero quando la variabile contatore del ciclo ha raggiunto il valore 599.

Si tenga presente che il numero di misure da effettuare e la distanza tra una misura e l'altra sono parametri da valutare in relazione ai valori di resistenza e capacità che si vogliono utilizzare.

Lo sketch completo è disponibile nel **Listato 3**.

Sono state effettuate cinque prove al variare del valore di resistenza del potenziometro che hanno

dato i risultati riportati in **Tabella 1**.

I risultati della prima prova visualizzati sul monitor seriale vengono riportati in **Fig. 9**.

CONCLUSIONI

L'utilizzo di Arduino per studiare il comportamento di un circuito RC alimentato in continua si rivela un approccio pratico ed efficace che supera le limitazioni legate all'assenza di strumentazione avanzata come oscilloscopi e generatori di funzioni. Grazie all'impiego del plotter seriale e delle funzioni di Arduino è possibile monitorare e visualizzare le forme d'onda caratteristiche del circuito, facilitando l'apprendimento dei principi fondamentali di elettronica. Inoltre, l'esperimento proposto dimostra come sia possibile calcolare con precisione la costante di tempo del circuito, rendendo Arduino uno strumento didattico versatile per chiunque si avvicini al mondo dell'elettronica. ■

```

scarica iniziale del condensatore
attendere 10 s .....

stato carica condensatore
0.029 V

inizio fase di carica
il circuito viene alimentato a 5 V

costante di tempo: 1.489 s

```

Fig.9
Risultati della prima prova.

COSA OCCORRE ?		
Codice	Descrizione	Prezzo € IVA inclusa
ARDUINOUNOR4W	Arduino Uno R4 Wi-Fi	31,90
BB400	Breadboard da 400 contatti	4,00
IL MATERIALE VA RICHIESTO A:		
Futura Elettronica, Via Adige 11, 21013 Gallarate (VA) Tel: 0331-799775 - www.futurashop.it		



Lampade da tavolo per ogni esigenza!

A LED, con lente, con ricarica wireless...

Lampada LED con lente d'ingrandimento



€94,00

Cod. TP246

- » Lente d'ingrandimento: 8 diottrie
- » Alimentazione: 220 VAC – 240 VAC / 50 Hz
- » Consumo: 9 W
- » Flusso luminoso: 685 lumen
- » Temperatura colore: 6000-7000 K
- » Montaggio: morsetto metallico per il fissaggio al banco

Lampada da tavolo a LED con lente d'ingrandimento



€39,00

Cod. TP206

- » Lente d'ingrandimento: 3 + 12 diottrie
- » Ingrandimento: 1,75 x + 4 x
- » Alimentazione: 230 VAC / 50 Hz – 5,5 W
- » Flusso luminoso: 400 lumen
- » Temperatura colore: 6000-7000 K
- » Diametro lente: 85 mm
- » Lunghezza braccio di supporto: 250 mm

Lampada LED con lente d'ingrandimento



€72,00

Cod. TP197

- » Lente d'ingrandimento: 3 diottrie
- » Flusso luminoso: 650 lumen
- » Temperatura colore: 6000K-7000K
- » Alimentazione: 230 VAC / 50 H
- » Consumo: 9 W
- » Montaggio: morsetto metallico per il fissaggio al banco

DISPONIBILE ANCHE CON
FLUSSO LUMINOSO DI 700 LUMEN
COD. TP193 - € 94,00

Lampada da tavolo a LED con lente d'ingrandimento



€49,50

Cod. TP248

- » Lente d'ingrandimento: 3 + 12 diottrie
- » Ingrandimento: 1,75 x + 4 x
- » Alimentazione: 230 VAC / 50 Hz – 6,5 W
- » Flusso luminoso: 520 lumen
- » Temperatura colore: 6500 K
- » Diametro lente: 90 mm
- » Lunghezza braccio di supporto: 290 mm

Lampada LED touch dimmerabile con ricarica wireless per smartphone



€34,00

Cod. EFL60

- » Temperatura colore: 2800 K – 3400 K – 4500 K – 5600 K
- » Flusso luminoso: 250 lumen
- » Alimentazione: 5 VDC / 2,5 A – a corredo cavo USB e adattatore da rete 220 VAC (opzionale)
- » Consumo: 5 watt
- » Ricarica wireless: 5 V / 1 A
- » Dimensioni (mm): 126x45x440

Lampada LED professionale con lente d'ingrandimento

- » Lente d'ingrandimento: 3 diottrie
- » Flusso luminoso: 850 lumen
- » Temperatura colore: 6500 K
- » Fissaggio: morsetto da tavolo
- » Lunghezza del braccio: 790 mm
- » Consumo: da 0,5 a 12 W
- » Alimentazione: 230 VAC / 50 Hz – 5,5 W
- » Montaggio: morsetto metallico per il fissaggio al banco



€134,00

Cod. TP243

Lampada professionale a LED dimmerabile - 880 lumen



€98,00

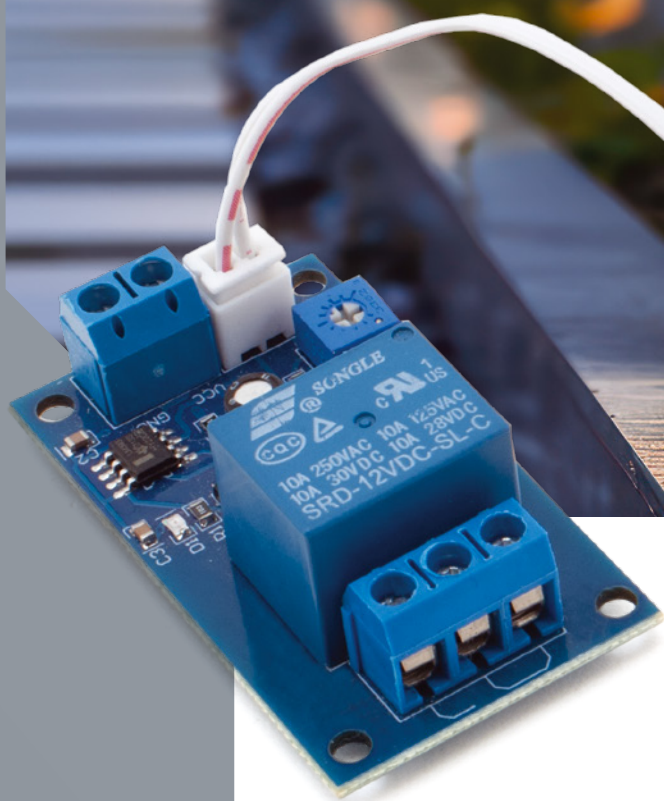
Cod. TP233

- » Alimentazione: 220 VAC – 240 VAC / 50 Hz
- » Consumo: da 1,5 a 14,5 W (dipende dalla luminosità)
- » Flusso luminoso: 880 lumen
- » Temperatura colore: 6000-7000 K
- » Montaggio: morsetto metallico per il fissaggio al banco
- » Dimensioni (mm): 450 x 80 x 400 (braccio)

Prezzi IVA inclusa.

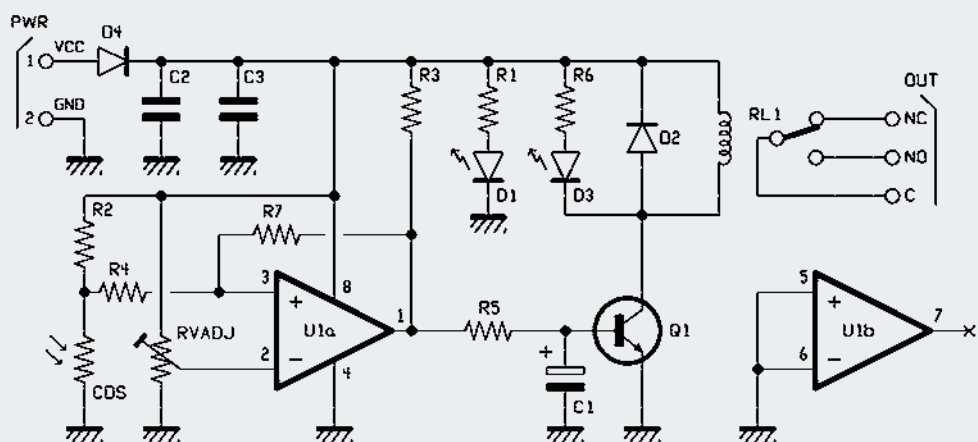
INTERRUTTORE CREPUSCOLARE

di DAVIDE SCULLINO



Comanda l'accensione di una o più lampade al calare della sera, spegnendole quando torna a fare giorno.

Il controllo automatico di un dispositivo elettrico, che si attiva in base alle condizioni di illuminazione di un ambiente, è una funzione ampiamente utilizzata e apprezzata da diversi decenni, in particolare da quando è stata resa possibile grazie all'elettronica. In passato, tale tecnologia veniva prevalentemente utilizzata per l'illuminazione pubblica (come per accendere automaticamente i lampioni di vie e piazze al calare della sera) o per le luci esterne delle abitazioni. Oggi, invece, la troviamo anche nelle automobili, integrata nella gestione elettronica di bordo, per attivare automaticamente i fari in galleria o al tramonto. Tale controllo viene implementato attraverso quel circuito elettronico noto come interruttore crepuscolare, che basandosi su un sensore di



illuminamento, determina quando viene superata la soglia (di norma regolabile per adattarsi ai vari ambienti e condizioni operative) che determina l'attivazione di un relé cui è demandato il controllo della linea dell'illuminazione elettrica.

Il grande successo e la notevole richiesta di controlli del genere ci spinge a pubblicare periodicamente interruttori crepuscolari, tant'è che in queste pagine trovate il progetto di uno di essi: un circuito canonico, realizzato con elettronica convenzionale, che non richiede né microcontrollori né le competenze e la strumentazione per programmarli; un progetto che piacerà soprattutto a chi è alle prime armi e vuol realizzare il proprio primo

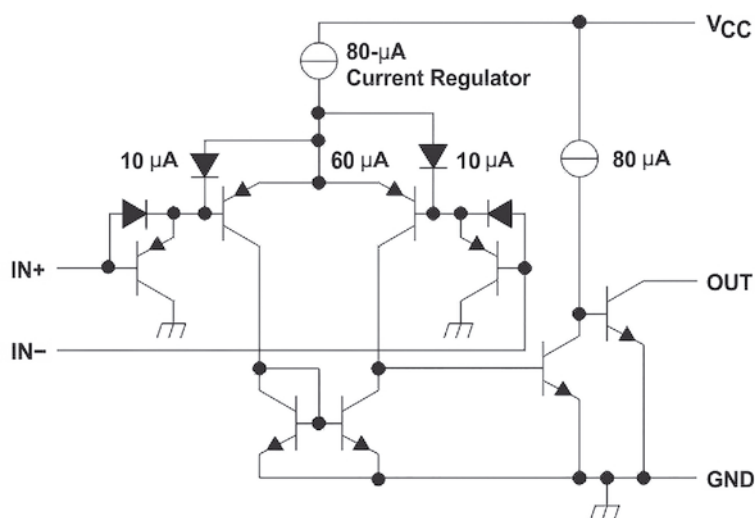
dispositivo elettronico.

Vediamo dunque di cosa si tratta, andando a studiarne il circuito.

SCHEMA ELETTRICO

Il nostro interruttore crepuscolare è un dispositivo essenziale basato sostanzialmente su un comparatore di tensione integrato, di tipo LM393, con uscita open-collector a transistor NPN, che confronta il potenziale determinato da un partitore in cui è collocato un fotosensore con quello di riferimento ottenuto da un trimmer, attivando la propria uscita in modo da controllare l'alimentazione della bobina di un relé, cui è demandato

Fig. 1
Schema elettrico di uno dei due comparatori contenuti nell'integrato LM393.



il compito di commutare la linea di alimentazione dell'utilizzatore elettrico.

Andiamo a vedere il dettaglio analizzando lo schema elettrico che trovate in queste pagine: il cuore del crepuscolare è il comparatore U1a (uno dei due comparatori contenuti nell'integrato LM393, ciascuno dei quali è basato sulla configurazione proposta nella **Fig. 1**) il cui ingresso invertente (pin 2) è polarizzato dalla tensione prelevata dal cursore del trimmer RVADJ (visibile nella **Fig. 2**), a sua volta alimentato dalla tensione della linea positiva che alimenta anche il piedino 8 dell'U1; l'input non-invertente di detto comparatore è polarizzato dal partitore resistivo formato dalla resistenza R2 e dal fotoresistore siglato CDS, tramite la R4.

Il comparatore è stato configurato come non-invertente dotato di isteresi operata dalla retroazione positiva attraverso la connessione dell'uscita con l'ingresso non-invertente effettuata dalla resistenza R7. Lo scopo dell'isteresi è fare in modo che quando il comparatore commuta lo stato della propria uscita da alto a basso o viceversa, la commutazione rimanga stabile anche nell'intorno di piccole variazioni di luminosità nell'ambiente, evitando così quel fastidioso fenomeno di pendolarismo che porterebbe allo sfarfallio delle lampade al momento in cui la luminosità nell'ambiente è intorno al valore di soglia.

Il fotoresistore è l'elemento fotosensibile del circuito, e il suo ruolo è consentire al comparatore di rilevare il livello di illuminazione dell'ambiente circostante. La sua caratteristica principale è quella di ridurre il proprio valore di resistenza (cioè la resistenza misurabile tra i suoi due terminali) con l'aumentare dell'intensità luminosa che colpisce la sua superficie sensibile; al contrario, quando l'intensità luminosa diminuisce (ovvero in condizioni di scarsa illuminazione), la resistenza aumenta. Nel nostro circuito, il fotoresistore fa partitore di tensione con una resistenza (la R2) collegata all'alimentazione positiva, pertanto la tensione ai suoi capi calerà all'aumentare dell'illuminamento dell'ambiente e aumenterà man mano che farà buio.

Grazie ai collegamenti del comparatore, finché la luminosità rilevata mantiene il potenziale del piedino 3 inferiore a quello del piedino 2, l'uscita

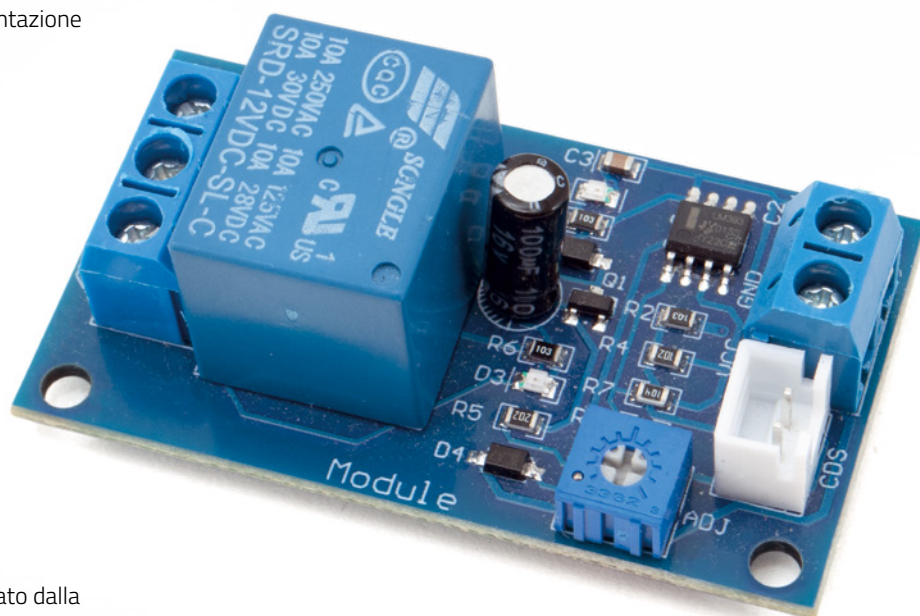


Fig. 2
Il crepuscolare
privo del
fotoresistore.

(piedino 1) resta a livello basso (con il transistor d'uscita in saturazione). Non appena la luminosità ambientale cala a sufficienza da far sì che il potenziale sull'ingresso non-invertente di U1a superi quello sull'ingresso invertente, l'uscita del comparatore passa a livello alto, portandosi a un valore prossimo al potenziale di alimentazione del piedino 8 dell'integrato. Questo avviene grazie al resistore di pull-up R3, indispensabile poiché l'uscita di U1a è open-collector. A questo punto, il transistor NPN Q1 viene polarizzato sulla base tramite la resistenza R5, con un certo ritardo dovuto alla presenza del condensatore elettrolitico C1, il quale, una volta caricato, introduce anche un ritardo nel rilascio del comparatore. Tale condensatore, insieme alla resistenza R5, definisce una costante di tempo che contribuisce a evitare false commutazioni del relé, in sinergia con l'isteresi del comparatore U1a. Il collettore del transistor Q1 alimenta la bobina del relé, facendone scattare l'equipaggio mobile e chiudere il contatto C (centrale) dello scambio con l'NO, ma anche il bipolo LED resistenza, che consente di ottenere un'indicazione visiva dello stato del relé: quando D3 è illuminato lo scambio del relé è chiuso tra C ed NO, mentre se è spento il relé è a riposo e C è chiuso su NC.

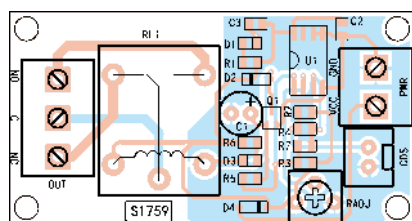
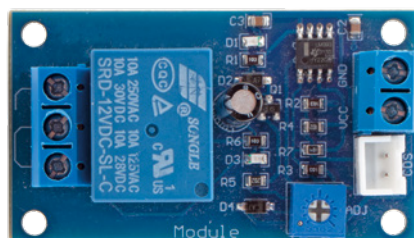
In parallelo alla bobina del relé RL1 è presente un diodo al silicio, posizionato per proteggere la giunzione collettore-emettitore del transistor. Questo diodo entra in conduzione e sopprime le extratensioni inverse generate dall'induttanza della bobina del relé quando viene disalimentata. In condizioni

Elenco Componenti:

R1, R2, R3, R6: 10 kohm (0805)
 R4: 1 kohm (0805)
 R5: 2 kohm (0805)
 R7: 100 kohm (0805)
 RVADJ: Trimmer 10 kohm
 C2, C3: 100 nF ceramico (0805)
 C1: 100 μ F 16 VL elettrolitico
 D1: LED verde (0805)
 D3: LED rosso (0805)
 D2, D4: MMSD4148T1G (SOD-123)
 U1: LM393 (SOIC8)
 Q1: SBC847BWT1G (SOT323)
 CDS: Fotoresistenza
 RL1: Relé 12Vdc - 250VAC/10A

Varie:

- Connettore JST 2.54mm
- Morsetto 2 vie passo 5.08 mm
- Morsetto 3 vie passo 5.08 mm
- Circuito Stampato S1759 (30x53 mm)



normali, il diodo D2 non conduce; tuttavia, quando il relè si disaccende, la bobina tende a mantenere la corrente precedente, generando una sovratensione inversa. La bobina, infatti, si comporta da generatore per scaricare l'energia accumulata durante l'alimentazione, e senza protezione, questa sovratensione potrebbe scaricarsi sulla giunzione base-collettore del transistor Q1, danneggiandola. In tale situazione, il diodo, essendo polarizzato direttamente (anodo positivo rispetto al catodo), va in conduzione e assorbe l'energia della bobina, evitando danni al circuito.

Stabilito questo, si osserva che il riferimento di tensione dell'U1a è impostato tramite il cursore del trimmer RVADJ, che consente di regolare la soglia di intervento dell'interruttore crepuscolare. Spostando il cursore verso massa, si abbassa la tensione di riferimento, permettendo al relé di attivarsi anche in presenza di un livello di luce ambientale relativamente alto. Al contrario, avvicinando il cursore al positivo dell'alimentazione, si aumenta il potenziale che il nodo R2-CDS deve raggiungere per determinare la commutazione del comparatore da zero a livello alto, richiedendo quindi un ambiente più buio affinché le luci si accendano. In sintesi, per aumentare la sensibilità del crepuscolare, si porta il cursore del trimmer verso massa (massima sensibilità), mentre per ridurla, si ruota verso il positivo.

È utile ora spiegare il funzionamento dell'isteresi, qui implementata sfruttando la configurazione che considera l'uscita open-collector dell'U1. Supponiamo che l'uscita dell'LM393 sia a livello basso e che il potenziale sull'ingresso non-invertente del comparatore superi quello sull'invertente. La retroazione positiva ha l'effetto di abbassare ulteriormente il potenziale del piedino 3, poiché R7 forma un partitore di tensione con R4. Trascurando per semplicità la caduta di tensione sulla R2 causata dal carico introdotto dalla chiusura a massa della resistenza di retroazione positiva, si può approssimare che la tensione sull'ingresso non-invertente diminuisca di circa 1/100 rispetto al valore che aveva causato la commutazione dell'uscita da alto a basso.

Ora, se il potenziale sull'ingresso invertente si abbassa ulteriormente, per far commutare l'uscita del comparatore a livello alto, dovrà scendere di oltre quel centesimo di volt.

Ad esempio, se la soglia di commutazione senza retroazione è di 1 volt, il potenziale dovrà scendere al di sotto di 0,99 V. Quando ciò accade, l'uscita dell'U1a passa a livello alto e la R7 innalza il potenziale sul piedino 3, rafforzando la commutazione e aumentando la separazione tra le tensioni sugli ingressi. In effetti, il partitore formato da R4 e R7 ora fornisce corrente a R4, aumentando di circa 1/100 il potenziale dell'ingresso non-invertente. Di conseguenza, per ottenere una nuova commutazione dell'uscita da alto a basso, il potenziale sull'ingresso invertente dovrà salire di circa 1/100 rispetto al valore della precedente commutazione da basso ad alto.

Bene, concludiamo la descrizione dello schema elettrico con la sezione di alimentazione, che fa

capo alla morsettiere siglata +/- PWR i cui poli, Vcc e GND, collegano l'ingresso rispettivamente al positivo e alla massa; la linea positiva passa attraverso il diodo al silicio D4, la cui funzione è proteggere il resto del circuito laddove si dovesse collegare l'alimentazione con la polarità rovesciata rispetto a quella prevista. La tensione a valle viene filtrata dai condensatori C2 e C3, che la ripuliscono da eventuali disturbi impulsivi e ad alta frequenza, concorrendo alla stabilità del funzionamento del comparatore e -di riflesso- dell'intero crepuscolare; inoltre alimenta il LED D1, attraverso il resistore R1 (utilizzato per limitarne la corrente diretta) il quale indica, illuminandosi, la presenza dell'alimentazione e quindi che il crepuscolare è in funzione. La tensione di alimentazione prevista per il circuito è di 12Vcc, meglio se stabilizzati.

REALIZZAZIONE PRATICA

Passiamo dunque agli aspetti pratici, descrivendo le fasi di costruzione del circuito, la prima delle quali coincide con la preparazione del circuito stampato, a doppia ramatura, ottenibile per fotoincisione dopo aver scaricato le tracce lato rame dal nostro sito elettronica.in.it ed averle stampate su carta da lucido o acetato per ricavarne le pellicole occorrenti a impressionare le due facce. Chi non attrezzato per realizzarsi la basetta, potrà richiederla attraverso il servizio realizzazione PCB della Futura Elettronica (tutti i dettagli e i costi sono consultabili su ftrlink.it/pcb) in pochi giorni.

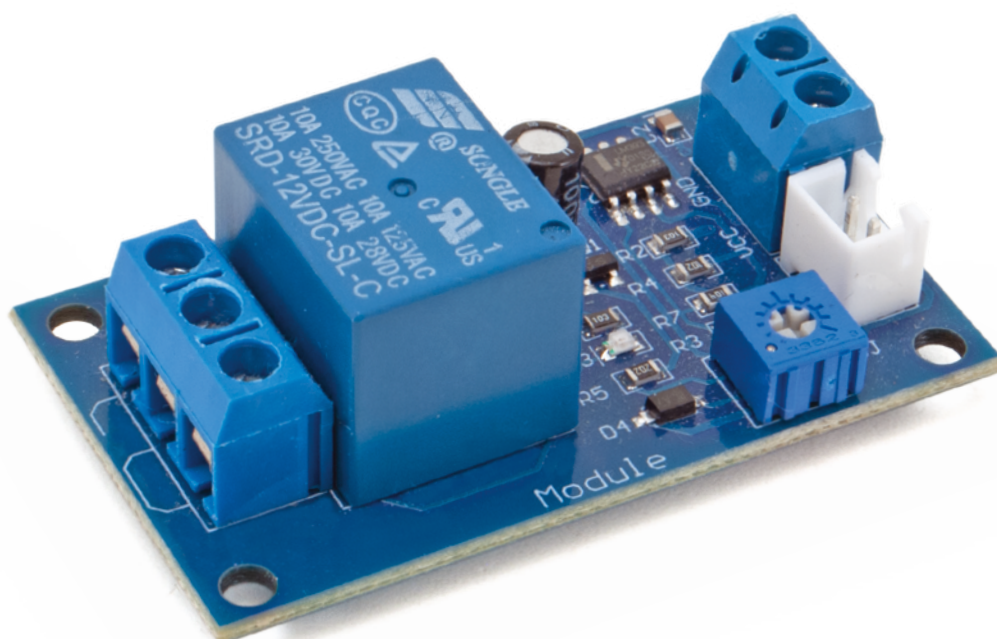
Gran parte dei componenti dell'interruttore cre-

CARATTERISTICHE TECNICHE

❖ Tensione di alimentazione: 12Vcc ❖ Corrente assorbita: 40 mA ❖ Tensione commutabile: 250Vca	❖ Corrente commutabile: 10 Ac.a. ❖ Uscita a relé con scambio intero ❖ Regolazione della soglia d'intervento
--	--

puscolare sono a montaggio superficiale, pertanto dovrete disporre di una pinzetta per posizzarli, di filo in lega saldante sottile (non oltre 0,5 mm di diametro) e della pasta flussante, oltre che di un saldatore con punta sottile.

Una volta ottenuto il circuito stampato, si inizierà il montaggio con le resistenze e i diodi, per i quali occorre rispettare la polarità indicata (nei diodi al silicio il catodo è evidenziato da una striscia grigia, mentre nei LED è marchiato con una banda colorata, tipicamente verde) e poi passare all'integrato LM393, staggandone uno dei piedini di un lato per fermarlo ben centrato sulle rispettive piazzole, quindi saldando i rimanenti procedendo uno per lato, per non surriscaldare troppo il componente.



← Del relé è disponibile l'intero scambio tramite una morsettiere tripolare.

Nei circuiti elettronici, quando bisogna rilevare il livello di una fonte luminosa o l'illuminamento dell'ambiente si può fare ricorso ad uno speciale resistore il cui valore varia in funzione proprio dell'entità della radiazione luminosa che lo investe; tale componente è il fotoreistore (anche detto "fotoresistenza") ed è un resistore variabile che risente della luce e viene quindi utilizzato come sensore nelle fotocellule o dovunque serva rilevare la luce del giorno, quella delle lampade elettriche o gli infrarossi.

Il fotoreistore è composto da un elemento resistivo appoggiato a un supporto e ricoperto da uno strato protettivo che gli permette di restare esposto all'ambiente esterno; lo strato è ottenuto con un impasto di materiale semiconduttore, che tipicamente è solfuro di cadmio (CdS) o tellururo di cadmio (CdTe).

Quando viene investito dalla luce, il composto libera al proprio interno un certo numero di elettroni, la cui disponibilità per la conduzione abbassa il valore resistivo e quindi agevola lo scorrimento di corrente.

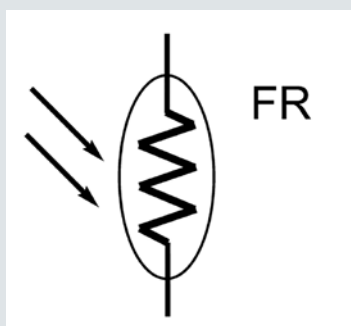
Per questa ragione il componente presenta una resistenza che è

relativamente elevata (anche oltre 1 Mohm) al buio e diventa sempre più bassa (fino a poche centinaia di ohm) in piena luce.

Come tutti i resistori, anche la fotoresistenza ha due soli terminali e si può collegare in qualsiasi modo (si fa percorrere dalla corrente egualmente nei due sensi).

Il simbolo grafico del fotoreistore usato nel disegno elettronico lo vedete nello schema elettrico e nell'immagine in questo riquadro. Prima dell'avvento del fotodiodo, il fotoreistore è stato il più importante elemento fotosensibile, non perché fosse l'unico componente in grado di rilevare la luce (già lo facevano le fotocellule a vuoto basate sull'effetto fotoemittente) ma per il fatto di essere il più piccolo sensore di luce e di poter funzionare anche a bassissima tensione.

Per la sua ottima sensibilità alla luce del giorno e a quella delle lampade elettriche a filamento (incandescenza comuni ed alogene) si presta a realizzare interruttori crepuscolari, sensori di prossimità, barriere ottiche e contapezzi per oggetti sui nastri trasportatori, automatismi per l'accensione dei fari delle auto ecc.



←
Simbolo
grafico del
fotoreistore.

Passate adesso ai componenti THT, ossia da montare a foro passante, inserendo e saldando nei rispettivi fori prima il trimmer, poi le due morsettiere a passo 5 mm (quella bipolare per l'alimentazione e la tripolare per rendere accessibile lo scambio del relé) l'elettrolitico (per la cui polarità seguite il piano di montaggio illustrato in queste pagine...) ed il relé, il quale entrerà nel circuito stampato solamente nel verso giusto. Il fotoreistore potete decidere se montarlo direttamente sullo stampato o collegarlo con due fili, da saldare o intestare su

un connettore a passo 2,54 mm femmina, il cui maschio salderete poi nelle rispettive piazzole del circuito stampato.

Bene, con questo avete completato il vostro circuito, che, dopo una verifica delle saldature e della corretta posizione dei componenti, è pronto per l'uso in quanto non richiede alcuna taratura preliminare. L'unica regolazione richiesta è quella del trimmer, che però dovrete effettuare una volta installato il circuito, possibilmente nelle condizioni di illuminazione ambientale (verso il tramonto...) in corrispondenza delle quali volete che si accendano o spengano le lampade collegate al relé.

A proposito di collegamenti, la linea delle lampadine andrà interrotta in un filo, un estremo del quale andrà fissato (opportunamente sguainato) nel morsetto C e l'altro nell'NO (sempre dopo averlo liberato dall'isolante).

Ricordiamo che il circuito è idoneo a controllare lampade funzionanti sia a bassa tensione (ad esempio quelle a LED) sia alla tensione di rete 230 Vca, con le dovute cautele nel maneggiarlo, in quest'ultimo caso.



COSA OCCORRE ?

Codice	Descrizione	Prezzo € IVA inclusa
Y171	Interruttore crepuscolare montato	5,50

IL MATERIALE VA RICHIESTO A:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - www.futurashop.it

NUOVI ALIMENTATORI SWITCHING DA RETE

COD. ER24W24V

€ 10,50



ALIMENTATORE SWITCHING
24V / 1A - 24W

Tensione di ingresso: da 90 a 240 VAC - 50/60 Hz
Potenza: 24 W
Tensione in uscita: 24 Vdc
Massima corrente di uscita: 1 A
Tensione di ripple: 200 mVpp
Consumo di potenza a vuoto (max): 0.075 W
Lunghezza del cavo: 1,5 metri
Plug: 5,5 x 2,1 mm, positivo centrale
Dimensioni (mm): 82,3 x 32,6 x 74,1 (L x W x H)
Peso: 112 grammi

ALIMENTATORE SWITCHING
6V / 1A - 6W

Tensione di ingresso: da 90 a 240 VAC - 50/60 Hz
Efficienza: 75%
Potenza: 6 W
Tensione in uscita: 6 Vdc
Massima corrente di uscita: 1 A
Tensione di ripple: 200 mVpp
Consumo di potenza a vuoto (max): 0.075 W
Lunghezza del cavo: 1,5 metri
Plug: 5,5 x 2,1 mm, positivo centrale
Temperatura di lavoro: 0°C ÷ 40°C
Dimensioni (mm): 38 x 25 x 69,6 (L x W x H)
Peso: 80 grammi

COD. ER06W06V

€ 7,00



Prezzi IVA inclusa.

**VASTA GAMMA DI MODELLI
PER TUTTE LE ESIGENZE**

DISPONIBILI ANCHE CON LE SEGUENTI CARATTERISTICHE:

CODICE		TENSIONE D'INGRESSO	TENSIONE D'USCITA	CORRENTE D'USCITA	POTENZA	RIPPLE	DIMENSIONI	PREZZO
ER09W09V		90 a 240 VAC	9 VDC	1 A	9 W	150 mVpp	76,2 x 27,6 x 61,6 mm	7,00
ER12W12V		90 a 240 VAC	12 VDC	1 A	12 W	150 mVpp	76,2 x 27,6 x 61,6 mm	7,50
ER15W15V		90 a 240 VAC	15 VDC	1 A	15 W	200 mVpp	79,2 x 27,6 x 69,1 mm	8,50
ER24W12V		90 a 240 VAC	12 VDC	2 A	24 W	120 mVpp	82,3 x 32,6 x 74,1 mm	11,50

ANCHE I DETTAGLI PIÙ PICCOLI DIVENTANO VISIBILI!

**MICROSCOPI DIGITALI IN GRADO DI CATTURARE IMMAGINI
CON UN'INGRANDIMENTO FINO A 1200X.**

MICROSCOPIO DIGITALE 1-1200X CON MONITOR LCD A COLORI DA 7"

Microscopio digitale con ampio monitor LCD a colori da 7 pollici e ingrandimento fino a 1200X. Dispone di una telecamera a colori che riprende l'oggetto da ingrandire e lo visualizza in alta risoluzione sullo schermo LCD.

Le foto e i video degli oggetti ingranditi possono essere salvati su una SD card inserita nell'apposito slot (max 128GB, acquistabile separatamente). Il microscopio dispone di una base stabile in alluminio e di doppia luce a LED con bracci flessibili. La distanza del punto di osservazione del campione può essere facilmente regolata tramite l'asta a cremagliera che consente di ottenere una distanza che va da 10 a 130 mm. Lo schermo è ruotabile di 360° e inclinabile. Può funzionare tramite batteria al litio integrata oppure tramite l'alimentatore USB in dotazione.



cod. LCDMICROS7

€154,00



Prezzi IVA inclusa.

MICROSCOPIO DIGITALE CON INGRANDIMENTO FINO A 600X CON MONITOR LCD DA 4,3"

Microscopio digitale portatile, dotato di telecamera a colori con sensore da 3,6 Megapixel, monitor LCD a colori da 4,3", illuminatore composto da 8 potenti LED a luce bianca, batteria al litio ricaricabile integrata, slot per micro SD card (max 64GB) e staffa a ventosa.



cod. LCDMICROS

€102,00

MICROSCOPIO USB DIGITALE CON INGRANDIMENTO FINO A 500X

Microscopio USB digitale a colori da 2 Megapixel dotato di stand con supporto regolabile in altezza. Si collega alla porta USB e consente di catturare e visualizzare immagini e video direttamente sullo schermo del computer. Dispone di 8 LED bianchi, con intensità luminosa regolabile.



cod. CP745

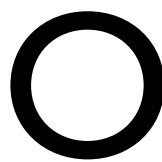
€28,00



RILEVATORE FACCIALE CON AI MEDIAPIPE

di ALOISIO ANTONIO

Analizziamo un software capace di riconoscere le espressioni del viso e tra esse lo stato di chiusura degli occhi per generare un allarme acustico.



Ogni giorno sono numerose le informazioni che giungono dal mondo dell'**intelligenza artificiale (AI)**.

È un argomento che sta prendendo e prenderà sempre più spazio in quasi tutti gli ambiti del vivere quotidiano.

Si sta ancora analizzando quale impatto possa avere nel mondo del lavoro,

poiché è evidente come in tanti compiti il suo utilizzo sia molto performante. Non entriamo nel dibattito morale e legislativo se tali sistemi vadano o no regolamentati. Affrontiamo di contro un compito sociale, per cui l'intelligenza artificiale dovrebbe essere fondamentale, la sicurezza stradale. Come sempre il nostro approccio vuole essere didattico, per capire le "belle" opportunità che questa tecnologia ci offre.

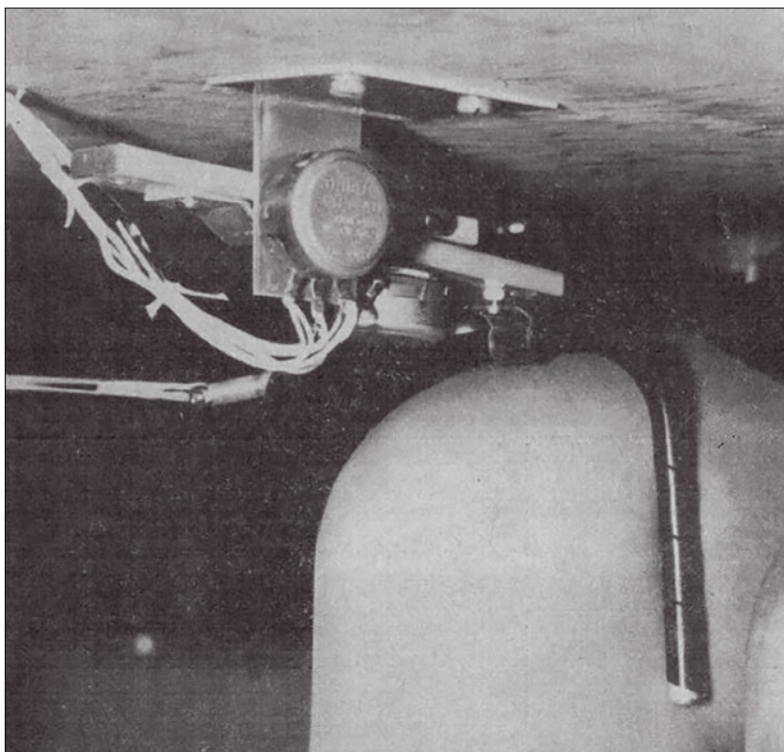
Il progetto che presentiamo in questo articolo è un rilevatore di stanchezza per guidatori, pensato per prevenire gravi conseguenze dovute ai colpi di sonno. Precisiamo che questo dispositivo non intende né sostituire né affiancarsi ai sistemi di assistenza alla guida certificati e collaudati, ma ha lo scopo di dimostrare l'utilità di questa tecnologia.

L'obiettivo di questo articolo è quindi spiegare come utilizzare la libreria di "machine learning" (ML) di Google, MediaPipe, per riconoscere, attraverso lo streaming video di un volto, se gli occhi sono chiusi o socchiusi, segnali tipici di un possibile addormentamento alla guida.

Rispetto alla articolo pubblicato sulla rivista n° 280, dove riconoscevamo i gesti delle mani per comandare una TV con segnali ad infrarosso, qui useremo un insieme di punti riconosciuti sul volto (**landmark**), per identificare i movimenti delle palpebre e di conseguenza la loro chiusura.

Se il movimento di chiusura è verificato per un determinato tempo, anche di un solo occhio, suonerà un allarme. Non ci soffermeremo molto sulla presentazione di MediaPipe, avendola già trattata. Procederemo allo sviluppo mediante la programmazione in **Python**, linguaggio di programmazione fondamentale per entrare nel mondo della AI. Potrete eseguire il programma direttamente sul vostro PC, con il vostro editor preferito, noi

Fig. 1
Sistema di
interazione uomo
macchina mediante
ginocchio
(fonte IEEE 1964).



consigliamo Visual Studio Code. Nella parte finale dell'articolo vedremo come integrare il tutto su una scheda Raspberry Pi con buzzer collegato sui GPIO, in modo da rendere il sistema portatile e soprattutto indipendente dalla connessione ad internet.

UN PO' DI STORIA

Il riconoscimento facciale è una tecnologia che spazia dalla sicurezza alla personalizzazione dell'esperienza utente. Sui nostri smartphone viene usato per accedere al dispositivo ed è notizia recente l'implementazione anche in Italia del face boarding per i passeggeri di alcuni aeroporti, in sostituzione delle carte di imbarco. Abbiamo fantasticato su di essa in tanti film di fantascienza, oggi molto più vicina di quanto potessimo immaginare.

Giungere alla migliore esperienza di utilizzo della tecnologia è una sfida che coinvolge da anni molte figure professionali. Ad esempio, nel documento "Display Selection Techniques for Text Manipulation" (Tecniche di selezione per la manipolazione del testo), pubblicato nel marzo 1967, si esploravano le migliori tecniche per la manipolazione del testo con il computer.

In questo articolo è interessante notare come l'interfaccia tra uomo e macchina sia passata da step di test e sviluppo, che hanno confermato alcune tecnologie e cancellate altre.

I dispositivi testati includevano: light-pen, Joystick, Mouse sviluppato da SRI, Controllo con il ginocchio, Grafacon. Il documento entra nel dettaglio della loro funzionalità ed facilità di uso, rendendo "vincitore" del confronto appunto il mouse.

Per quanto possa far sorridere un sistema di controllo con ginocchio **Fig. 1**, il **Grafacon Fig. 2** ha destato il nostro maggiore interesse.

Approfondendo l'argomento, abbiamo trovato delle similitudini tra i landmark forniti da MediaPipe ed il Grafacon. Questo dispositivo, nelle sue prime versioni, forniva le coordinate dei punti selezionati, con due movimenti, rotativo su un asse ed assiale. In uno spazio piano come, ad esempio un foglio, paragonabile ad un asse cartesiano, bastano quindi due coordinate x - y per identificarne un punto. Nel tempo si sarebbe poi evoluto nella penna grafica (Rand Tablet) e comunque accantonata come tecnologia.

Ma il suo utilizzo fu fondamentale per i pionieri del riconoscimento facciale come Woodrow W. Bledsoe, Helen Chan e Charles Bisson. Tra il 1964 e il 1966, questi scienziati hanno svolto ricerche innovative presso la Panoramic Research a Palo Alto, California, finanziate da un'agenzia di intelli-

gence non ben specificata.

L'obiettivo del progetto era sviluppare un programma che potesse riconoscere volti umani da un ampio database di immagini, come un libro di fotografie segnaletiche, e estrarre da esse un set di dati che potesse corrispondere alla fotografia. Questo problema di riconoscimento era complicato dalla grande variabilità dovuta alla rotazione e alla inclinazione della testa, alla intensità e all'angolo della luce, alle espressioni facciali e all'invecchiamento.

Mediante il Grafacon gli operatori umani estraevano le coordinate di una serie di caratteristiche delle foto, come il centro delle pupille, l'angolo interno ed esterno degli occhi, il punto distinto della linea dei capelli posto al centro della fronte e così via. Tutti landmark che anche MediPipe ci fornisce.

Da queste coordinate, venivano calcolate 20 distanze, come la larghezza della bocca e la distanza pupilla a pupilla. Il programma determinava l'inclinazione, il piegamento e la rotazione della testa e ne calcolava gli angoli. Si usarono oltre 2000 fotografie, ed alla conclusione del progetto fu dimostrato che il computer superava costantemente gli umani nei compiti di riconoscimento facciale (fonte www.historyofinformation.com)

MEDIPIPE

Abbiamo accennato come MediaPipe faccia oggi, solo mediante software, quello che negli anni 60 era possibile con sistemi hardware come il Grafacon. Andiamo sul sito https://mediapipe-studio.webapps.google.com/demo/face_landmarker e mettiamo subito alla prova. Autorizzate all'uso della webcam del vostro pc, ed in breve tempo comparirà un avatar di un orsetto sul vostro volto **Fig. 3**. Per quanto sia carino, il suo scopo è farci vedere come MediaPipe possa riconoscere le espressioni facciali, elaborando i landmark del volto. Ma vediamo invece questi punti, perché più utili nel nostro percorso, cambiando l'impostazione "Display mode" da "Avatar" ad "Grid Render" **Fig. 4**.

Il software traccia nel frame 478 landmark ed unisce i punti tra loro creando una griglia. In **Fig. 5** vi è un dettaglio di un volto, preso direttamente dal sito ufficiale di MediaPipe ove i landmark sono numerati, per essere distinti e quindi identificati. Abbiamo usato il browser per vedere la griglia dei landmark, quindi intuimmo che il sistema sia multi piattaforma, utilizzando per il web il linguaggio JavaScript. Ma uno dei nostri obiettivi è eseguire in locale, senza necessità di connessione ad internet, la potenza di MediaPipe.

Scarichiamo ora il modello di riconoscimento fac-

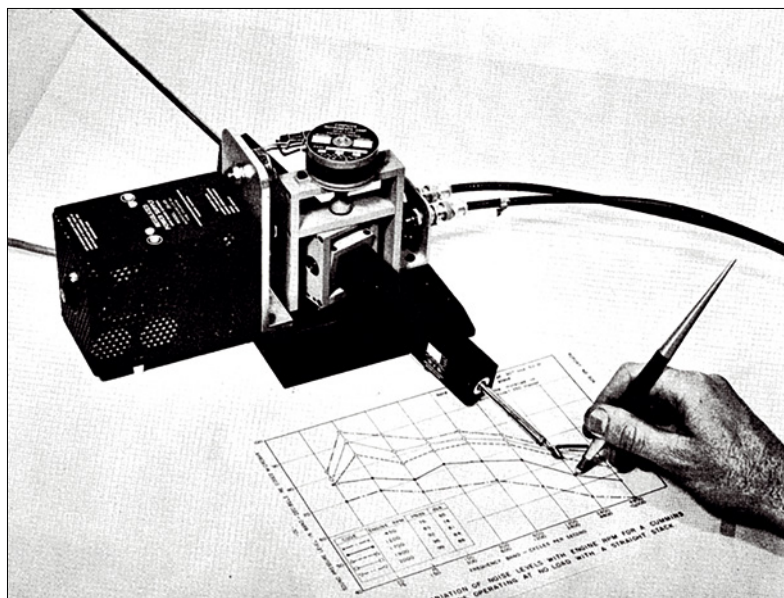


Fig. 2
Uno dei primi Grafacon, con interazione rotativa e assiale (fonte Decus 1963).

ciale (https://ai.google.dev/edge/mediapipe/solutions/vision/face_landmarker/index?hl=it#models) e salviamolo nella cartella di lavoro Python rinominandolo, se differente, in "face_landmarker.task".

Dopo aver installato le librerie MediaPipe (pip install mediapipe) ed OpenCV (pip install opencv-python) per la manipolazione delle immagini dalla webcam, apriamo il nostro editor Python e mandiamo in esecuzione il **Listato.1**.

Fig. 3
Avatar che permette di notare le capacità di MediaPipe di riconoscere le espressioni facciali.



Fig. 4
Demo MediaPipe
con i landmark e i
punti di congiun-
zione tra essi che
creano una griglia.

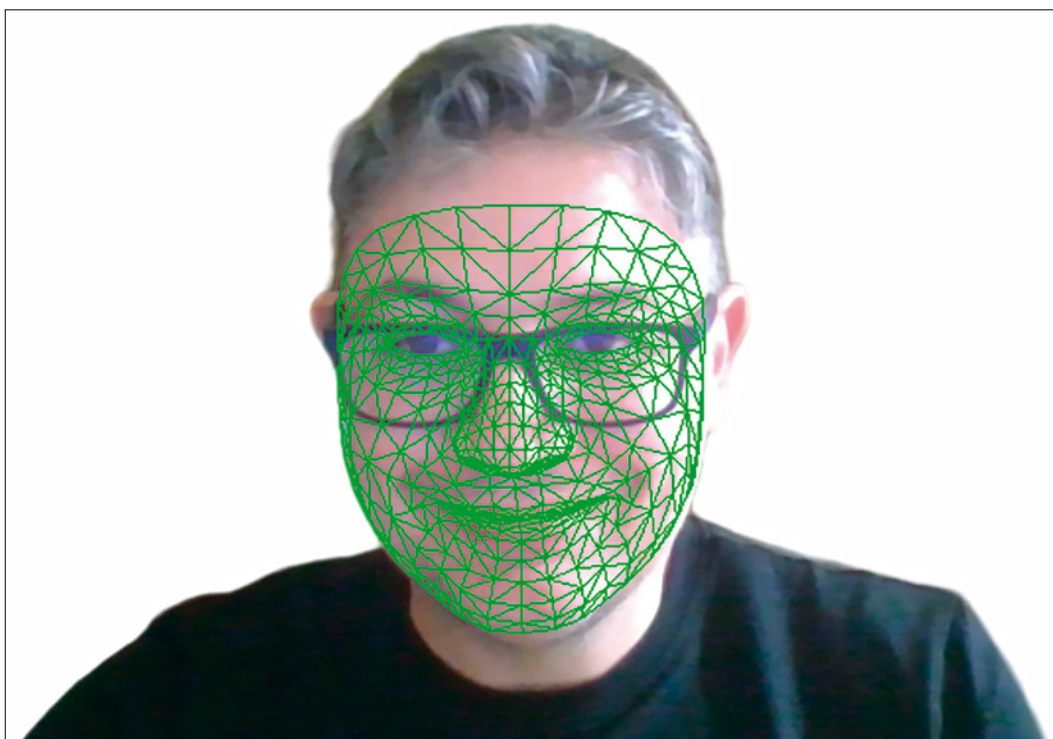
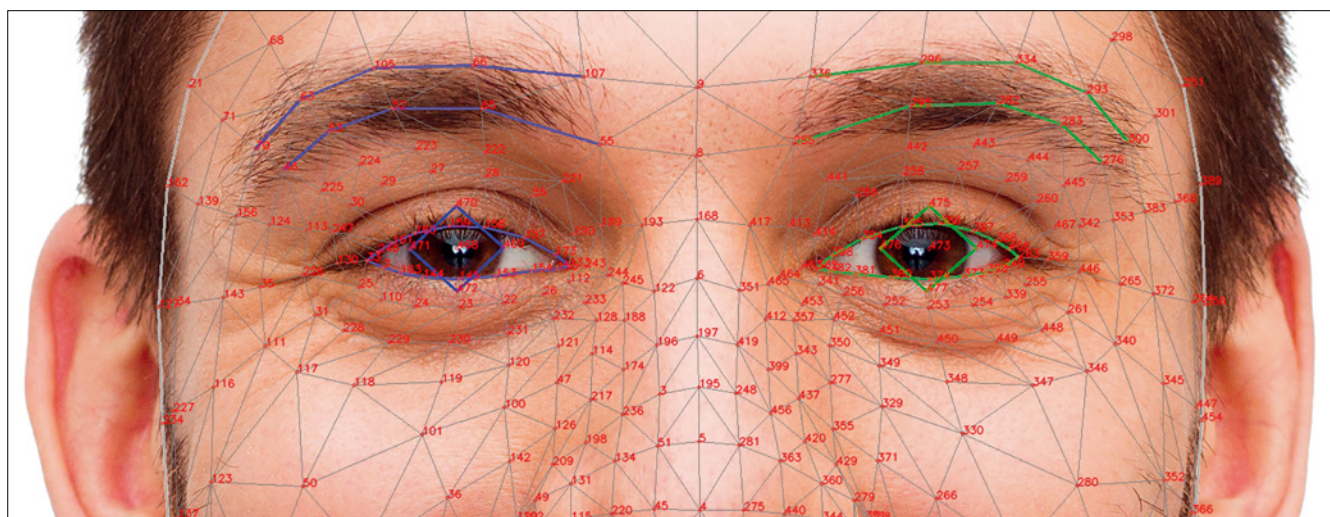


Fig. 5
Dettaglio di imma-
gine campione dal
sito web di Media-
Pipe con la nume-
razione di alcuni
dei 478 landmark
riconosciuti.



Si aprirà una finestra ed avremo il nostro volto contornato dalla griglia dei landmark come visto in precedenza sul sito web.

Noterete nel codice la presenza di vari punti commentati, questo per seguire un percorso di apprendimento, che ci permetta di avanzare a piccoli passi e quindi di analizzare non il solo listato finito.

EVIDENZIAMO IL LANDMARK 4

Il primo passo sarà quello di evidenziare un landmark specifico, ad esempio il **numero 4**

situato più o meno sulla punta del naso. Cancellate i commenti dai gruppi di righe comprese tra : `"""` *Primo passo da de-commentare* ed `"""`. Mandate in esecuzione, dovrebbe comparire un bel pallino rosso sul vostro naso, che si muove con voi, ed al suo lato le coordinate. Esse sono riferite al punto $x=0$, $y=0$ situato il alto a sinistra del frame (sempre rispetto a come guardate il frame sul vostro schermo).

Se alzate la testa e la spostate a sinistra i valori infatti diminuiranno verso lo zero (**Fig. 6**).

▼ Listato 1

```
import cv2
import mediapipe as mp
import math
# Inizializzazione di MediaPipe Face Mesh per tracciare i landmark sul frame
mp_face_mesh = mp.solutions.face_mesh
mp_drawing = mp.solutions.drawing_utils
def disegna_landmark(frame, landmarks):

    # Disegna i landmark facciali sul frame.
    for face_landmarks in landmarks:
        # Disegna tutti i landmark facciali e le connessioni
        mp_drawing.draw_landmarks(
            image=frame,
            landmark_list=face_landmarks,
            connections=mp_face_mesh.FACEMESH_TESSELATION,
            landmark_drawing_spec=None,
            connection_drawing_spec=mp_drawing.DrawingSpec(color=(0, 255, 0), thickness=1, circle_radius=2),
        )
        ''' Primo passo da decommentare
        # Landmark 4 (punta del naso)
        landmark_4 = face_landmarks.landmark[4]
        x4, y4 = int(landmark_4.x * frame.shape[1]), int(landmark_4.y * frame.shape[0])
        cv2.circle(frame, (x4, y4), 5, (0, 0, 255), -1)
        cv2.putText(frame, f"({x4},{y4})", (x4,y4-10), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0,0,255), 1, cv2.LINE_AA)
        '''
        ''' Secondo passo da decommentare
        # Landmark 105 sopracciglio destro
        landmark_105 = face_landmarks.landmark[105]
        x105, y105 = int(landmark_105.x * frame.shape[1]), int(landmark_105.y * frame.shape[0])
        # Calcola la distanza tra 4 e 105
        distanza_105_4_px = math.sqrt((x105 - x4) ** 2 + (y105 - y4) ** 2)
        cv2.putText(frame, f"{distanza_105_4_px:.2f}px", (x105,y105-20), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,0,0), 1, cv2.LINE_AA)
        # Disegna una linea tra i landmark 4 e 105
        cv2.line(frame, (x4, y4), (x105, y105), (255, 0, 0), 2)
        '''

def main():
    cap = cv2.VideoCapture(0)
    with mp_face_mesh.FaceMesh(min_detection_confidence=0.5, min_tracking_confidence=0.5) as face_mesh:
        while cap.isOpened():
            ret, frame = cap.read()
            if not ret:
                break
            frame = cv2.flip(frame, 1)
            frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
            risultati = face_mesh.process(frame_rgb)
            if risultati.multi_face_landmarks:
                disegna_landmark(frame, risultati.multi_face_landmarks)
            cv2.imshow('Riconoscimento Landmark Facciali', frame)
            if cv2.waitKey(1) & 0xFF == 27: # Esci se si preme il tasto Esc
                break
    cap.release()
    cv2.destroyAllWindows()
if __name__ == "__main__":
    main()
```

Analizziamo il codice appena de-commentato. In OpenCV, quando carichiamo o catturiamo un'immagine (o un frame da un video), possiamo estrarre un attributo chiamato 'shape' che fornisce il numero di righe di pixel nell'immagine che identificano altezza (y) ed il suo numero di colonne in pixel che identificano la larghezza (x). Usando le istruzioni:

- 'frame.shape[0]': ottengo quanti pixel ci sono verticalmente nell'immagine.
 - 'frame.shape[1]': ottengo quanti pixel ci sono orizzontalmente nell'immagine.
- Nel contesto del codice che stiamo esaminando, usiamo 'frame.shape[0]' e 'frame.shape[1]' per convertire le coordinate dei landmark facciali in coordinate di pixel effettive. Le coordinate dei landmark restituiti da MediaPipe sono valori compresi tra 0 e 1, rappresentando la posizione relativa del landmark rispetto all'immagine.



Fig. 6
Verifica sul frame video della posizione del landmark 4, punta del naso.

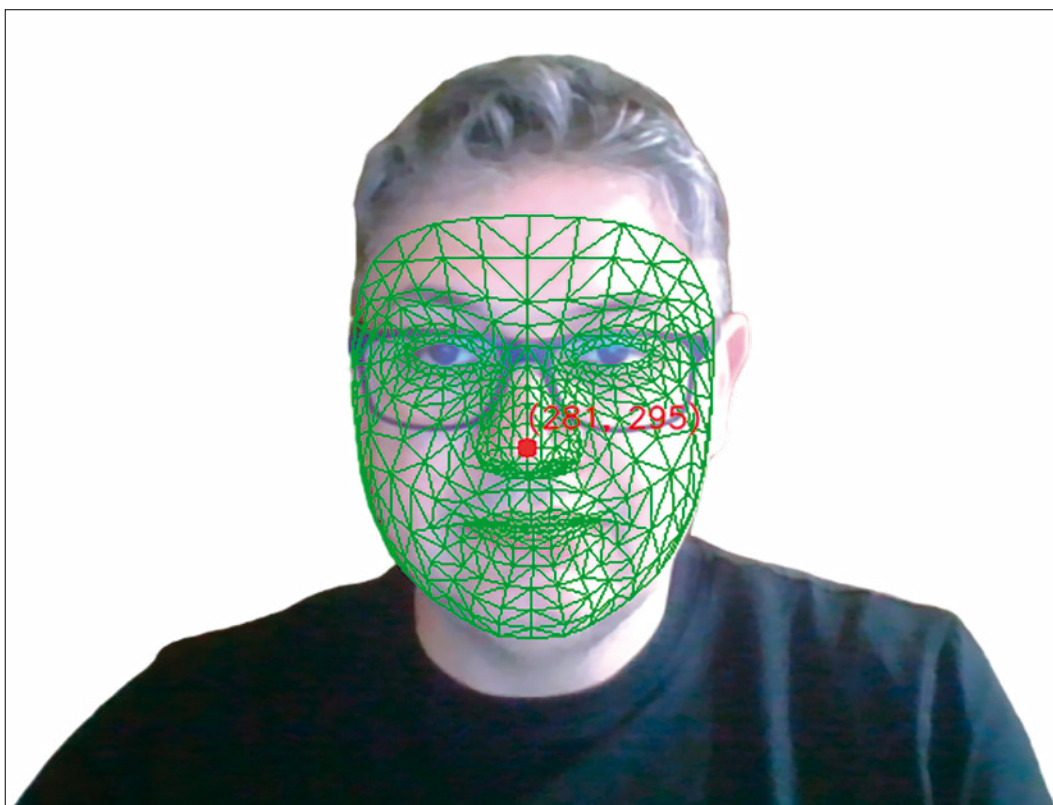
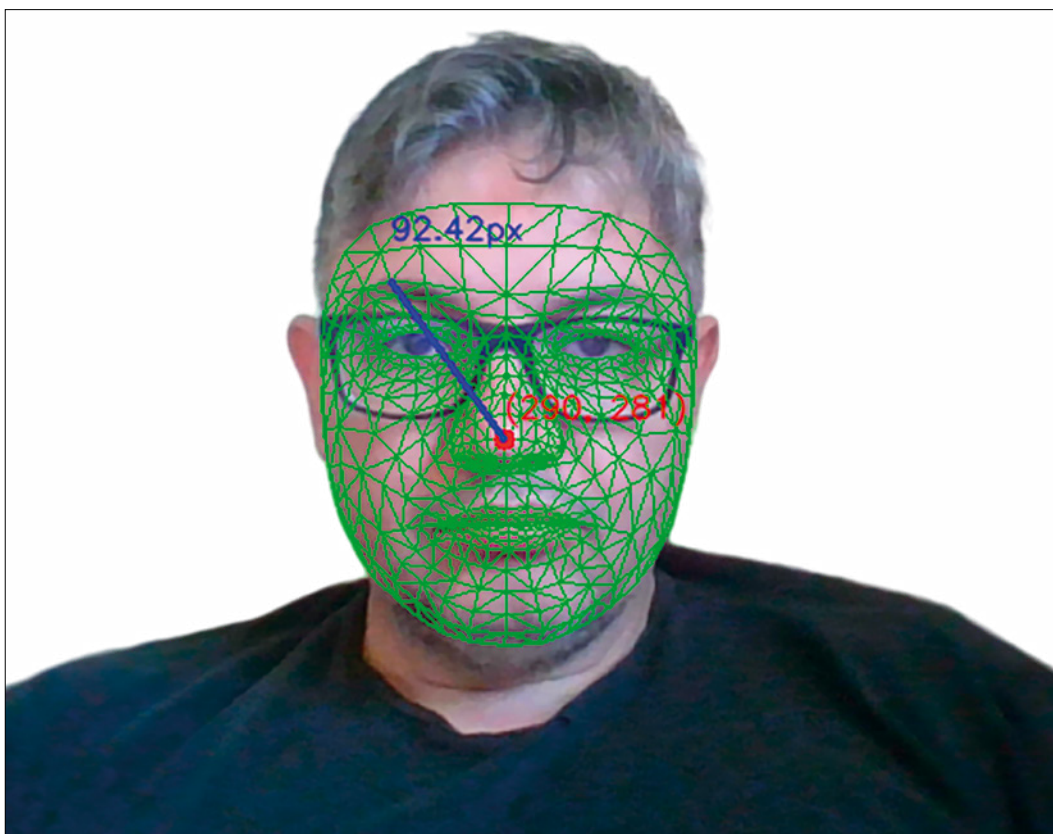


Fig. 7
Tracciamento di una linea blu tra i landmark 4 e 105.



Per ottenere le coordinate reali in pixel, moltiplichiamo questi valori per l'altezza e la larghezza dell'immagine (frame).

TRACCIAMO UNA LINEA TRA I LANDMARK 4 E 105

Proviamo ora ad tracciare una linea tra due landmark, sopracciglio destro, landmark 105 e il già definito punta del naso, landmark 4. Cancellate i commenti dai gruppi di righe comprese tra : `"""` *Secondo passo da de-commentare* ed `"""`.

Mandate in esecuzione, dovrebbe comparire una linea blu tra il sopracciglio e la punta del naso con la sua misura espressa in pixel **Fig. 7**.

Per calcolare la distanza tra i due landmark utilizziamo il teorema di Pitagora.

Quindi la formula:

$$\text{distanza_105_4_px} = \text{math.sqrt}((x105 - x4)^2 + (y105 - y4)^2)$$

si spiega nel seguente modo:

- $x105 - x4$: La differenza tra le coordinate x dei due punti.
- $y105 - y4$: La differenza tra le coordinate y dei due punti.
- $(x105 - x4)^2$: Eleviamo al quadrato la differenza delle coordinate x.
- $(y105 - y4)^2$: Eleviamo al quadrato la differenza delle coordinate y.
- Sommiamo i due quadrati:
 $(x105 - x4)^2 + (y105 - y4)^2$.
- Eseguiamo la radice quadrata della somma dei quadrati per ottenere la distanza:
`math.sqrt(...)`.

Una misura così calcolata, potrebbe essere usata confrontata al suo valore minimo, per sapere se vengono inarcate le sopracciglia, in segno di stupore, sorpresa e similari. Iniziamo a capire come, scegliendo i giusti landmark tra i 148, e calcolando le giuste distanze in confronto a misure di riferimento potremmo giungere a riconoscere tante espressioni facciali. La distanza così calcolata, non è assoluta, infatti basta spostare il volto in avanti e indietro e i valori varieranno notevolmente, perché cambia la proporzione del volto rispetto al frame. Dovremmo quindi "normalizzare" la distanza riferendoci ad un'altra distanza che sappiamo non vari, come ad esempio i punti lacrimali. È sicuramente un lungo lavoro, fatto di analisi e calcoli matematici, ma non vi sconsigliate, MediaPipe ci aiuta anche in questo.

BLENDSHAPE

MediaPipe effettua le analisi che abbiamo visto fino ad ora e definisce direttamente 51 **blendshape** del volto, cioè diverse configurazioni muscolari a cui è abbinata una specifica espressione.

Di seguito l'elenco con una breve descrizione:

0. Neutral : nessuna espressione rilevata.
1. BrowDownLeft: Sopracciglio sinistro abbassato.
2. BrowDownRight: Sopracciglio destro abbassato.
3. BrowInnerUp: Parte interna delle sopracciglia sollevata.
4. BrowOuterUpLeft: Parte esterna del sopracciglio sinistro sollevata.
5. BrowOuterUpRight: Parte esterna del sopracciglio destro sollevata.
6. CheekPuff: Guance gonfiate.
7. CheekSquintLeft: Guancia sinistra contratta.
8. CheekSquintRight: Guancia destra contratta.
9. EyeBlinkLeft: Occhio sinistro chiuso.
10. EyeBlinkRight: Occhio destro chiuso.
11. EyeLookDownLeft: Occhio sinistro guarda in basso.
12. EyeLookDownRight: Occhio destro guarda in basso.
13. EyeLookInLeft: Occhio sinistro guarda verso il naso.
14. EyeLookInRight: Occhio destro guarda verso il naso.
15. EyeLookOutLeft: Occhio sinistro guarda verso l'esterno.
16. EyeLookOutRight: Occhio destro guarda verso l'esterno.
17. EyeLookUpLeft: Occhio sinistro guarda in alto.
18. EyeLookUpRight: Occhio destro guarda in alto.
19. EyeSquintLeft: Occhio sinistro socchiuso.
20. EyeSquintRight: Occhio destro socchiuso.
21. EyeWideLeft: Occhio sinistro spalancato.
22. EyeWideRight: Occhio destro spalancato.
23. JawForward: Mandibola spostata in avanti.
24. JawLeft: Mandibola spostata a sinistra.
25. JawOpen: Mandibola aperta.
26. JawRight: Mandibola spostata a destra.
27. MouthClose: Bocca chiusa.
28. MouthDimpleLeft: Fossetta sulla guancia sinistra.
29. MouthDimpleRight: Fossetta sulla guancia destra.
30. MouthFrownLeft: Angolo sinistro della bocca rivolto verso il basso.
31. MouthFrownRight: Angolo destro della bocca rivolto verso il basso.
32. MouthFunnel: Bocca a imbuto.
33. MouthLeft: Bocca spostata a sinistra.



Fig. 8
Elenco dei 51 blendshape con barra blu che aumenta tanto più la condizione sarà veritiera.

34. MouthLowerDownLeft: Labbro inferiore sinistro abbassato.
35. MouthLowerDownRight: Labbro inferiore destro abbassato.
36. MouthPressLeft: Labbro sinistro premuto.
37. MouthPressRight: Labbro destro premuto.
38. MouthPucker: Labbra increspate.
39. MouthRight: Bocca spostata a destra.
40. MouthRollLower: Labbro inferiore arrotolato verso l'interno.
41. MouthRollUpper: Labbro superiore arrotolato verso l'interno.
42. MouthShrugLower: Labbro inferiore sollevato.
43. MouthShrugUpper: Labbro superiore sollevato.
44. MouthSmileLeft: Angolo sinistro della bocca rivolto verso l'alto (sorriso).
45. MouthSmileRight: Angolo destro della bocca rivolto verso l'alto (sorriso).
46. MouthStretchLeft: Angolo sinistro della bocca tirato indietro.
47. MouthStretchRight: Angolo destro della bocca tirato indietro.
48. MouthUpperUpLeft: Labbro superiore sinistro sollevato.
49. MouthUpperUpRight: Labbro superiore destro sollevato.
50. NoseSneerLeft: Narice sinistra sollevata.
51. NoseSneerRight: Narice destra sollevata.



Listato 2

```
import cv2
import numpy as np
from mediapipe.tasks import python
from mediapipe.tasks.python import vision
import mediapipe as mp
#from gpiozero import LED
#led = LED(15)
#led.off()
# Impostazioni di base per il modello di rilevamento dei landmark facciali
base_options = python.BaseOptions(model_asset_path='face_landmarker.task', delegate="CPU")
options = vision.FaceLandmarkerOptions(base_options=base_options,
                                       output_face_blendshapes=True,
                                       output_facial_transformation_matrixes=True,
                                       num_faces=1)

detector = vision.FaceLandmarker.create_from_options(options)
# Inizializzazione di MediaPipe Face Mesh per tracciare i landmark sul frame
mp_face_mesh = mp.solutions.face_mesh
mp_drawing = mp.solutions.drawing_utils
# Inizializzazione della cattura video dalla webcam
cap = cv2.VideoCapture(0)
# Lista dei nomi dei blendshape di interesse
blendshapes_di_interesse = ['eyeBlinkLeft', 'eyeBlinkRight', 'eyeSquintLeft', 'eyeSquintRight']
# Funzione per filtrare i blendshapes di interesse
def filtra_blendshapes(blendshape_result):
    return {blendshape.category_name: blendshape.score for blendshape in blendshape_result if blendshape.
            category_name in blendshapes_di_interesse}
# Funzione per disegnare i valori dei blendshapes sul frame principale
def disegna_blendshapes(frame, blendshape_scores):
    overlay_texts = [
        ("CHIUSURA", blendshape_scores.get('eyeBlinkLeft', 0), (60, 40), (255, 0, 0)),
        ("SOCCHIUSSO", blendshape_scores.get('eyeSquintLeft', 0), (60, 60), (0, 255, 0)),
        ("CHIUSURA", blendshape_scores.get('eyeBlinkRight', 0), (350, 40), (255, 0, 0)),
        ("SOCCHIUSSO", blendshape_scores.get('eyeSquintRight', 0), (350, 60), (0, 255, 0))
    ]
    for text, score, position, color in overlay_texts:
        cv2.putText(frame, f"{text}: {score:.2f}", position, cv2.FONT_HERSHEY_SIMPLEX, 0.6, color, 2)
# Funzione per disegnare tutti i blendshapes su un secondo frame
def disegna_tutti_blendshapes(blendshape_result):
    height, width = 720, 900 # larghezza tra le colonne
    info_frame = np.zeros((height, width, 3), dtype=np.uint8)
    y0, dy = 20, 20
    column = 0
    num_columns = 2
    num_per_column = len(blendshape_result) // num_columns
    for i, blendshape in enumerate(blendshape_result):
        y = y0 + (i % num_per_column) * dy
        x = 10 + (column * 450) # spazio tra le colonne
        bar_length = int(blendshape.score * 50)
        cv2.putText(info_frame, blendshape.category_name, (x, y), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255, 255), 1)
        cv2.rectangle(info_frame, (x + 200, y - 10), (x + 250, y + 5), (0, 0, 255), 1)
        cv2.rectangle(info_frame, (x + 200, y - 10), (x + 200 + bar_length, y + 5), (255, 0, 0), cv2.FILLED)
        if (i + 1) % num_per_column == 0:
            column += 1
    return info_frame
# Soglia di verità per i blendshape
soglia = 0.5
# Funzione per gestire gli allarmi
def gestisci_allarmi(frame, blendshape_scores):
    alert_positions = [
        ('eyeBlinkLeft', (10, 40)),
        ('eyeBlinkRight', (290, 40)),
        ('eyeSquintLeft', (10, 60)),
        ('eyeSquintRight', (290, 60)),
    ]
    ]
```


↓ Listato 2 - segue

```
for category, position in alert_positions:
    if blendshape_scores.get(category, 0) > soglia:
        cv2.putText(frame, "-->", position, cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 3)
#         led.on()
#     else:
#         led.off()
# Funzione principale per eseguire il riconoscimento facciale

def main():
    with mp_face_mesh.FaceMesh(
        min_detection_confidence=0.5,
        min_tracking_confidence=0.5) as face_mesh:
        while True:
            ret, frame = cap.read()
            if not ret:
                break
            # Conversione del frame da BGR a RGB per i landmark e per i blendshapes
            rgb_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
            # Elaborazione dei landmark facciali
            results = face_mesh.process(rgb_frame)
            # Elaborazione dei blendshapes
            image = mp.Image(image_format=mp.ImageFormat.SRGB, data=rgb_frame)
            result = detector.detect(image)
            if result.face_blendshapes:
                blendshape_result = result.face_blendshapes[0]
                blendshape_scores = filtra_blendshapes(blendshape_result)
                info_frame = disegna_tutti_blendshapes(blendshape_result)
                disegna_blendshapes(frame, blendshape_scores)
                gestisci_allarmi(frame, blendshape_scores)
                # Mostra i frame separati solo se c'è un risultato
                cv2.imshow('Valori dei Blendshapes', info_frame)
            # Disegna i landmark facciali sul frame
            if results.multi_face_landmarks:
                for face_landmarks in results.multi_face_landmarks:
                    mp_drawing.draw_landmarks(
                        image=frame,
                        landmark_list=face_landmarks,
                        connections=mp_face_mesh.FACEMESH_TESSELATION,
                        landmark_drawing_spec=None,
                        connection_drawing_spec=mp_drawing.DrawingSpec(color=(0,255,0),thickness=1,circle_radius=2),
                    )
            # Visualizza i dati sul frame
            cv2.putText(frame, "OCCHIO DESTRO", (350, 20), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 0), 1)
            cv2.putText(frame, "OCCHIO SINISTRO", (60, 20), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 0), 1)
            # Mostra il frame principale
            cv2.imshow('Riconoscimento Landmark e Blendshapes volto', frame)
            # Se il tasto Esc viene premuto, interrompi il ciclo
            if cv2.waitKey(1) & 0xFF == 27:
                break
        # Rilascia la cattura video e chiudi tutte le finestre
        cap.release()
        cv2.destroyAllWindows()
# Esecuzione della funzione principale
if __name__ == "__main__":
    main()
```

Questi blendshape coprono un'ampia gamma di movimenti ed espressioni facciali, permettendo di rappresentare in modo dettagliato le diverse emozioni e stati d'animo di una persona. Per il nostro progetto ci concentreremo sul comportamento dell'occhio.

Più specificatamente ci interessa identificare questi blendshape:

- 9. EyeBlinkLeft: Occhio sinistro chiuso.
- 10. EyeBlinkRight: Occhio destro chiuso.
- 19. EyeSquintLeft: Occhio sinistro socchiuso.
- 20. EyeSquintRight: Occhio destro socchiuso.



MediaPipe restituirà un valore variabile tra 0 ed 1, quindi con la virgola, quanto più tendente ad 1 tanto più la condizione dell'occhio sarà veritiera. Vediamo subito in azione MediaPipe con i 4 blendshape già definiti nel **Listato 2**.

In questo listato, rispetto al precedente, abbiamo tolto i tracciamenti didattici delle misure e dei punti tra i landmark, andando invece ad implementare solo l'analisi dei blendshape.

Aspettiamo il caricamento delle librerie, l'editor aprirà due frame, uno con il volto ricoperto della griglia e con in alto i valori di veridicità della condizione degli occhi, l'altro con tutti i 51 blendshape espressi con una barra orizzontale di colore blu che varia sempre tra 0 (minima lunghezza barra) ed 1 (massima lunghezza barra) **Fig. 8**.

Abbiamo impostato come soglia di veridicità 0.5, ma potete cambiarla a piacimento per rendere più o meno sensibile il sistema.

Ad ogni chiusura degli occhi, sia parziale che totale, verrà evidenziato con una freccia rossa quale valore lo ha generato nel frame del volto.

Per meglio vedere in funzione il programma, non abbiamo aggiunto tempi entro cui si dovrà verificare l'esattezza dei blendshape. Tale funzionalità potrete aggiungerla per non far suonare alla minima variazione l'allarme.

Dopo aver sperimentato sul proprio PC, possiamo rendere portatile il nostro progetto copiando il **Listato 2** su una scheda Raspberry Pi 5.

Monteremo il modulo buzzer attivo sul GPIO con appositi cavetti, facendo attenzione di inserire il suo PIN – (negativo) sul Ground (pin 6), ed il PIN 5 (segnale) sul GPIO 15 (pin 10). Il pin centrale del buzzer non serve. Tale modulo è acquistabile sullo store Futuranet con il codice KY012 **Fig. 9**.

Assicuratevi di avere installato la libreria *gpiozero*, una delle vie più semplici per la gestione dei pin GPIO di Raspberry Pi compatibile anche con la versione 5. Puoi installarla eseguendo il comando `apt install python3-gpiozero`.

Di seguito il codice da aggiungere al **Listato 2** per comandare il buzzer, iniziando con l'importazione della libreria:

```
from gpiozero import LED
```

Dovendo semplicemente mandare uno stato uno al buzzer, usiamo solo la funzionalità LED della libreria:

```
led = LED(15)
led.off()
```



Fig. 9
Modulo buzzer KY012.

Invece nella sezione che genera la freccia rossa di allarme, cioè la " Funzione per gestire gli allarmi " va aggiunto:

```
led.on()
else:
    led.off()
```

Nel **Listato 2** queste voci aggiuntive sono tutte commentate, quindi basta cancellare il #.

CONCLUSIONE

Questo progetto ha permesso di continuare lo studio delle funzionalità di MediaPipe, integrandole su una scheda Raspberry Pi per creare un sistema di rilevamento delle espressioni facciali, utilizzabile da noi maker in svariati contesti, ma dettagliatamente riferito alla chiusura degli occhi per generare un allarme sonoro (ad esempio stanchezza alla guida), affrontando quindi una funzionalità evoluta nel mondo dell'intelligenza artificiale. ■



COSA OCCORRE ?

Codice	Descrizione	Prezzo € IVA inclusa
RPI5-4GB	Raspberry Pi 5 versione con 4 GB di memoria	105,00
RPI5-8GB	Raspberry Pi 5 versione con 8 GB di memoria	136,00
27WSUPPLYEU	Alimentatore Raspberry Pi USB-C	24,00
KY012	Modulo buzzer	1,90

IL MATERIALE VA RICHIESTO A:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - www.futurashop.it

Oscilloscopio palmare 1 canale - 200 KHz



Prezzo IVA inclusa.

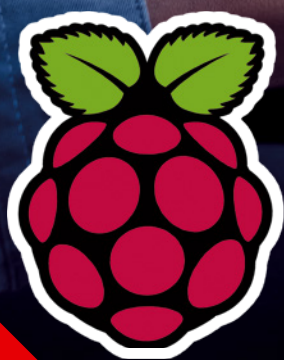
€ 79,00

cod. DSO200

Dimensioni compatte, solo 99 x 69 x 20 mm! Dispone di display LCD a colori da 2,8 pollici con risoluzione di 320 x 240 pixel. Il DSO200 è alimentato da una batteria al litio integrata da 1000 mAh, ricaricabile tramite la porta USB di tipo C (cavo incluso). Il supporto ribaltabile a 90 gradi facilita l'uso su superfici piane. Sonde e accessori inclusi nella confezione.

- › Frequenza di campionamento: 2,5 Msa/s (milioni di campioni al secondo)
- › Sensibilità verticale: da 10 mV/Div a 20 V/Div
- › Base dei tempi: da 10 μ s/Div a 50 s/Div
- › Range tensione di test:
X1: 80 Vpp (-40 a 40 V)
X10: 800 Vpp (-400 a 400 V)
- › Accoppiamento: AC e DC
- › Calibrazione (onda quadra):
Frequenza: 1 kHz
Ciclo di lavoro: 50%
- › Tipo di trigger: fronte in salita o discesa
- › Modalità di trigger: automatica, normale, singola misurazione
- › Peso: 100 grammi

ANDROID AUTO WIRELESS CON RASPBERRY Pi



di **DANIEL SCAINI**

Come trasformare un Raspberry Pi in un dongle wireless per Android Auto, offrendo un'integrazione avanzata tra smartphone e veicolo, migliorando navigazione, musica e messaggistica.

In un mondo sempre più connesso, anche il modo in cui interagiamo con le nostre automobili sta evolvendo rapidamente. Android Auto rappresenta una delle innovazioni più significative in questo campo, offrendo agli utenti la possibilità di portare l'esperienza del proprio smartphone direttamente sul cruscotto. Lanciato da Google, Android Auto è progettato per integrare le funzioni del telefono con il sistema di infotainment del veicolo, garantendo un accesso sicuro e intuitivo a navigazione, musica, messaggistica e molto altro, tutto tramite comandi vocali e un'interfaccia user-friendly. Con Android Auto, la guida diventa più sicura e piacevole, riducendo le distrazioni e mantenendo gli occhi sulla strada e le mani sul volante. In quest'articolo affrontiamo questa sfida: trasformare in modo ingegnoso un Raspberry Pi 4 in un dongle per Android Auto.

Questa soluzione non solo offre una maggiore flessibilità rispetto ai sistemi integrati di serie, ma dimostra anche come la creatività e la tecnologia possano convergere per migliorare la nostra vita quotidiana. L'adattatore Android Auto fatto con Raspberry Pi 4 permette di avere accesso a navigazione, musica, messaggistica e altre funzioni direttamente sul cruscotto, portando l'esperienza digitale su strada ad un nuovo livello. Scopriamo insieme come questa innovazione può trasformare il modo in cui interagiamo con i nostri veicoli (**Fig. 1**).

Nella parte conclusiva invece vediamo come poter ridurre le dimensioni e portare il tutto su un'altra piattaforma della famiglia Raspberry Pi Foundation: Raspberry Pi Zero W (**Fig. 2**).

HARDWARE

L'hardware per questo progetto è molto semplice e si suddivide in tre componenti principali.

La prima, che è anche la più importante, è la base su cui costruiremo l'intero sistema: il Raspberry Pi 4. Questo modello ci garantisce almeno 4GB di RAM e un sistema aggiornato, capace di supportare un'elaborazione audio e video avanzata.

Questo rappresenta la quarta generazione prodotta dalla Raspberry Foundation e non è solo adatta a chi vuole imparare a programmare, sviluppare e giocare, ma grazie a numerosi pin è particolarmente adatta anche per progetti di IOT complessi.

Dotata di un processore ARM, offre prestazioni paragonabili a quelle di un PC entry-level, ma è anche in grado di gestire flussi video in 4K.

Considerata ormai un caposaldo per chi si cimenta in questa tipologia di progetti offre numerosi dispositivi da poter abbinare come: moduli telecamera, sensori, GPS, display LCD, ecc..

Inoltre, rende disponibili numerosi pin, sia di alimentazione a 3.3v, sia per poter interfacciare periferiche, relay o sensori di diversa natura e gestibili tranquillamente a livello di codice Python o C++ attraverso un vasto set di librerie.

Come sistema operativo consigliamo Raspbian OS 64bit così da non aspettarsi sorprese durante l'installazione dei vari pacchetti successivamente.

A corredo di questa piattaforma, abbiamo previsto un microfono USB (cod. USBMICMINI), ottimale sia per il nostro sistema che per PC e MAC. Questo microfono offre una risposta in frequenza di 100-16 kHz e una sensibilità di -67 dBV/pBar, -47 dBV/Pascal ± 4 dB. Due dei principali vantaggi di questa scelta sono la soppressione dei rumori di fondo indesiderati e la semplice installazione, grazie alla funzionalità plug and play che elimina la necessità di driver esterni.

Infine, come terzo e ultimo componente, ma non meno importante, abbiamo previsto un paio di casse da PC o delle semplici cuffiette, che collegheremo al jack audio presente sul nostro Raspberry Pi.

Fig. 1
Il Raspberry Pi 4
diventa il dongle
per Adroid Auto.





Fig. 2
Riduciamo le dimensioni utilizzando un Raspberry Pi Zero W.

VIRTUALIZZAZIONE

Una parte fondamentale è svolta dalla virtualizzazione di una macchina per poter compilare correttamente l'immagine da inserire all'interno del Raspberry Pi.

Ma di preciso cosa significa virtualizzare?

La virtualizzazione è un processo che consente un utilizzo più efficiente dell'hardware del computer fisico ed è alla base del cloud computing, argomento molto dibattuto negli ultimi anni.

La virtualizzazione utilizza il software per creare un livello di astrazione sull'hardware del computer, che consente agli elementi hardware di un singolo computer - processori, memoria, storage e altro ancora - di essere suddivisi in più computer virtuali comunemente detti macchine virtuali (virtual machine, VM). Ogni macchina virtuale esegue il proprio sistema operativo e funziona come un computer indipendente, anche se è in esecuzione su una parte dell'hardware del computer effettivo. È una tecnologia che consente di creare rappresentazioni virtuali di server, archiviazione, reti e altre macchine fisiche estraniandosi dall'hw sottostante. Il software virtuale imita le funzioni dell'hardware fisico per eseguire più macchine contemporaneamente su un'unica unità fisica.

Il sistema operativo che ospita la vm è detto host, mentre invece quello virtualizzato viene chiamato guest.

La virtualizzazione è una tecnologia che sfrutta risorse tradizionalmente vincolate all'hardware. Consente di utilizzare tutte le capacità di una macchina fisica distribuendo le funzionalità tra utenti o ambienti diversi.

Esistono diversi tipi di virtualizzazione, tra cui:

- La virtualizzazione dei dati, che consente alle organizzazioni di consolidare le sorgenti di dati in un'unica fonte di erogazione dinamica.
- La virtualizzazione dei desktop, con cui più ambienti desktop simulati possono essere distribuiti e controllati tramite un amministratore centrale.
- La virtualizzazione dei server, che permette agli amministratori di suddividere i server in configurazioni con funzioni specifiche.
- La virtualizzazione dei sistemi operativi, grazie alla quale più sistemi operativi possono essere eseguiti su un unico computer.
- La virtualizzazione delle funzioni di rete, che separa le funzioni di una rete (come i servizi directory, la condivisione di file e la configurazione IP) in modo da poterle distribuire tra gli ambienti.

Fra i diversi vantaggi offerti invece si hanno:

- Efficienza delle risorse: prima della virtualizzazione, ogni server applicativo richiedeva la propria CPU fisica dedicata - il personale IT acquistava e configurava un server separato per ogni applica-

Fig. 3
La pagina di benvenuto di VMware Workstation Player.



zione che desiderava eseguire.

L'IT preferiva una singola applicazione e un singolo sistema operativo per ogni computer per motivi di affidabilità.

Invariabilmente, ogni server fisico veniva sottoutilizzato. La virtualizzazione del server, invece, ti consente di eseguire diverse applicazioni - ciascuna sulla sua VM con il suo sistema operativo - su un singolo computer fisico (di solito un server x86) senza andare a discapito dell'affidabilità. Ciò consente il massimo utilizzo della capacità di calcolo dell'hardware fisico.

- Gestione semplificata: la sostituzione dei computer fisici con le VM definite dal software semplifica l'utilizzo e la gestione delle politiche scritte nel software. Ciò ti consente di creare dei flussi di lavoro di gestione dei servizi IT automatizzati. Ad esempio, gli strumenti di implementazione e configurazione automatizzata consentono agli amministratori di definire delle raccolte di VM e applicazioni come servizi, in modelli software. Ciò significa che possono installare questi servizi ripetutamente e in modo coerente senza impostazioni manuali difficili, dispendiose in termini di tempo e soggette ad errori. Gli amministratori possono utilizzare le politiche di sicurezza della virtualizzazione per imporre determinate configurazioni della sicurezza in base al ruolo della VM. Le politiche possono anche aumentare l'efficienza delle risorse ritirando le VM inutilizzate per risparmiare spazio e potenza di calcolo.
- Tempi di inattività minimi: gli arresti anomali del sistema operativo e delle applicazioni possono causare tempi di inattività e interrompere la pro-

duttività degli utenti. Gli amministratori possono eseguire più VM ridondanti affiancate ed eseguire il failover tra di esse quando si verificano dei problemi. Eseguire più server fisici ridondanti è più costoso.

- Provisioning più rapido: l'acquisto, l'installazione e la configurazione dell'hardware per ogni applicazione richiedono molto tempo. A condizione che l'hardware sia già implementato, il provisioning di VM per eseguire tutte le tue applicazioni è molto più veloce. Puoi persino automatizzarlo utilizzando il software di gestione e integrarlo nei flussi di lavoro esistenti.

SISTEMA OPERATIVO

Sulla macchina guest per poter compilare correttamente l'archivio che andremo ad estrarre da git serve un sistema operativo. Nell'esempio proposto è stato scelto Ubuntu, grazie alla sua flessibilità, è il candidato perfetto offrendo anche diverse soluzioni per librerie necessarie alla compilazione. Rechiamoci quindi su www.ubuntu-it.org/download e scarichiamo la versione Ubuntu 22.04.4 LTS Desktop, per permettere a tutti, attraverso un'interfaccia grafica di poter interagire al meglio.

DOCKER

Prima di creare la Virtual Machine e cominciare ad utilizzarla è importante sottolineare che per le operazioni di compilazione vere e proprie ci sono due alternative molto differenti fra loro. Una prima manuale, esattamente come studiato sui libri di scuola, che prevede di installare sulla propria macchina tutte le librerie necessarie e quindi iniziare a compilare. In questo caso se una libreria viene dimenticata e non c'è un processo di check, come nel nostro caso, arrivati al punto in cui questa serve il processo di compilazione fallisce, bisogna sanare la falla data dalla libreria mancante e ripetere nuovamente il processo. Inoltre, se il pc avesse delle variabili di ambiente sporche o non di default si rischierebbe di incappare in enormi scocciature. Per ovviare a tutte queste problematiche c'è quindi una seconda modalità nella quale ci serviremo di un docker.

Cos'è un docker?

Quando si parla di una delle tecnologie più innovative nell'ambito del cloud computing, cioè dei container, che corrispondono a quel complesso di dati che occorrono a un'applicazione cloud nativa per essere eseguita, il riferimento a Docker è d'obbligo. Docker, infatti, rappresenta il software open source per la containerizzazione in ambiente Linux,

ma non solo, forse più noto a livello mondiale. Con lo stesso nome vengono indicati sia il progetto e la community impegnata nel miglioramento costante della soluzione, sia la società Docker Inc., comprata nel novembre 2019 da Mirantis Inc., azienda specializzata in servizi cloud B2B e nello sviluppo di Kubernetes e OpenStack. Un'acquisizione che segue, perciò, una logica ben precisa. Kubernetes è infatti la piattaforma di orchestrazione di container per antonomasia. OpenStack è un'infrastruttura cloud "container friendly", sebbene non sia l'unica. Docker tende a semplificare il ciclo di vita del software, isolando processi e applicazioni. Obiettivo è che la loro esecuzione indipendente sia in grado di ottimizzare le risorse dell'infrastruttura. In tal senso si pone in continuità, migliorandolo, con il paradigma SOA (Service-Oriented Architecture). Essa è una metodologia di scomposizione alternativa allo sviluppo monoblocco che ha anticipato l'avvento dei microservizi. Rispetto alla SOA e ad altre modalità di sviluppo, la novità che la porta Docker si concentra in particolare nell'affrancamento dalla virtualizzazione come standard univoco di testing e deployment. Basti pensare che all'epoca del suo rilascio, avvenuto nel marzo 2013, veniva spesso equiparato agli hypervisor delle macchine virtuali (VM), di cui condivide diversi benefici, ma non i relativi costi. Tanto che non è raro imbattersi ancora oggi in articoli e post che spiegano perché Docker non vada confuso con un sistema di virtualizzazione dell'hardware. Oltre a discostarsi dal meccanismo di astrazione tipico delle VM, superando i suoi potenziali sprechi di risorse di calcolo, il download di software Docker ha il duplice vantaggio della portabilità e della leggerezza. La portabilità deriva dal suo processo di distribuzione basato su file di immagine. Un processo che facilita la condivisione delle applicazioni e dei servizi tra più ambienti. Per quanto riguarda la leggerezza, questa dipende dalla capacità dei container di sfruttare il kernel del sistema operativo, eliminando l'overhead richiesto per l'esecuzione di ogni singolo kernel. Da qui la possibilità della macchina fisica di ospitare tantissimi container.

CREAZIONE DELLA VIRTUAL MACHINE

Questo capitolo possiamo dividerlo in due macro-fasi: una prima in cui creiamo l'istanza della macchina virtuale e una seconda in cui installiamo il sistema operativo al suo interno. Come client per virtualizzare la creazione dell'immagine da caricare su Raspberry Pi in questo

articolo utilizziamo VMWare Workstation Player. Apriamo quindi quest'ultimo e clicchiamo su **Crea a New Virtual Machine** per poter istanziare una nuova macchina come nello screenshot. (**Fig. 3**) Successivamente scegliamo la seconda opzione quindi **Installer disc image file (iso)** attraverso il tasto sfoglia e carichiamo la ISO scaricata durante uno dei passi precedenti.

Diamo quindi un nome alla nuova istanza e scegliamo un path per la locazione fisica dei file, nel caso in cui quello di default non vada bene. Solitamente in caso di PC con più HDD o più partizioni l'utente può scegliere una posizione personalizzata per non appesantire il disco su cui gira il sistema operativo.

Nell'esempio proposto il nome è AARpiDongleUbuntu e il path è stato lasciato di default.

L'ultimo step prima della creazione effettiva riguarda l'hdd della nuova macchina. Il consiglio è quello di non lesinare i GB: per la compilazione ne servono diversi, anche se il risultato è abbastanza leggero, negli step intermedi il suo peso cresce molto. Il software proporrà di default 20.0 GB che estendiamo a 80.0 GB in modo da poter anche permetterci più prove in caso di errori o problemi. In generale una build può impiegare fino a 20/25 GB.

Ultimo passo di questa prima fase è un riepilogo di tutto quello specificato nei passi precedenti (**Fig. 4**). Come da screenshot lasciamo almeno 4GB di RAM per poter compilare il tutto in un tempo accettabile. Se non fosse possibile o si vuole aumentare questo valore o altri, cliccare su **Customize Hardware**. Qui è possibile agire su Memoria, Processore o altri parametri riguardanti per l'appunto l'hardware

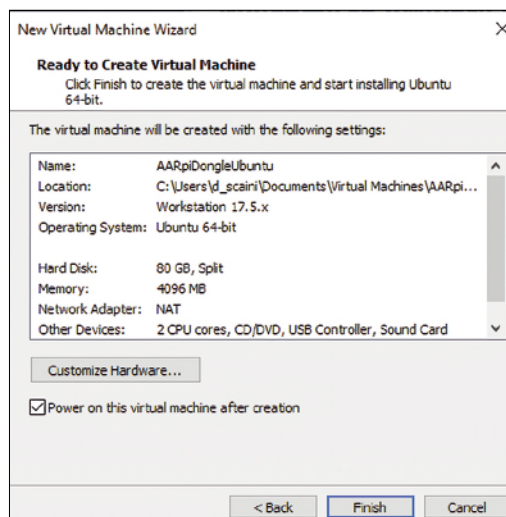


Fig. 4
Riepilogo delle impostazioni scelte.



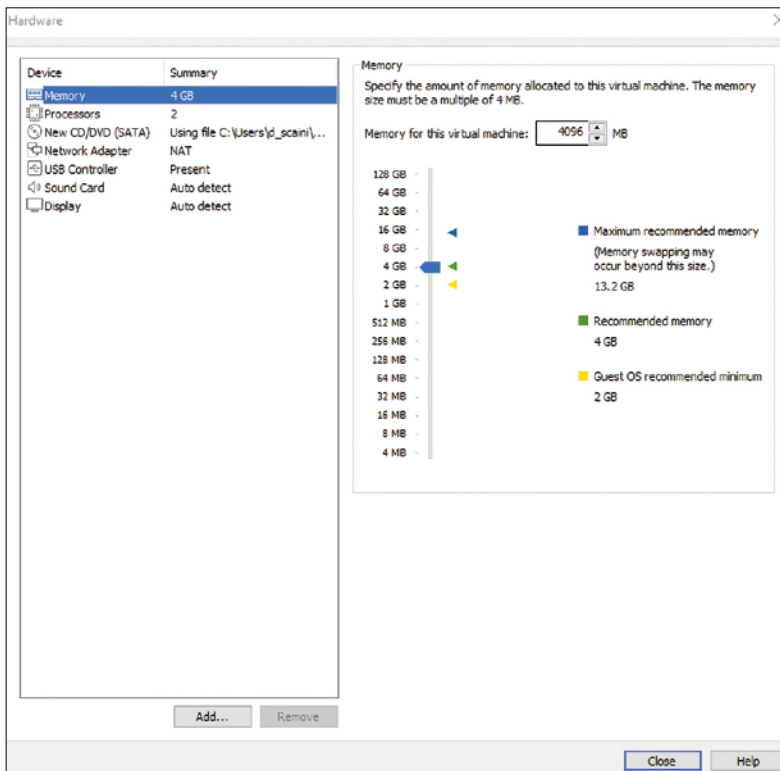


Fig. 5
Aumentando la memoria allocata, si velocizzano i tempi di compilazione.

attraverso il menu di sinistra (**Fig. 5**).
Cliccando su **Finish** potremo proseguire e iniziare ad usare la vm.

Qui inizia la seconda macro-fase: installazione del sistema operativo.

Questa è sicuramente più semplice e l'importante è seguire le istruzioni presenti a video.

In prima battuta scegliamo il layout della tastiera, consigliamo italiano, in modo tale che tutti i caratteri combacino con quelli a cui siamo abituati, e clicchiamo su Continue per rendere effettivo il tutto (**Fig. 6**).

Nella schermata successiva confermiamo subito il default e passiamo oltre dove ci verrà chiesto di inserire username e password. Qui possiamo crearla con un nostro username personalizzato oppure seguendo pedissequamente il nostro esempio. Noi abbiamo scelto *elettin* sia per il nome che per User e Password, per semplicità di utilizzo futuro (**Fig. 7**). Procediamo ora con **Install now** ed infine **Continue** ancora per poter chiudere la dialog nella quale si evidenzia il fatto che l'intero hdd sarà cancellato e ripartizionato.

Attendiamo ora alcuni istanti e otterremo la schermata nella quale clicchiamo su **Restart now** (**Fig. 8**).

Se abbiamo seguito alla lettera le indicazioni ad ogni login ci chiederà user e password che andremo ad inserire. Nel nostro caso riporteremo quindi *elettin, elettin*. A seguito del login, clicchiamo l'icona in basso a sinistra per poter accedere a tutte le applicazioni come nello screenshot (**Fig. 9**).

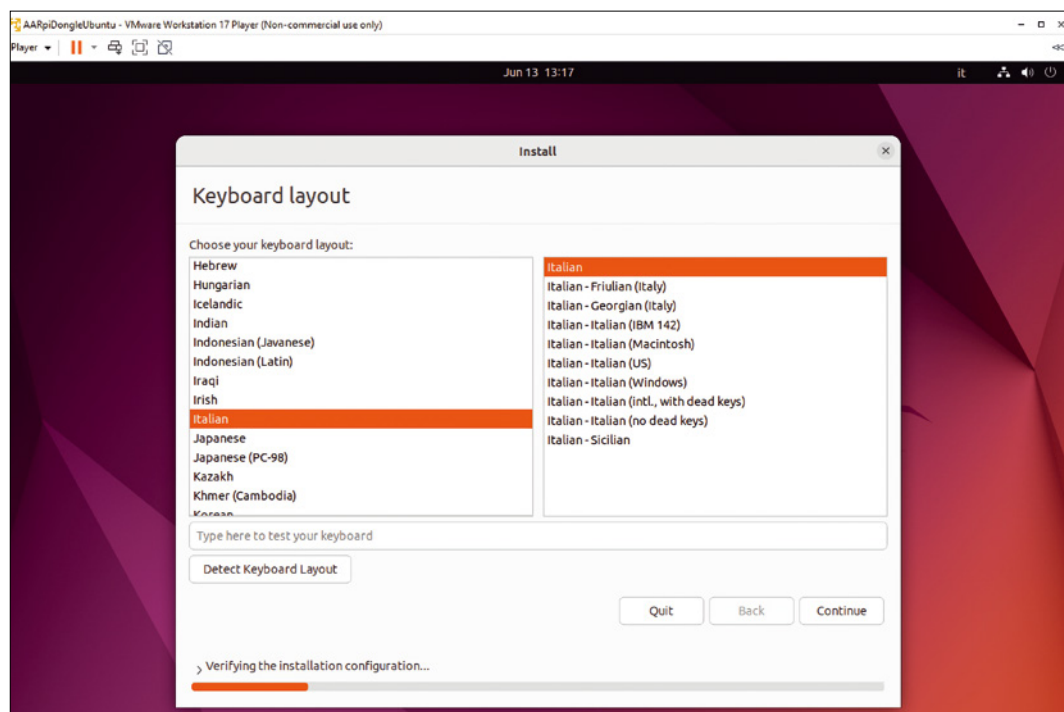


Fig. 6
Schermata di selezione del layout della tastiera.

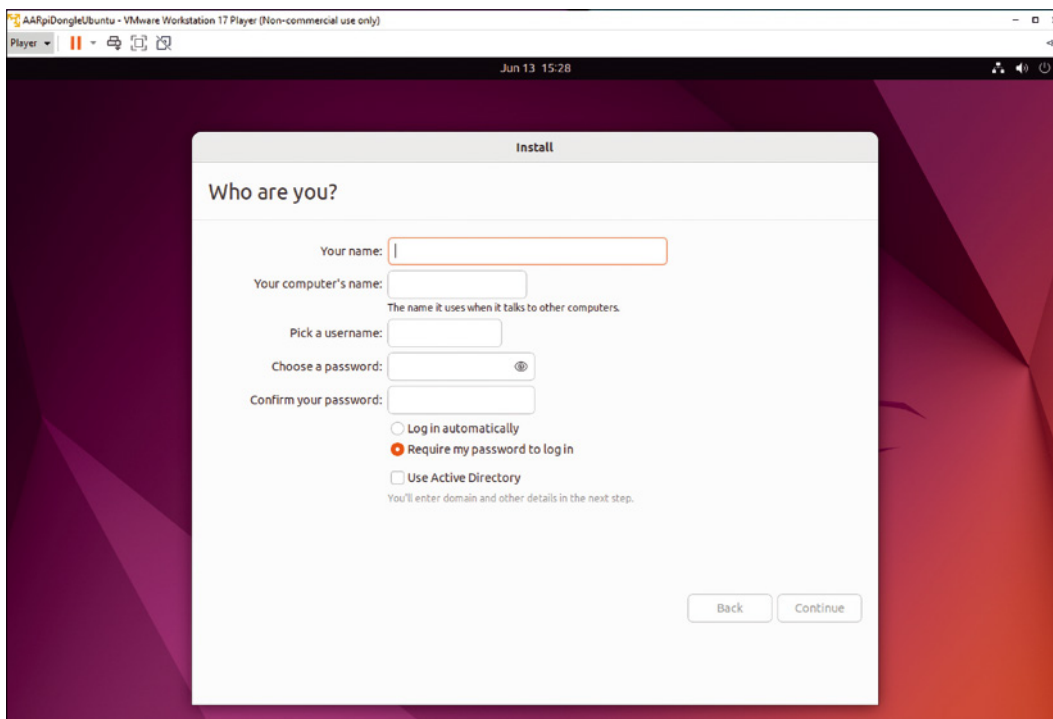


Fig. 7
Da questa schermata è possibile impostare un user e una password.

INSTALLAZIONE, REQUISITI E BUILD

Prima di parlare dei comandi veri e propri per poter compilare tutto il progetto facciamo una prima distinzione.

Ci sono infatti due modi per poter compilare cor-

rettamente, come già accennato: un primo manuale, esattamente come siamo sempre stati abituati e come troviamo in parecchi articoli didattici ed un secondo invece, più smart, attraverso l'avvio di un **docker**.

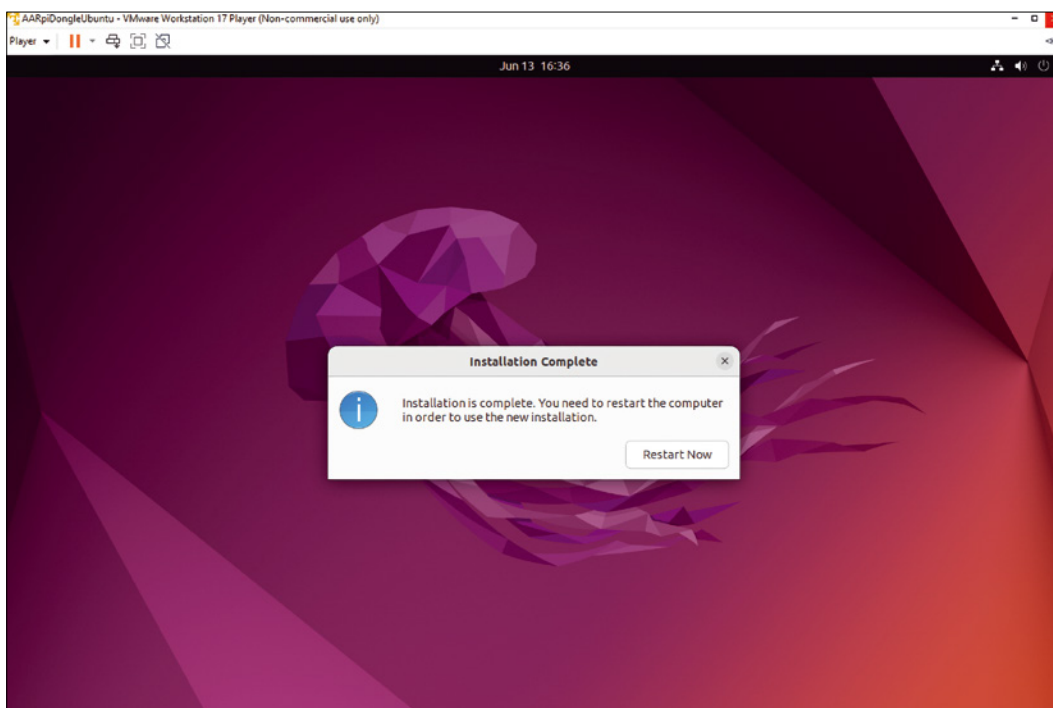
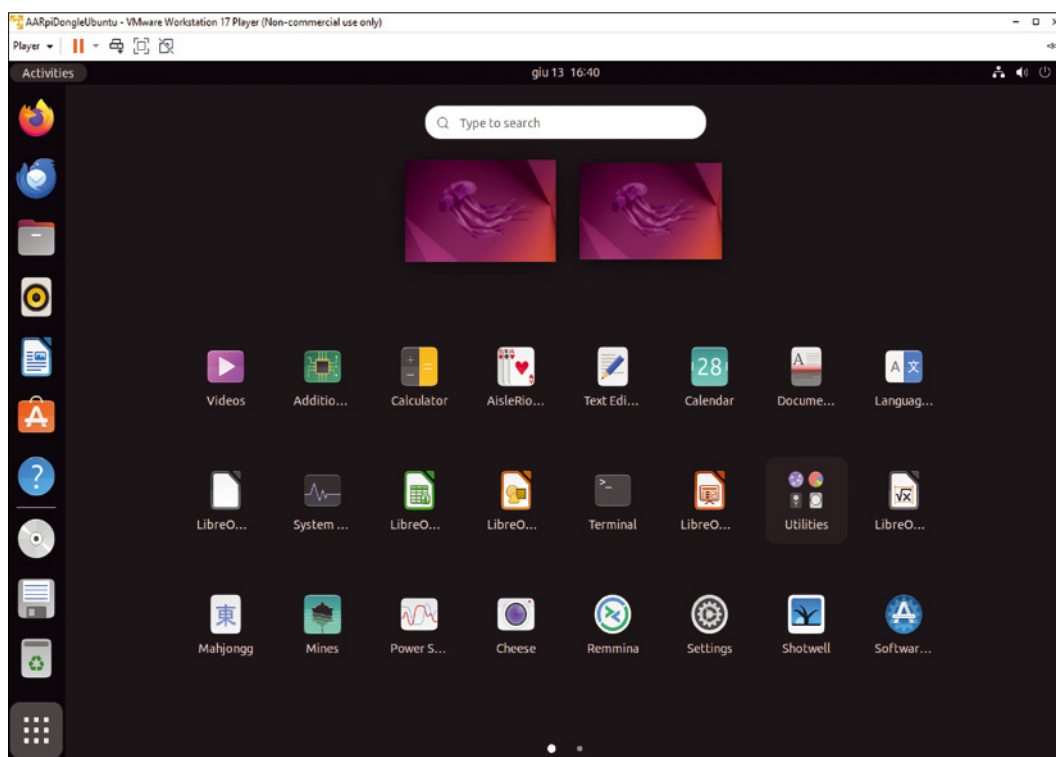


Fig. 8
A installazione completata, viene richiesto il riavvio.

Fig. 9
Dopo il login, vedremo le applicazioni installate.



Per entrambe, però, il primo passo è sicuramente quello di installare il pacchetto di **git** per poter recuperare l'archivio con tutto il progetto. Una volta avviata la macchina virtuale (VM) e completato gli step precedenti, apriamo quindi una console (*terminal*) e digitiamo:

```
sudo apt install git
```

Viene subito chiesta la password di root, di amministratore insomma, poiché stiamo installando un pacchetto attraverso sudo. Oltre alla password viene chiesta anche la conferma per l'installazione. Dopo aver installato correttamente il tutto andiamo a recuperare l'archivio attraverso l'istruzione:

```
git clone --recurse-submodules
https://github.com/nisargjhaveri/WirelessAndroid
AutoDongle
```

Il ramo di default su cui andiamo è quello scelto dall'autore dell'archivio, in questo caso: master, che è allineato con quelli che sono i rilasci precompilati di cui parleremo più avanti. Ora abbiamo tutto e possiamo cominciare a dividerci a seconda dell'alternativa scelta. Per procedere con il primo, le tre librerie fondamen-

tali sono rispettivamente *make*, *gcc* e *g++* attraverso:

```
sudo apt install make
sudo apt-get install gcc
sudo apt-get install g++
```

A questo punto partiamo con la fase di compilazione vera e propria. Per prima cosa ci rechiamo all'interno della cartella **buildroot**, dopodiché lanciamo il comando di make basato sul file interno già confezionato. L'utility make è un programma che di per se non fa parte del famoso GCC, ma è un program manager molto usato per gestire un gruppo di moduli di un programma, una raccolta di programmi o un sistema completo. Questa fornisce automaticamente gli strumenti per fare questi controlli e quindi, garantisce una compilazione esente da errori e solo di quei moduli di cui non esiste già il file oggetto aggiornato. Successivamente ci rechiamo nella cartella **rpi0w** all'interno di **output** e qui lanciamo la compilazione. Rispettivamente i comandi sono i seguenti:

```
cd buildroot
make BR2_EXTERNAL=../aa_wireless_dongle/ O=output/
rpi0w raspberrypi4_defconfig
cd output/rpi0w
make
```

Come potete notare questi comandi devono essere personalizzati a seconda di cosa vogliamo ottenere come output.

Durante il primo make, infatti, possiamo utilizzare le seguenti configurazioni:

- *raspberrypi0w_defconfig*, per poter compilare per Raspberry Pi Zero W
- *raspberrypizero2w_defconfig*, per poter compilare per Raspberry Pi Zero 2 W
- *raspberrypi3a_defconfig*, per poter compilare per Raspberry Pi 3A+
- *raspberrypi4_defconfig*, per poter compilare per Raspberry Pi 4

In realtà potremmo personalizzare anche la cartella di output mettendo per esempio **output/rpi4**, ma per semplicità non la modificheremo, essendo solo una label.

Dopo il comando di make possiamo tranquillamente andare a prendere un caffè poiché il processo sarà abbastanza lungo.

Il secondo approccio, invece, come già accennato, è sicuramente più smart ma le risorse in gioco sono le medesime.

Subito dopo aver recuperato l'archivio su git installiamo il pacchetto che concerne la gestione dei docker attraverso il comando:

```
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
sudo apt install docker-ce
```

Questo ci permetterà di poter eseguire correttamente il prossimo comando, per eseguire il docker:

```
docker compose run --rm rpi4
```

Anche qui, come nel primo metodo, possiamo scegliere l'output opportuno modificando l'ultima

parte del precedente comando.

Le configurazioni ammesse sono:

- *rpi0w*, per poter compilare per Raspberry Pi Zero W
- *rpi02w*, per poter compilare per Raspberry Pi Zero 2 W
- *rpi3a*, per poter compilare per Raspberry Pi 3A+
- *rpi4*, per poter compilare per Raspberry Pi 4

Come si può notare la fase preliminare richiede l'installazione di molte meno librerie poiché è tutto contenuto all'interno del nostro container. I tempi non variano molto fra i due metodi, si parla di circa 90/120 minuti, quello che può variare è l'affidabilità della compilazione e i possibili errori che potremmo riscontrare.

CREAZIONE CARD E PRIMA CONNESSIONE

L'immagine appena creata è ora all'interno della vm. Il primo passo è portarla fuori, all'interno del nostro OS host. Il metodo più veloce, senza l'utilizzo di cloud o soluzioni online, è senza dubbio quello di creare una cartella condivisa fra host e guest. Senza spegnere la macchina, rechiamoci in **Player**, in alto a sinistra, poi **Manage** ed infine nel sotto-menù premere **Virtual Machine Settings**, esattamente come nella **Fig. 10**.

Rechiamoci nel tab **Options** e andiamo su **Shared Folders**.

Qui abbiamo tre scelte: la prima ci preclude la possibilità di poter aggiungere o selezionare una cartella, mentre la seconda e la terza ci permetteranno di aggiungerne una.

A noi interessa averla a disposizione per un tempo limitato e il consiglio è quindi di cliccare sull'ultima opzione. Questa ad ogni riavvio della vm partirà con la cartella disabilitata.

Clicchiamo quindi su **Add** e dopo una prima schermata informativa ci chiede di inserire il percorso di una cartella del pc host accompagnata da un nome (**Fig. 11**).

Come si può notare dallo screenshot, per semplicità possiamo inserire direttamente il Desktop.

Infine, clicchiamo Ok per chiudere le impostazioni della macchina virtuale.

Tornando sulla nostra console andiamo ad eseguire il seguente comando per localizzare la nuova cartella condivisa:

```
ls -lh /mnt/hgfs/Desktop/
```

Queste cartelle, infatti, vengono mappate all'interno del percorso **/mnt/hgfs**.



Fig. 10
Creazione della
cartella condivisa.

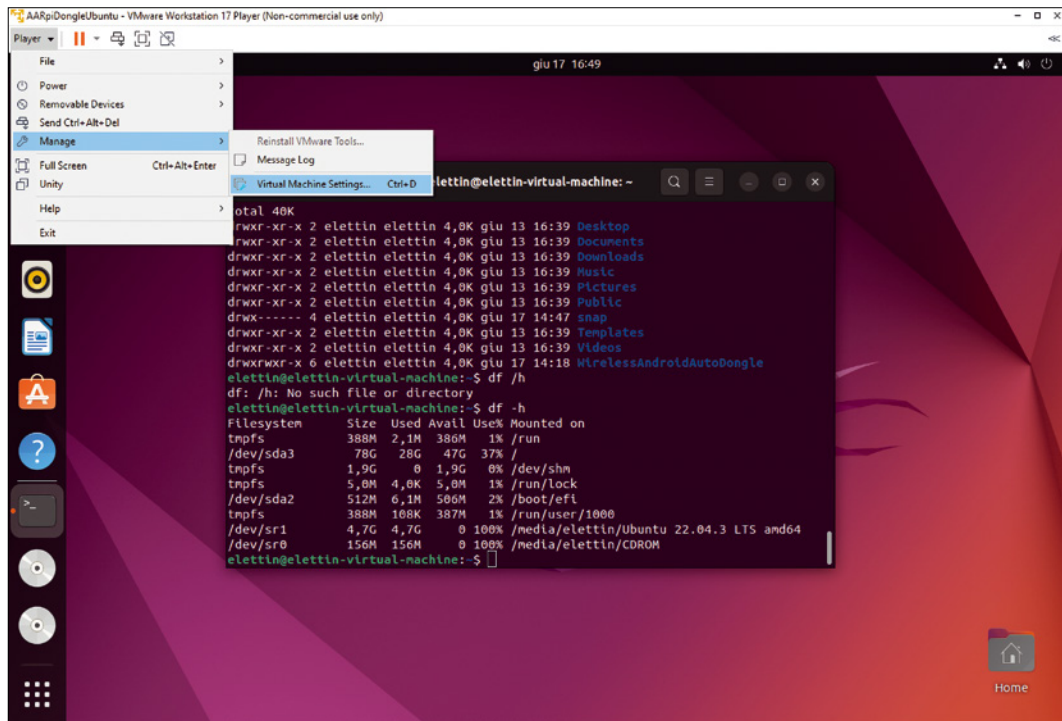
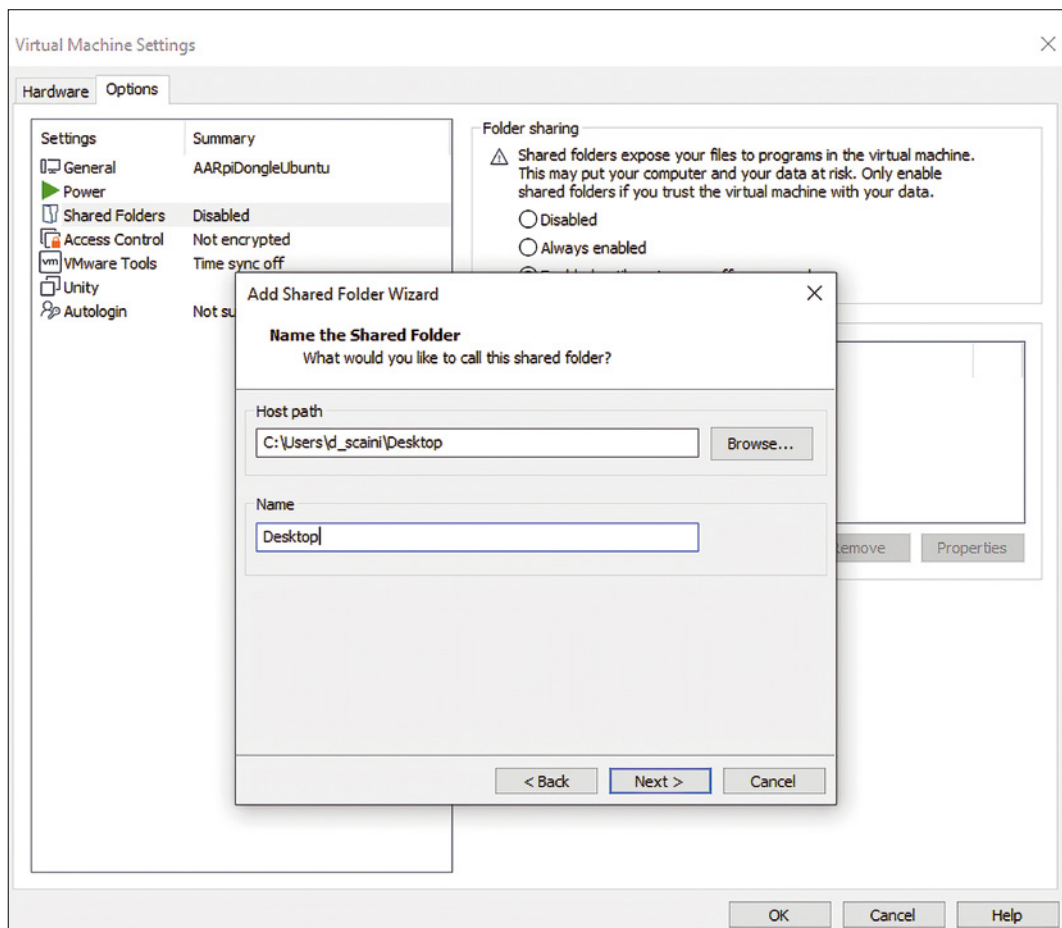


Fig. 11
Definizione del per-
corso della cartella
del pc host.



Ora copiamo il file img creato durante l'operazione di compilazione all'interno di questa nuova cartella e come per magia il file comparirà sul Desktop del pc ospitante. Nel caso di una compilazione manuale per procedere alla copia usiamo:

```
cp /home/elettin/WirelessAndroidAutoDongle/buildroot/output/rpi0w/images/sdcard.img /mnt/hgfs/Desktop/aarpi4_dongle.img
```

Ovviamente, nel primo percorso l'utente potrà variare nel caso in cui i passi non fossero stati seguiti alla lettera oppure nel caso l'installazione del SO fosse stata ulteriormente personalizzata. In generale il comando prevede:

```
cp percorso di origine [/home/UTENTE/ WirelessAndroidAutoDongle/buildroot/output/rpi0w/images/sdcard.img] percorso di destinazione [/mnt/hgfs/Desktop/aarpi4_dongle.img]
```

Mentre invece nel caso di una compilazione attraverso il docker usiamo:

```
cp /home/elettin/WirelessAndroidAutoDongle/images/sdcard-raspberrypi4.img /mnt/hgfs/Desktop/aarpi4_dongle.img
```

Anche qui il path potrebbe cambiare a seconda dell'Rpi scelto o dell'utente configurato. In ogni caso ora abbiamo a disposizione il nostro file sul desktop e possiamo quindi caricarlo all'interno del Raspberry Pi. In questa operazione ci viene incontro Raspberry Pi Imager, consigliato dalla referenza e dal sito ufficiale dell'azienda produttrice (**Fig. 12**). L'interfaccia è piuttosto semplice e già vista: come prima azione clicchiamo su Scegli dispositivo per poter individuare la versione del nostro Raspberry Pi, dopodiché scegliamo, attraverso il secondo tasto, il sistema operativo, selezionando **Use custom** e cliccando in seguito sulla nostra immagine, ed infine scegliamo il nostro supporto, ovvero l'sd card che conterrà la distro di AndroidAuto dongle. Premiamo avanti ed avviamo il processo di scrittura. Una volta terminato ci verrà chiesto di estrarre l'SD. A questo punto non ci resta che inserire la scheda all'interno del nostro Raspberry Pi e prepararci per il primo lancio.

Solo per il primo lancio, colleghiamo il nostro smartphone alla nostra auto attraverso il cavo USB. Assicuriamoci che parta Android Auto. A quel punto disconnettiamo e con lo stesso cavo

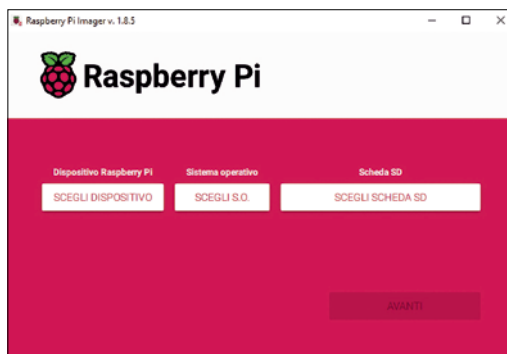


Fig. 12
Raspberry Pi Imager.

connettiamo il nostro Raspberry Pi. Utilizzando la versione 4 di questa scheda possiamo disinteressarci di abilitare la USB OTG poiché questa è già attiva di default. Ricordiamoci che l'USB OTG nel caso di Raspberry Pi 4 è la medesima dell'alimentazione, ovvero l'USB type C (**Fig. 13**).

Ora apriamo la schermata del Bluetooth sul nostro smartphone e lo colleghiamo alla periferica chiamata "AndroidAuto-Dongle" or "AA Wireless Dongle". Abilitiamo anche il Wi-Fi sempre dal nostro smartphone che verrà usato automaticamente da Android Auto. Ora dovremmo essere connessi senza un cablaggio diretto fra auto e smartphone.

VARIANTE CON RASPBERRY PI ZERO W

Come abbiamo già accennato precedentemente, solo il Raspberry Pi 4 ha l'USB OTG già abilitata di default, mentre per le altre versioni compatibili, quindi Zero W, Zero 2 W e 3A+, è necessario abilitarla.

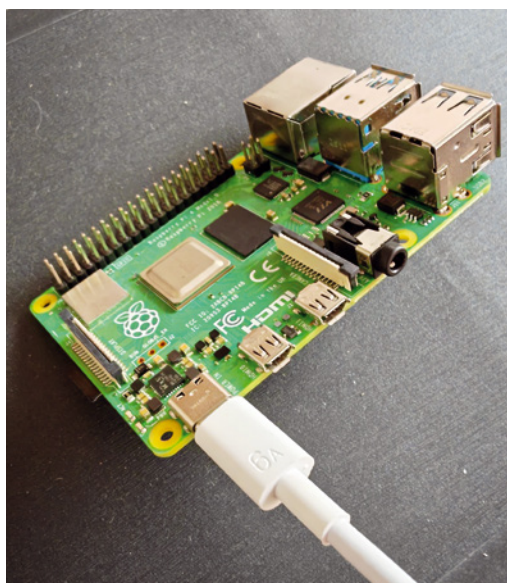


Fig. 13
Collegiamo Raspberry Pi alla porta USB dell'auto tramite il connettore di alimentazione.

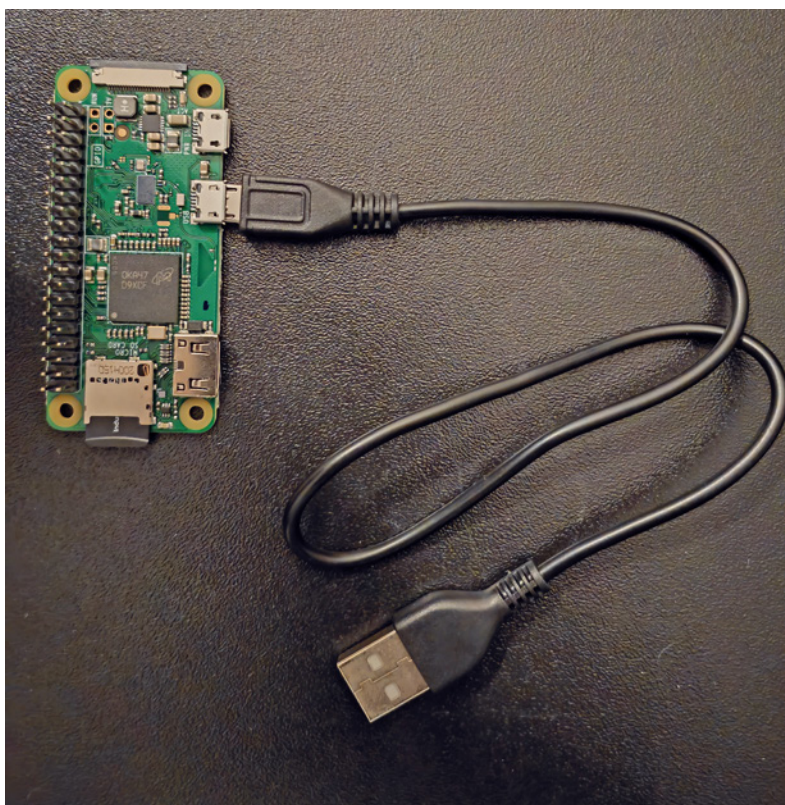


Fig. 14
Collegate
il cavo alla
presa chiamata
"USB".

Prendiamo per esempio, un po' per le dimensioni, un po' per il costo, la versione Zero W che troviamo disponibile nello store di Futura Elettronica con il codice RASPKITV9.

Il primo passo al fine di abilitare la scheda è inserire nuovamente l'SD all'interno del PC, se l'abbiamo già estratta come richiesto da Raspberry Pi Imager. Questa sarà vista come un'unità esterna, all'interno di questa troveremo un file chiamato **config.txt**.

Aggiungiamo quindi la seguente linea all'interno di quest'ultimo e salviamo il file:

```
dtoverlay=dwc2
```

Il secondo passo è creare un nuovo file sempre all'interno della directory principale, vuoto e senza estensione, chiamato **ssh**. Infine, ultimo step, inseriamo la seguente linea all'interno del file **cmdline.txt** dopo il termine **rootwait**:

```
modules-load=dwc2,g_ether
```

Salviamo quindi il file ed estraiamo l'SD dal nostro PC per inserirla all'interno del Raspberry Pi. Come si può notare, a differenza del Raspberry Pi 4, nel quale l'USB a cui collegarsi è la medesima dell'alimentazione, qui colleghiamo il cavo al connettore denominato USB, come nell'immagine in **Fig. 14**.

SVILUPPI FUTURI

Un ulteriore spunto per approfondire o personalizzare ulteriormente la nostra distribuzione, potrebbe essere per esempio cambiare il nome bluetooth del dispositivo.

In questo caso, poiché la modularità del codice ci offre un'enorme facilitazione, sarà molto semplice. Ci basta infatti editare il seguente file:

```
aa_wireless_dongle/package/aawg/src/bluetoothHandler.  
cpp
```

Tale modifica è possibile con l'editor **nano**, dovendo cambiare solamente una variabile.

In una delle prime righe c'è il nome dell'adattatore bluetooth che modifichiamo come segue:

```
static constexpr const char* ADAPTER_ALIAS = "AA  
ElettronicaIn Dongle";  
static constexpr const char* ADAPTER_ALIAS_DONGLE =  
"ElettronicaIn-Dongle";
```

Concludendo, possiamo dire che il progetto rappresenta un esempio di come la tecnologia e la creatività possano convergere per migliorare la nostra esperienza quotidiana. Utilizzando un Raspberry Pi, è possibile ottenere un dongle versatile ed economico che porta le funzionalità di Android Auto su qualsiasi veicolo. Leggendo questo articolo, anche chi ha competenze tecniche di base può realizzare questa soluzione innovativa, rendendo la guida più sicura e piacevole grazie all'accesso immediato a navigazione, musica, messaggi e altre applicazioni. ■



COSA OCCORRE ?

Codice	Descrizione	Prezzo € IVA inclusa
RPI4-2GB	Raspberry Pi 4 Tipo B con 2GB di memoria	83,00
RPI4-4GB	Raspberry Pi 4 Tipo B con 4GB di memoria	95,00
RPI4-8GB	Raspberry Pi 4 Tipo B con 8 GB di memoria	129,00
USBMICMINI	Mini microfono USB	3,50
RASPKITV9	Starter kit per Raspberry Pi Zero W	39,90

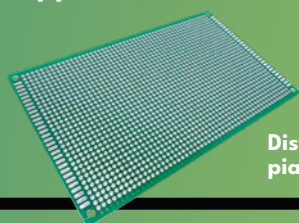
Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - www.futurashop.it

Piastre sperimentali

Ideali per creare velocemente un circuito sperimentale per effettuare test, senza dover necessariamente realizzare un PCB.

MILLEFORI

Piastra Millefori 150×90 mm
doppia faccia



cod. YK653

€ 4,50

Distanza tra
piazzole: 2,54 mm

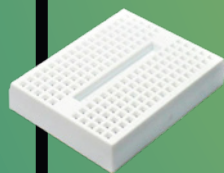
CODICE	MISURA (mm)	FACCIA	DISTANZA PIAZZOLE (mm)	PREZZO (€)
YK604	150×90	singola	2,54	3,80
YK647	100×50	singola	2,54	2,60
YK009	70×30	doppia	2,54	1,50
HR2570	120×80	doppia	-	4,50

BREADBOARD

Breadbord 170 contatti
colore bianco
45×35 mm

cod. BBMINIW

€ 2,50



● ROSSO

cod. BBMINIR

● NERO

cod. BBMININ

● BLU

cod. BBMINIB

● VERDE

cod. BBMINIV

Breadbord con base
autoadesiva 400 contatti
82×55 mm

cod. BB400

€ 4,00



Breadboard 1660 contatti
con base in metallo
e boccole

Misura piastra:
164×109×9,5 mm

Misura base:
215×131×1 mm

cod. YJ319

€ 14,00



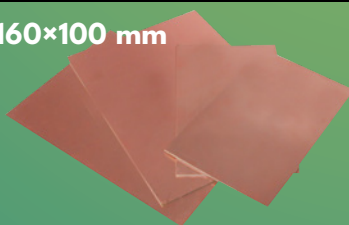
COD.	MISURA (mm)	N. CONT.	PREZZO (€)	COD.	MISURA PIASTRA (mm)	MISURA BASE (mm)	N. CONT.	PREZZO (€)
BB830	165×55	830	5,50	BB3220	188×182×9,5	240×210	3220	21,00

RAMATE

Piastra Ramata 160×100 mm
singola faccia

cod. P-951

€ 3,00



CODICE	MISURA (mm)	FACCIA	PREZZO (€)
P-952	200×150	singola	6,00
P-956	160×100	doppia	5,40
P-957	200×150	doppia	6,20

PRESENSIBILIZZATE

Piastra Presensibilizzata
160×100 mm singola faccia

cod. 4062

€ 6,00



CODICE	MISURA (mm)	FACCIA	PREZZO (€)
4072	160×100	doppia	7,80
P-995	233×160	singola	13,50
P-997	233×160	doppia	17,00

Prezzi IVA inclusa.

Impara facilmente l'elettronica!

EDUKIT 01

11

**FACILI PROGETTI
PER IMPARARE
L'ELETTRONICA
DI BASE**

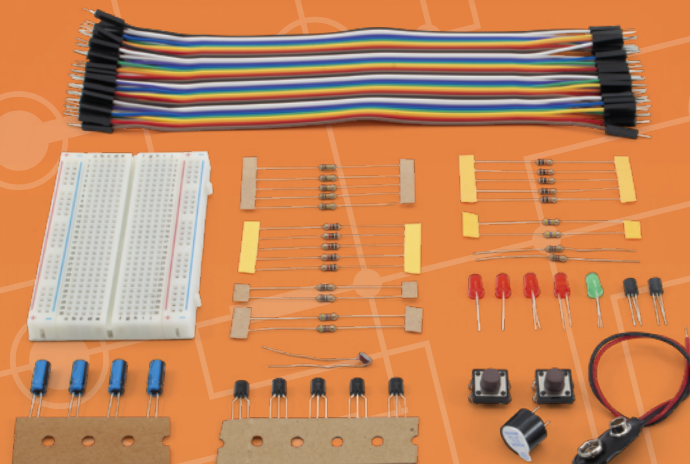
Kit di 11 esperimenti di elettronica base

Divertente, sicuro ed educativo, non richiede alcuna saldatura. Il kit contiene una breadboard 400 contatti con base autoadesiva e una serie di componenti per realizzare 11 progetti di elettronica di base. Tra i progetti: LED lampeggianti, sensore di luce, tester di polarità, circuito di start-stop, timer e molti altri.



€ 19,90

cod. EDUKIT01



Kit di 10 esperimenti con energia solare

Avvicinati facilmente al mondo dell'energia solare apprendendo e mettendo in pratica semplici circuiti. Per la realizzazione dei 10 progetti non serve alcuna attrezzatura, nemmeno il saldatore, in quanto il sistema a breadboard permette ogni genere di collegamento elettrico tra i componenti forniti, mediante l'innesco degli stessi nella piastra; basta seguire le immagini presenti nel manuale in dotazione corredato di schemi elettrici e di descrizione tecnica.

EDUKIT 02

10

**ESPERIMENTI
CON DIFFERENTI
GRADI DI
DIFFICOLTÀ**



€ 17,90

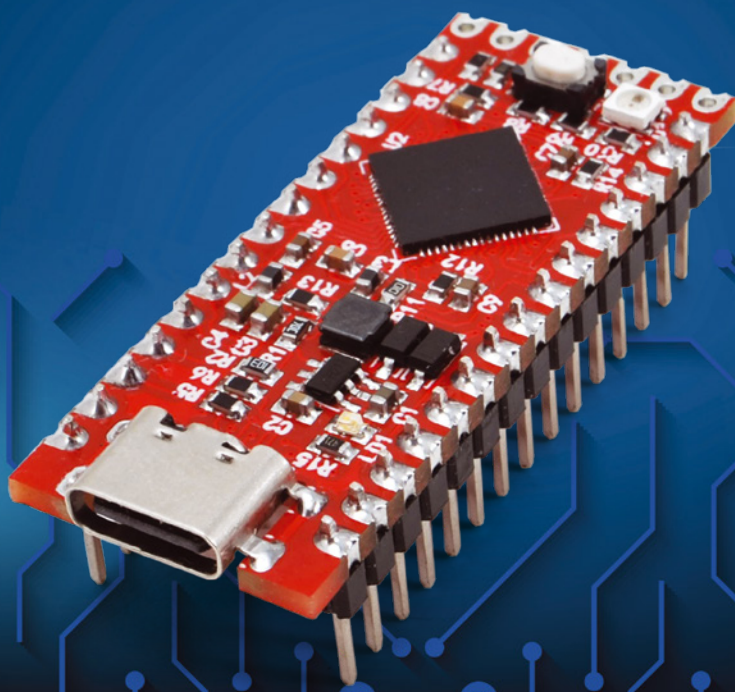
cod. EDUKIT02



Prezzi IVA inclusa.

NANO R4:

GRANDE TECNOLOGIA IN PICCOLO FORMATO



di **PIER CALDERAN**

La board NANO R4 sviluppata da Futura Elettronica è una versione compatta della scheda Arduino UNO R4 minima. Mantiene lo stesso processore e le principali caratteristiche.

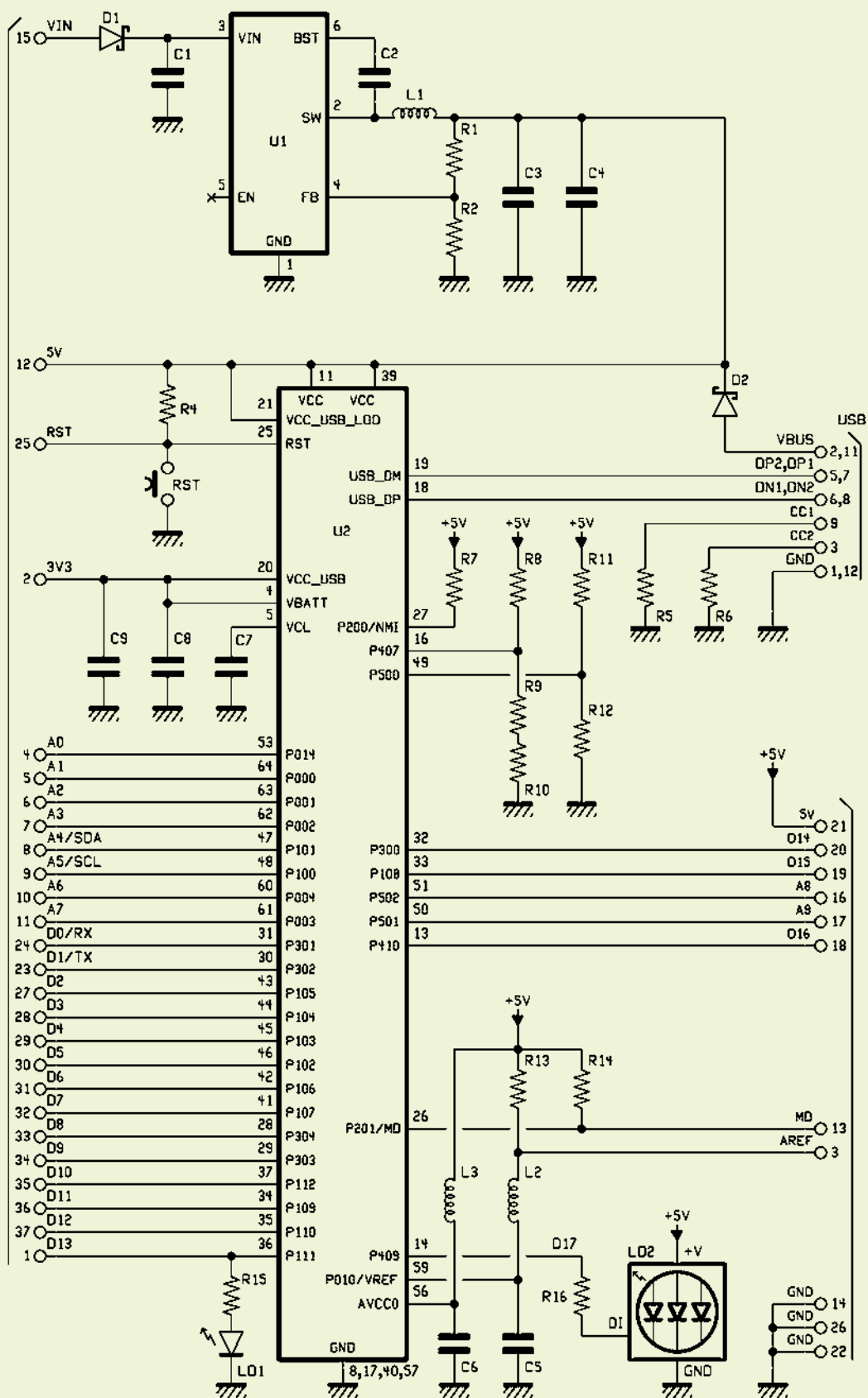
N

el giugno 2023, Arduino ha rilasciato un importante aggiornamento della sua scheda Arduino UNO Rev 3. Sebbene i microcontrollori a 32 bit non siano una novità per Arduino, questa è la prima scheda UNO ufficiale a includerne uno: l'Arduino UNO R4 utilizza infatti il chip RA4M1 di Renesas.

Arduino distribuisce Arduino UNO R4 in due versioni: MINIMA (**Fig. 1**) e WIFI (**Fig. 2**).

Entrambi montano il chip Renesas RA4M1, che è un Arm Cortex-M4 a 32 bit. Le schede rispettano in tutto e per tutto lo stesso fattore di forma del "vecchio" Arduino UNO Rev. 3, la stessa piedinatura con la stessa designazione e disposizione dei pin di ingresso e uscita.

Nella tabella riportata nella pagina successiva, sono indicate



le caratteristiche comuni alle due versioni: la versione WIFI è identica alla MINIMA, con l'aggiunta di un modulo WiFi ESP32-S3, una matrice di LED 8 x 12 e un connettore Qwiic.

NANO R4

La mancata introduzione di una versione NANO da parte di Arduino ha spinto il nostro team di progettisti a sviluppare una scheda NANO R4 (**Fig. 3**), perfettamente compatibile con la nuova piattaforma Renesas a 32 bit.

Il modello di riferimento è Arduino UNO R4 MINIMA, con alcune sostanziali differenze. Abbiamo aggiunto un LED neopixel, utile per poter visualizzare vari stati della scheda con una selezione di colori RGB, e abbiamo esposto alcuni pin “nascosti” che sono disponibili nel chip Renesas.

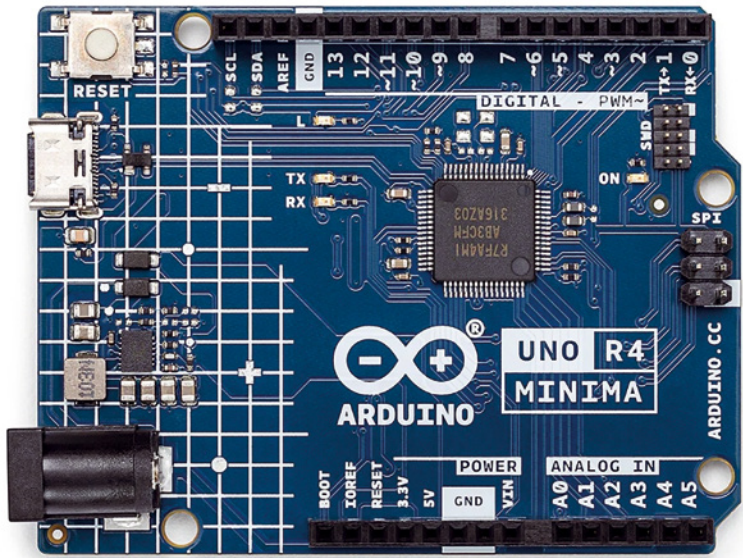
La piedinatura è del tutto compatibile con i modelli Arduino NANO, come si può vedere in **Fig. 3** e **4**.

Per rendere la scheda NANO R4 visibile nel menù schede dell’interfaccia IDE di Arduino, è necessario installare il pacchetto “futura”, scaricabile dalla pagina futuranet.it/download/pacchetto-futura-per-la-board-nano-r4 nella cartella hardware dell’IDE. Per fare questo scompattare lo zip scaricato e copiare la cartella “futura” in “C:\Utenti\AppData\Local\Arduino15\packages”.

All’avvio dell’IDE di Arduino, la scheda sarà visibile nel menu Strumenti, nel sottomenu Scheda, alla voce Futura Group Boards.

Quindi, selezionare la voce Futura NANO R4

CARATTERISTICHE COMUNI UNO R4 MINIMA E UNO R4 WIFI	
Pin I/O digitali	14
Pin di ingresso analogici	6
PWM	6
DAC	1x
UART	1x
i2c	1x
SPI	1x
CAN Bus	1x
Tensione del circuito	5 Volt (alimentazione con presa USB-C)
Tensione di ingresso (VIN)	6-24 Volt
Corrente CC per pin I/O	8mA
Velocità di clock	48MHz
Memoria	RA4M1 256 kB di Flash, 32 kB di RAM, 6kB EEPROM



Minima e la porta seriale che viene installata dalla connessione USB (**Fig. 5**).

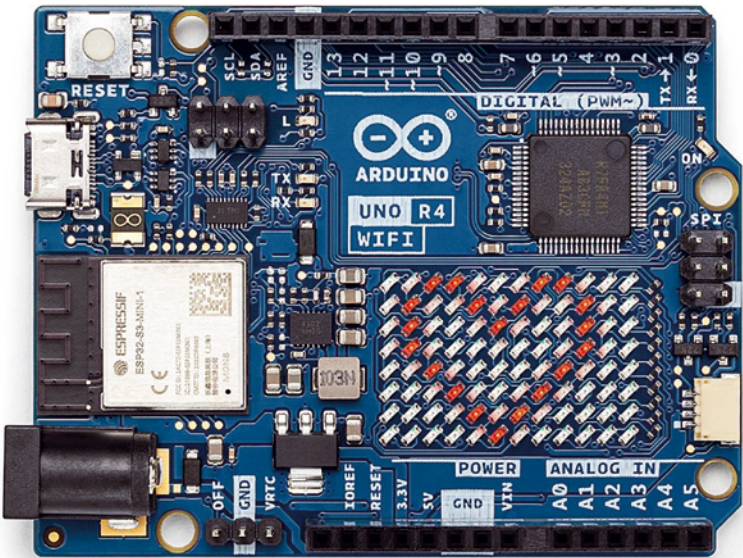
SCHEMA ELETTRICO

Lo schema elettrico della board Nano R4 evidenzia l’integrazione del microcontrollore Renesas RA4M1, simile alla versione UNO R4 MINIMA, ma in un formato compatibile con Arduino Nano.

Una caratteristica molto interessante della Nano R4 è la presenza dei castellated holes, noti anche come plated half-holes. Queste aperture semicircolari, placcate con materiale conduttivo, si trovano lungo i bordi del PCB e facilitano la connessione

Fig. 1
La versione
Arduino UNO
R4 MINIMA.

Fig. 2
La versione
Arduino UNO
R4 WIFI.

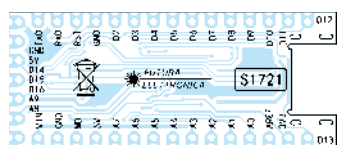
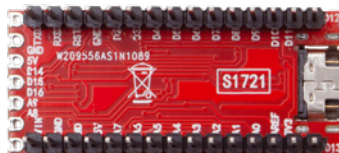
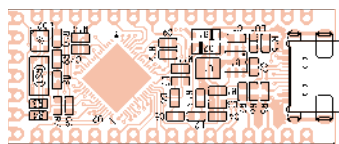
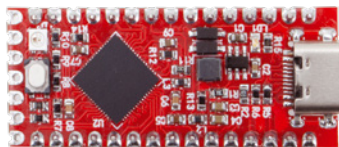


Elenco Componenti:

R1: 52,3 kohm 0,1% (0603)
 R2: 10 kohm 0,1% (0603)
 R15: 330 ohm (0603)
 R4, R5, R6, R7, R8, R9, R10, R13, R14:
 5,1 kohm (0603)
 R11: 100 kohm 0,1% (0603)
 R12: 13,7 kohm 0,1% (0603)
 R3, R15: 470 ohm (0603)
 C1 10: μ F 16 VL ceramico (0603)
 C3, C4: 22 μ F 6.3 VL ceramico (0603)
 C2, C6, C8: 100 nF ceramico (0603)
 C5: 1 μ F ceramico (0603)
 C7, C9: 4,7 μ F 6.3 VL ceramico (0603)
 L1: 22 μ H / 1 A - 540 mohm
 (3x3 mm)
 L2, L3: BLM18PG471SN1D 0603
 LD1: LED verde (0603)
 LD2: LED Neopixel WS2812 (3535)
 D1, D2: MBR0520LT1G (SOD123)
 1 RST PTS810SJK250SMTRLFS
 1 U1 AP62250WU-7
 1 U2 R7FA4M1AB3CNB#AA0
 1 USB Molex USB connector
 2169900002

Varie:

- Circuito Stampato S1721
 (44x18 mm)



elettrica e meccanica tra la Nano R4 e la scheda che la ospita, senza la necessità di connettori aggiuntivi. Questo permette di realizzare progetti compatti, ideali per design ad alta densità. Per chi invece preferisce un design modulare, che consenta di rimuovere la Nano R4 facilmente, è possibile saldare degli strip, mantenendo la piena compatibilità con le versioni precedenti di Arduino Nano. Il design elettrico include collegamenti per il controllo di un LED Neopixel RGB, aggiunto per offrire una visualizzazione di stato versatile tramite variazione di colori. Lo schema mette inoltre in evidenza pin di I/O digitali e analogici aggiuntivi, che espandono la versatilità della scheda rispetto alle classiche Nano. Questi pin extra permettono un accesso più completo alle funzionalità del chip, utili per progetti che richiedono un numero maggiore di input e output o per sfruttare moduli aggiuntivi tramite protocolli come I2C e SPI.

Test di esempio

Per provare se tutto funziona è possibile eseguire il consueto esempio "Blink" dell'IDE di Arduino, collegando un LED al pin 13 della scheda. Ma avendo la nostra scheda un LED neopixel RGB incorporato,

vi invitiamo a provare il nostro semplice sketch di esempio per illustrarne le potenzialità.

Nel **Listato 1** potete vedere tutto il codice commentato. All'inizio dello sketch, viene inclusa la libreria FastLED.

Si tratta di un libreria già incorporata nell'installazione dell'hardware della nostra scheda.

Questa libreria funziona con i LED neopixel, al contrario della libreria originale Adafruit che risulta incompatibile con le nuove versioni di Arduino UNO R4. Nello sketch di esempio viene definito il numero di LED con la variabile NUM_LEDS 1, ovvero 1 LED. Viene inoltre definito il pin di controllo con la variabile DATA_PIN 17, che è il pin che abbiamo scelto al di fuori dei pin convenzionali. Con CRGB leds[NUM_LEDS] si definisce un array per contenere i dati del colore dei LED.

Nel setup, si inizializza la matrice con la funzione

```
FastLED.addLeds<NEOPIXEL, DATA_PIN>(leds, NUM_LEDS);
```

La luminosità viene impostata con FastLED.setBrightness(20).

All'interno del loop vengono richiamate le istruzioni per accendere il led di blu, rosso, verde e infine bianco. Per spegnere i LED, si utilizza il colore nero (Black), mentre per accenderli tutti si usa il colore bianco (White).

Di seguito, vediamo nel dettaglio il loop del programma:

```
leds[dot] = CRGB::Blue; // LED blu
FastLED.show(); // Aggiorna i LED
leds[dot] = CRGB::Black; // Spegni il LED
delay(500); // Attende 500ms
leds[dot] = CRGB::Red; // LED rosso
FastLED.show(); // Aggiorna i LED
leds[dot] = CRGB::Black; // Spegni il LED
delay(500); // Attende 500ms
leds[dot] = CRGB::Green; // LED verde
FastLED.show(); // Aggiorna i LED
leds[dot] = CRGB::Black; // Spegni il LED
delay(500); // Attende 500ms
leds[dot] = CRGB::White; // LED su bianco
FastLED.show(); // Aggiorna i LED
leds[dot] = CRGB::Black; // Spegni il LED
delay(500); // Attende 500ms
```

Si fa notare che con questa libreria si può utilizzare un qualsiasi LED neopixel, come per esempio, i ring neopixel a 16 o 24 LED, o strisce di LED. Basterà cambiare il numero del pin di controllo, compreso fra 2 e 16 (non il 17) e il numero di LED della catena.

A questo scopo potete provare la libreria FastLED scegliendo un esempio dal suo menu (Fig. 6).

COMPATIBILITÀ DELLE LIBRERIE CON UNO R4

L'elenco di librerie per Arduino UNO R3 è a dir poco enorme e il passaggio a un nuovo chip a 32 bit ha introdotto un problema di compatibilità; principalmente per le straordinarie librerie sviluppate nel corso degli anni, in particolare attorno al chip ATmega328P di Arduino UNO R3.

Lo stato attuale della compatibilità è riportato a questo indirizzo:

<https://github.com/arduino/uno-r4-library-compatibility#tested-and-confirmed-libraries>.

Si tratta del GitHub ufficiale di Arduino riservato a tutti coloro che desiderano partecipare al test e/o al porting delle librerie sulla nuova piattaforma a 32 bit. Qui gli sviluppatori possono presentare i loro progetti di porting e/o segnalare eventuali incompatibilità.

Visitando il repository, si vedrà l'elenco (incompleto) di librerie testate che è stato suddiviso per categoria e mostra il livello di compatibilità con un segno di spunta verde o un simbolo di X per quanto riguarda la compilazione e il test hardware (Fig. 6). Questo significa che la libreria a volte sembra funzionare perché viene compilata, ma non funziona a livello hardware.

Ogni libreria ha un link alla pagina di riferimento che comunque consigliamo di visitare. È capitato però di trovare un aggiornamento che rende compatibile la libreria nonostante fosse contrassegnata

come non funzionante.

Non potendo in questa sede verificare tutte le librerie, abbiamo cercato di sopperire alla incompatibilità di alcune librerie che riteniamo fondamentali.

A questo scopo abbiamo cercato librerie alternative e in alcuni casi siamo stati fortunati.

Pertanto, nonostante alcune librerie della lista non siano compatibili abbiamo trovato alternative valide per alcune librerie di fondamentale importanza. Bisognerà attendere che gli sviluppatori della community facciano il lavoro di porting delle librerie incompatibili, rendendo così un servizio alla community di inestimabile valore (come accade di solito).

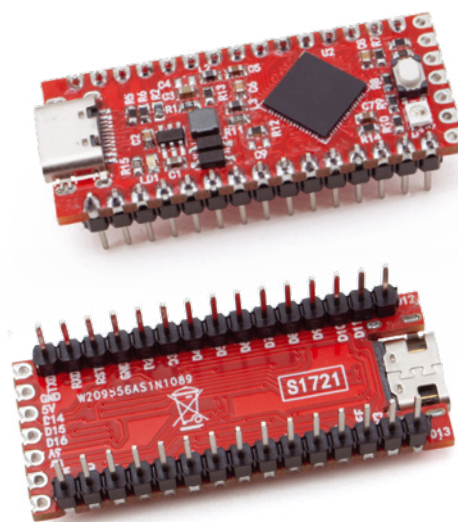


Fig. 3
Vista superiore del nostro NANO R4.

Fig. 4
Le serigrafie sul lato inferiore del nostro NANO R4.

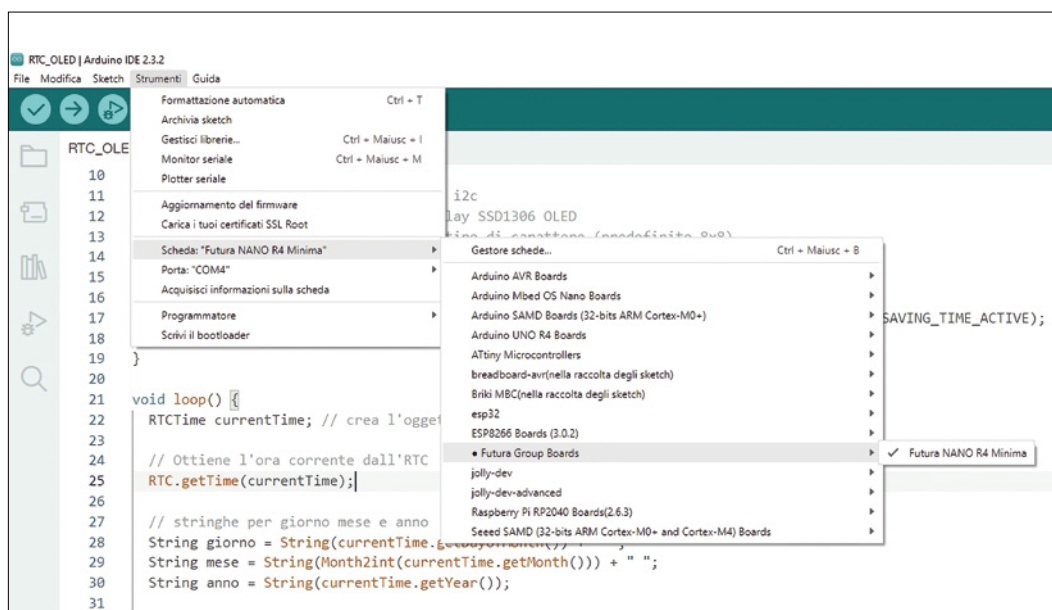
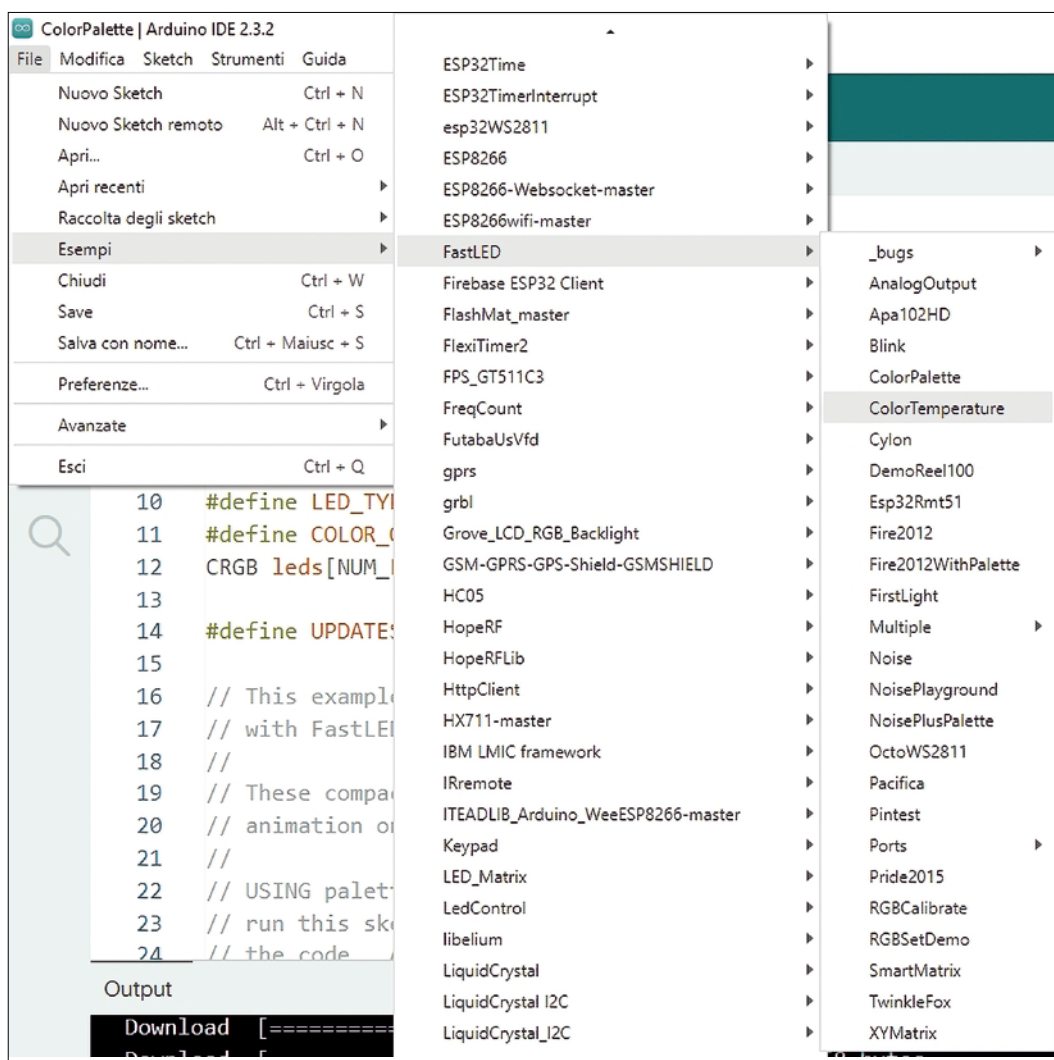
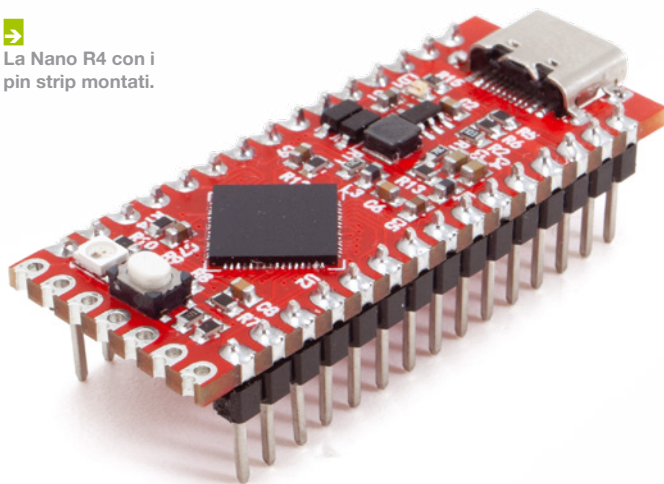


Fig. 5
La scheda Futura NANO R4 Minima nel menu Strumenti.

Fig. 6
Il menu di esempi
della libreria
FastLed.



La Nano R4 con i
pin strip montati.



LIBRERIA PER DISPLAY OLED I2C

Una delle funzionalità più importanti di una scheda è senza dubbio quella di poter comunicare con la porta i2c e soprattutto con un display OLED. Purtroppo la nota libreria **Adafruit SSD1306** per display OLED I2C e l'omonimo chip, al momento in cui scriviamo risulta incompatibile a livello hardware, ovvero mostra pixel sparsi sul display, rendendo tutto illeggibile.

Essendo una libreria che fa funzionare tutti i display OLED basati su chip SSD1306, abbiamo trovato un'ottima alternativa, che utilizzeremo per i nostri progetti OLED più avanti.

Il GitHub di riferimento è il seguente:

https://github.com/datasith/Ai_ArduLib_SSD1306

Per installare la libreria seguire la consueta procedura per installarla da file zip.

Una volta installata, sarà disponibile come

Tested and Confirmed Libraries

This list contains the status of tested Arduino libraries with the UNO R4 board. To change the list, submit a pull request to this repository.

Robotics

No.	Library	Compilation	Hardware Test	Notes
1.	Servo	PASS ✓	PASS ✓	
2.	Stepper	PASS ✓	PASS ✓	
3.	AccelStepper	PASS ✓	FAIL ✗	
4.	MobaTools	PASS ✓	PASS ✓	

Sensors

No.	Library	Compilation	Hardware Test	Notes
1.	Adafruit Neopixel	PASS ✓	FAIL ✗	
2.	Adafruit DHTSensor	PASS ✓	PASS ✓	
3.	IRremote	FAIL ✗	FAIL ✗	
4.	Grove_BMP280	PASS ✓	PASS ✓	
5.	Seeed_Arduino_LIS3DHTR	PASS ✓	PASS ✓	

Fig. 7
Il GitHub
con l'elenco
delle librerie
compatibili.

ACROBOTIC_SSD1306.h, da includere in qualsiasi progetto che preveda l'utilizzo di un display OLED. La libreria offre esempi di codice per l'utilizzo della nostra scheda NANO R4 e di un display OLED da 128x64 pixel (**Fig. 8**).

TEMPERATURA E UMIDITÀ SU DISPLAY OLED

Per rendere più interessante l'uso di un display OLED, abbiamo bisogno di inserire in una breadboard la

nostra scheda, un sensore di temperatura e umidità come il noto DHT11 e un qualsiasi display OLED SSD1306 con risoluzione 128 x 64 pixel.

Per libreria DHT11 consigliamo di installare DHTstable di Rob Tillaart, che consideriamo la più stabile, come dice il nome stesso.

Il GitHub di riferimento è il seguente:

<https://github.com/RobTillaart/DHTstable>

Si fa notare che per la porta I2C, il pin A4 viene

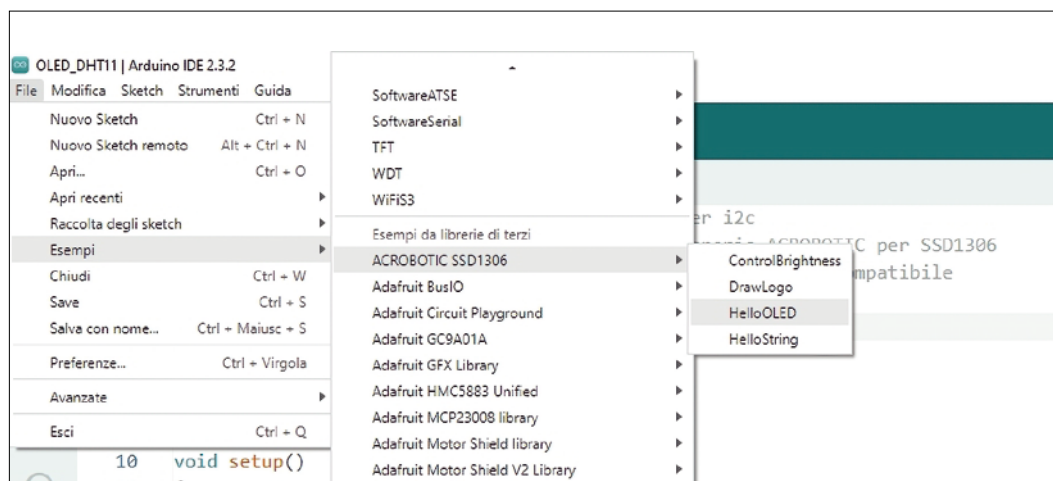
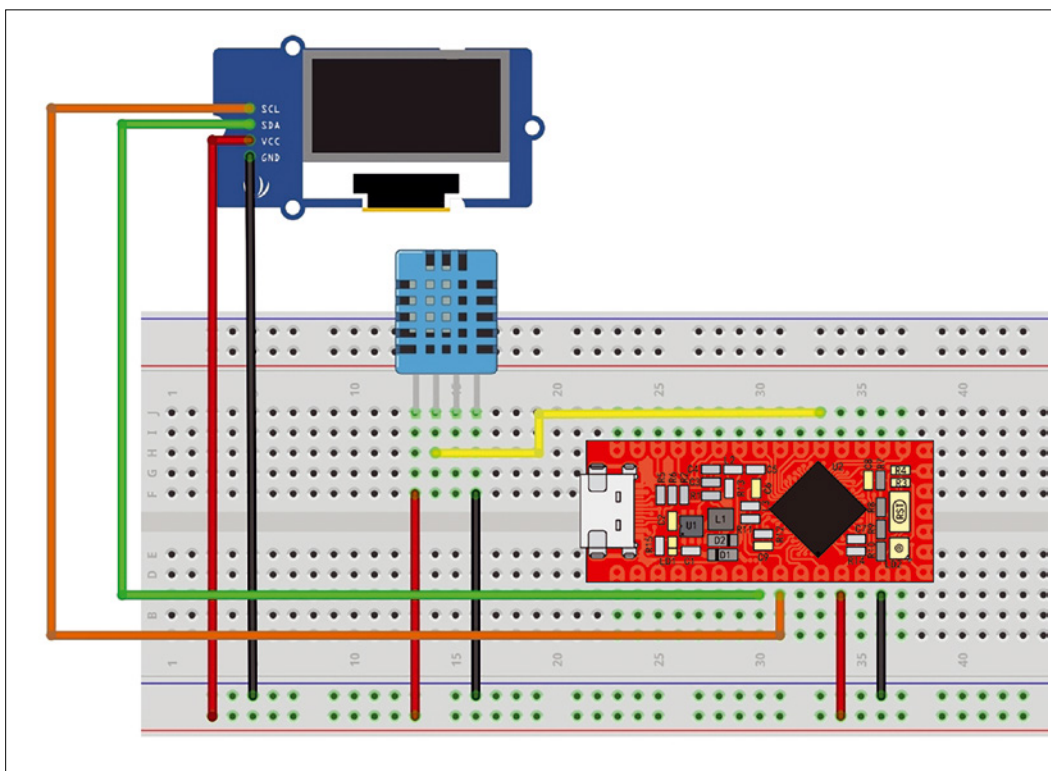


Fig. 8
Il menu degli
esempi della
libreria
Acrobotics
per SSD1306.

Fig. 9
Schema di cablaggio per l'esempio con DHT11.



usato come SDA e il pin A5 come SCL, come nel precedente Arduino UNO Rev 3. Lo sketch (**Listato 2**) include all'inizio le librerie Wire.h per il protocollo i2c, ACROBOTIC_SSD1306.h per il display OLED SSD1306 e DHTstable.h per il sensore DHT11. Creiamo un oggetto DHTstable DHT e definiamo il pin di dati del sensore con DHT11_PIN 2. Nel setup, Wire.begin() inizializza il bus i2c,

oled.init() inizializza il display SSD1306, con oled.setFont(font8x8) impostiamo il tipo di carattere (predefinito 8x8) e con oled.clearDisplay() puliamo lo schermo. Nel loop, leggiamo i dati di temperatura e umidità con:

```
int chk = DHT.read11(DHT11_PIN)
```

Quindi memorizziamo i dati nelle due variabili:

```
float hum = DHT.getHumidity(); //umidità
float tem = DHT.getTemperature(); //temperatura
```

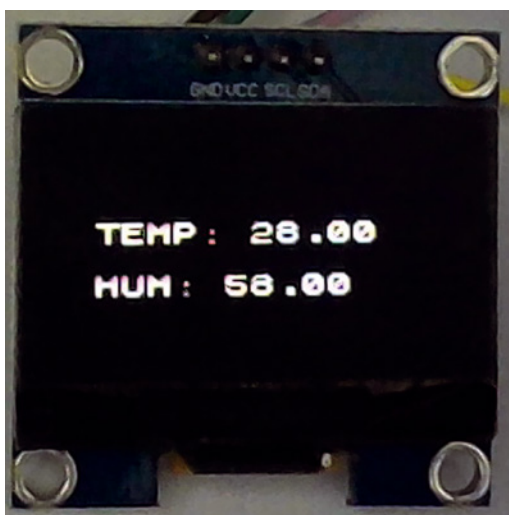
Creiamo due stringhe per i dati da stampare sul display:

```
String temperature = String(tem,2); // 2 cifre dec.
String humidity = String(hum,2); // 2 cifre decimali
```

E stampiamo il testo sul display:

```
oled.setTextXY(3,2);
// Posizione del cursore, inizio 2 della riga 3
oled.putString("TEMP: " + temperature);
// stampa sul display "Tem: " e la temperatura
oled.setTextXY(5,2);
// Posizione del cursore, inizio 2 della riga 5
oled.putString("HUM: " + humidity);
// stampa sul display "Hum: " e l'umidità
```

Fig. 10
Il display OLED con i dati temperatura e di umidità.



Con `delay(2000)` creiamo un ritardo per la lettura dei dati dal DHT11, che impone essere minimo due secondi.

Il risultato reale è visibile nella **Fig. 10**.

RTC SU DISPLAY OLED

Il chip Renesas RA4M1 integra un Real Time Clock (RTC), ovvero un orologio in tempo reale. Utilizzando il display OLED del circuito precedente, possiamo mostrarne l'uso per visualizzare ora, minuti, secondi, giorno, mese e anno, come illustrato nel **Listato 3**.

Per utilizzare l'RTC del chip, è necessario includere la libreria `RTC.h` dell'IDE. Includiamo inoltre le librerie `Wire.h` per la comunicazione I2C e `ACROBOTIC_SSD1306.h` per il controllo dell'OLED SSD1306, precedentemente installato. Creiamo una variabile `bool first_access = 0` per consentire l'accesso al display una sola volta, in modo da visualizzare continuamente l'orario e la data senza dover cancellare il display ogni secondo. Nel setup, inizializziamo l'RTC, la porta I2C e il display OLED.

```
RTC.begin
Wire.begin
oled.init
```

Impostiamo il font con `oled.setFont(font8x8)` e puliamo il display con `oled.clearDisplay`.

La data e l'orario di inizio viene impostato con la seguente funzione;

```
RTCTime startTime(30, Month::OCTOBER, 2024, 14, 24,
00, DayOfWeek::WEDNESDAY, SaveLight::SAVING_TIME_ACTIVE);
RTC.setTime(startTime); // imposta la data e l'orario
```

Notare che le costanti del mese e del giorno sono in inglese e sempre in maiuscolo. La costante `SAVING_TIME_ACTIVE` attiva il passaggio all'ora legale.

Nel loop, creiamo l'oggetto `currentTime` e prendiamo il tempo dall'RTC:

```
RTCTime currentTime // crea l'oggetto currentTime
RTC.getTime(currentTime) // ottiene l'ora corrente
```

Creiamo le stringhe per giorno mese e anno e per

↓ Listato 1

```
#include <FastLED.h> // Include la libreria FastLED
#define NUM_LEDS 1 // Numero di LED nella catena
#define DATA_PIN 17 // Pin dati per il controllo LED

CRGB leds[NUM_LEDS]; // Array per contenere i dati del colore dei LED

void setup() {
    FastLED.addLeds<NEOPIXEL, DATA_PIN>(leds, NUM_LEDS); // Inizializza i LED
    FastLED.setBrightness(20); // Imposta la luminosità
}

void loop() {
    leds[dot] = CRGB::Blue; // Imposta il LED corrente su blu
    FastLED.show(); // Aggiorna i LED
    leds[dot] = CRGB::Black; // Cancella il LED corrente
    delay(500); // Attende un breve periodo prima di passare al LED successivo
    leds[dot] = CRGB::Red; // Imposta il LED corrente su rosso
    FastLED.show(); // Aggiorna i LED
    leds[dot] = CRGB::Black; // Cancella il LED corrente
    delay(500); // Attende un breve periodo prima di passare al LED successivo
    leds[dot] = CRGB::Green; // Imposta il LED corrente su verde
    FastLED.show(); // Aggiorna i LED
    leds[dot] = CRGB::Black; // Cancella il LED corrente
    delay(500); // Attende un breve periodo prima di passare al LED successivo
    leds[dot] = CRGB::White; // Imposta il LED corrente su bianco
    FastLED.show(); // Aggiorna i LED
    leds[dot] = CRGB::Black; // Imposta il LED corrente su nero
    delay(500); // Attende un breve periodo prima di passare al LED successivo
}
```

▼ Listato 2

```
#include <Wire.h> // importa la libreria Wire per i2c
#include <ACROBOTIC_SSD1306.h> // importa la libreria ACROBOTIC per SSD1306
#include "DHTStable.h" // importa la libreria DHT stabile e compatibile

DHTStable DHT; // crea l'oggetto DHT

#define DHT11_PIN 2 // definisce il pin del DHT

void setup()
{
  Serial.begin(115200); // apre il monitor seriale
  Wire.begin(); // inizializza il bus i2c
  oled.init(); // inizializza il display SSD1306 OLED
  oled.setFont(font8x8); // imposta il tipo di carattere (predefinito 8x8)
  oled.clearDisplay(); // pulisce lo schermo
}

void loop()
{
  // variabile che legge i dati dal DHT
  int chk = DHT.read11(DHT11_PIN);
  switch (chk) // uno switch per controllare lo stato del DHT11
  {
    case DHTLIB_OK:
      Serial.print("OK,\t"); //significa che lo stato di lettura è OK
      break; // esce dal case
    case DHTLIB_ERROR_CHECKSUM:
      Serial.print("Checksum error,\t"); //significa che il checksum dei dati è errato
      break; // esce dal case
    case DHTLIB_ERROR_TIMEOUT:
      Serial.print("Time out error,\t"); //significa che il DHT ha superato il tempo di lettura dei dati
      break; // esce dal case
    default:
      Serial.print("Unknown error,\t"); // è accaduto qualcosa di inaspettato
      break; // esce dal case
  }

  float hum = DHT.getHumidity(); //variabile float per rilevare l'umidità
  float tem = DHT.getTemperature(); //variabile float per rilevare la temperatura

  //stampa i dati nel monitor seriale
  Serial.println(tem); // temperatura
  Serial.print(",\t"); //tabulazione
  Serial.print(hum); //umidità

  String temperature = String (tem,2); //solo 2 cifre decimali
  String humidity = String (hum,2); //solo 2 cifre decimali

  oled.setTextXY(3,2); // Imposta la posizione del cursore, inizio 2 della riga 3
  oled.putString("TEMP: " + temperature); // stampa sul display "Tem: " e la stringa di temperatura
  oled.setTextXY(5,2); // Imposta la posizione del cursore, inizio 2 della riga 5
  oled.putString("HUM: " + humidity); // stampa sul display "Hum: " e la stringa di umidità

  delay(2000); // ritardo per la lettura dei dati dal DHT11
}
```

ora minuti e secondi:

```
String giorno=String(currentTime.getDayOfMonth())+" ";
String mese = String(Month2int(currentTime.getMonth())) + " "; // sarà un numero intero
String anno = String(currentTime.getYear());
String ora = String(currentTime.getHour()) + ":";
String minuti=String(currentTime.getMinutes())+".";
String secondi = String(currentTime.getSeconds());
```

La condizione che cancella il display solo una volta è la seguente:

```
if (first_access==0) // se è 0... {
  oled.clearDisplay();
  // pulisce lo schermo al primo accesso e basta
  first_access=1;
  // diventa 1 e quindi il display non viene pulito
}
```



▼ Listato 3

```
#include "RTC.h" // importa la libreria RTC
#include <Wire.h> // importa la libreria Wire per i2c
#include <ACROBOTIC_SSD1306.h> // importa la libreria ACROBOTIC per SSD1306

bool first_access=0; // booleana di controllo per l'accesso

void setup() {
  RTC.begin(); // inizializza l'RTC
  Wire.begin(); // inizializza il bus i2c
  oled.init(); // inizializza il display SSD1306 OLED
  oled.setFont(font8x8); // imposta il tipo di carattere (predefinito 8x8)
  oled.clearDisplay(); // pulisce lo schermo
  // data e orario di inizio
  RTC.setTime(startTime, 0); // imposta la data e l'orario
}

void loop() {
  RTCTime currentTime; // crea l'oggetto currentTime
  // Ottiene l'ora corrente dall'RTC
  RTC.getTime(currentTime);
  // stringhe per giorno mese e anno
  String giorno = String(currentTime.getDayOfMonth()) + " ";
  String mese = String(currentTime.getMonth()) + " ";
  String anno = String(currentTime.getYear());
  // stringhe per ora minuti e secondi
  String ora = String(currentTime.getHour()) + ":";
  String minuti = String(currentTime.getMinutes()) + ":";
  String secondi = String(currentTime.getSeconds());
  if (first_access==0) // se è 0...
  {
    oled.clearDisplay(); // pulisce lo schermo al primo accesso e basta
    first_access=1; // diventa 1
  }

  oled.setTextXY(0,1); // imposta la posizione del cursore, inizio 1 della riga 0
  oled.putString("ORARIO RTC"); // stampa sul display il testo ORARIO RTC
  //vogliamo uno zero davanti alle cifre da 1 a 9
  if(ora.length() == 2 ) ora = "0" + ora;
  if(minuti.length()== 2 ) minuti = "0" + minuti;
  if(secondi.length(< 2 ) secondi = "0" + secondi;

  oled.setTextXY(2,1); // imposta la posizione del cursore, inizio 1 della riga 2
  oled.putString(giorno); // stampa sul display il giorno
  oled.setTextXY(2,int(giorno.length()+1); // imposta la posizione del cursore, inizio 1
  // + lunghezza del giorno

  // convertiamo in mesi italiani
  if (mese.toInt() == 1) mese = "GEN";
  if (mese.toInt() == 2) mese = "FEB";
  if (mese.toInt() == 3) mese = "MAR";
  if (mese.toInt() == 4) mese = "APR";
  if (mese.toInt() == 5) mese = "MAG";
  if (mese.toInt() == 7) mese = "LUG";
  if (mese.toInt() == 8) mese = "AGO";
  if (mese.toInt() == 9) mese = "SET";
  if (mese.toInt() == 10) mese = "OTT";
  if (mese.toInt() == 11) mese = "NOV";
  if (mese.toInt() == 12) mese = "DIC";

  oled.putString(mese); // stampa sul display il mese
  oled.setTextXY(2,int(mese.length()) + 5); // imposta la posizione del cursore, inizio 4
  // + lunghezza del mese

  oled.putString(anno); // stampa sul display l'anno
  oled.setTextXY(4,1); // imposta la posizione del cursore, inizio 1 della riga 4
  oled.putString(ora); // stampa sul display l'ora
  oled.setTextXY(4,int(ora.length()+1); // imposta la posizione del cursore, inizio 1 + lunghezza dell'ora
  oled.putString(minuti); // stampa sul display i minuti
  oled.setTextXY(4,int(minuti.length()+4); // imposta la posizione del cursore, inizio 4
  // + lunghezza dei minuti
  oled.putString(secondi); // stampa sul display i secondi

  delay(1000); //ritardo di un secondo
}
```

Fig. 11
Il display OLED con
l'orario RTC.



Per comodità di posizionamento delle cifre sul display, vogliamo uno zero davanti alle cifre da 1 a 9

```
if (ora.length() == 2 ) ora = "0" + ora;
if (minuti.length() == 2 ) minuti = "0" + minuti;
if (secondi.length() < 2 ) secondi = "0" + secondi;
```

A questo punto, stampiamo nel display:

```
oled.setTextXY(0,1); // imposta la posizione del
// cursore, inizio 1 della riga 0
oled.putString("ORARIO RTC");
// stampa sul display il relativo testo
```

Visto che vogliamo i mesi in italiano, convertiamo in mesi italiani il numero intero ottenuto in precedenza per il mese:

```
if (mese.toInt() == 1) mese = "GEN";
if (mese.toInt() == 2) mese = "FEB";
if (mese.toInt() == 3) mese = "MAR";
if (mese.toInt() == 4) mese = "APR";
if (mese.toInt() == 5) mese = "MAG";
if (mese.toInt() == 7) mese = "LUG";
```

```
if (mese.toInt() == 8) mese = "AGO";
if (mese.toInt() == 9) mese = "SET";
if (mese.toInt() == 10) mese = "OTT";
if (mese.toInt() == 11) mese = "NOV";
if (mese.toInt() == 12) mese = "DIC";
```

A questo punto, stampiamo tutto sul display:

```
oled.setTextXY(2,1); // inizio 1 della riga 2
oled.putString(giorno); // stampa il giorno
oled.setTextXY(2,int(giorno.length()+1);
// inizio 1 + lunghezza del giorno
oled.putString(mese); // stampa sul display il mese
oled.setTextXY(2,int(mese.length()) + 5);
// inizio 4 + lunghezza del mese
oled.putString(anno); // stampa sul display l'anno
oled.setTextXY(4,1); // inizio 1 della riga 4
oled.putString(ora); // stampa sul display l'ora
oled.setTextXY(4,int(ora.length()+1);
// inizio 1 + lunghezza dell'ora
oled.putString(minuti); // stampa i minuti
oled.setTextXY(4,int(minuti.length()+4);
// inizio 4 + lunghezza dei minuti
oled.putString(secondi); // stampa i secondi
```

Infine aggiorniamo la stampa sul display con un ritardo di un secondo:

`delay(1000).`

Il risultato è visibile in **Fig. 11**.

I PIN IN PIÙ

Nella nostra scheda NANO R4 Minima, abbiamo voluto esporre alcuni pin che sono disponibili nel chip Renesas, ma che non ci sono nell'omonima scheda Arduino. Oltre al pin digitale 17, che abbiamo usato per controllare il LED neopixel, sono disponibili i seguenti pin sul lato corto della scheda:

D14: pin digitale 14

D15: pin digitale 15

D16: pin digitale 16

A9: pin analogico 9

A0: pin analogico 0

Oltre ai pin analogici da A0 ad A5 della scheda Arduino R4 MINIMA, sono disponibili anche i pin A6, A7 e A8.

CONCLUSIONI

Ovviamente ci sarebbe ancora molto da dire sulle altre caratteristiche della scheda NANO R4, ma ci fermiamo qui.

Con la speranza di essere stati sufficientemente chiari, vi invitiamo a sperimentare con la nostra scheda. □



COSA OCCORRE ?

Codice	Descrizione	Prezzo € IVA inclusa
NANOR4	Nano R4 Minima con Renesas RA4M1	29,00
DHT11	Sensore di temperatura e umidità	3,50
OLED096C	Display OLED I2C 0,96"	11,00

IL MATERIALE VA RICHIESTO A:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - www.futurashop.it

Indispensabili in ogni laboratorio!

STAZIONE SALDANTE 60 W
CON BRACCI FLESSIBILI
E TEMPERATURA
REGOLABILE
DA 160 A 480°C

cod. SS203
€39,00



MINI STAZIONE SALDANTE 10W
CON TEMPERATURA
REGOLABILE
DA 100 A 450°C
IDEALE PER
COMPONENTI SMD

cod. SS202N
€19,90



STAZIONE SALDANTE 130 W
CON TEMPERATURA
REGOLABILE
DA 160 A 480°C
STAGNO,
PORTA STAGNO,
PAGLIETTA
IN METALLO
E SPUGNA

cod. TP237
€49,00



STAZIONE SALDANTE 48 W
CON DISPLAY LCD E
INDICAZIONE DELLA
TEMPERATURA IMPOSTATA E
DELLA TEMPERATURA REALE
DA 150 A 450°C

cod. VTSSC40N
€94,00



STAZIONE SALDANTE 48 W
CON TEMPERATURA
REGOLABILE DA 150 A 480°C

cod. TPGSS501
€21,00



STAZIONE SALDANTE/
DISSALDANTE PER COMPONENTI
SMD CON SALDATORE
DA 160 A 480 °C E
STILO AD ARIA CALDA
DA 160 A 480 °C

cod. TP282
€278,00



STAZIONE SALDANTE/
DISSALDANTE PER COMPONENTI
SMD CON SALDATORE
DA 50 A 480 °C
E STILO AD ARIA CALDA
DA 100 A 500 °C

cod. VTSS230
€165,00



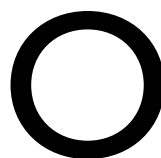
Scopri la gamma completa su **www.futurashop.it**

PRESEPE CON NANO R4



di ALESSANDRO SOTTOCORNOLA

Centralina di gestione delle luci e delle musiche per un presepe sempre al passo con l'evoluzione tecnologica.



Ogni anno le festività natalizie sono l'occasione per rispolverare addobbi e statuette e preparare quelli che sono i simboli, più o meno religiosi, della festa che ricorda la natività.

Indubbiamente, i cambiamenti culturali, con il passare del tempo, modificano le abitudini, allontanandoci da quelle con cui molti di noi sono cresciuti. Tuttavia, esistono tradizioni che resistono ancora all'omologazione dei costumi e all'allontanamento da fede e religioni. Una di queste, in particolare nel periodo natalizio, è quella di preparare il presepe, arricchendolo magari ogni anno con qualcosa di nuovo, come i dispositivi che la tecnologia, e in particolare l'elettronica, oggi offre a prezzi più accessibili rispetto al passato.

Ecco perché, periodicamente, ci dedichiamo alla realizzazione di una centralina per la gestione del presepe.

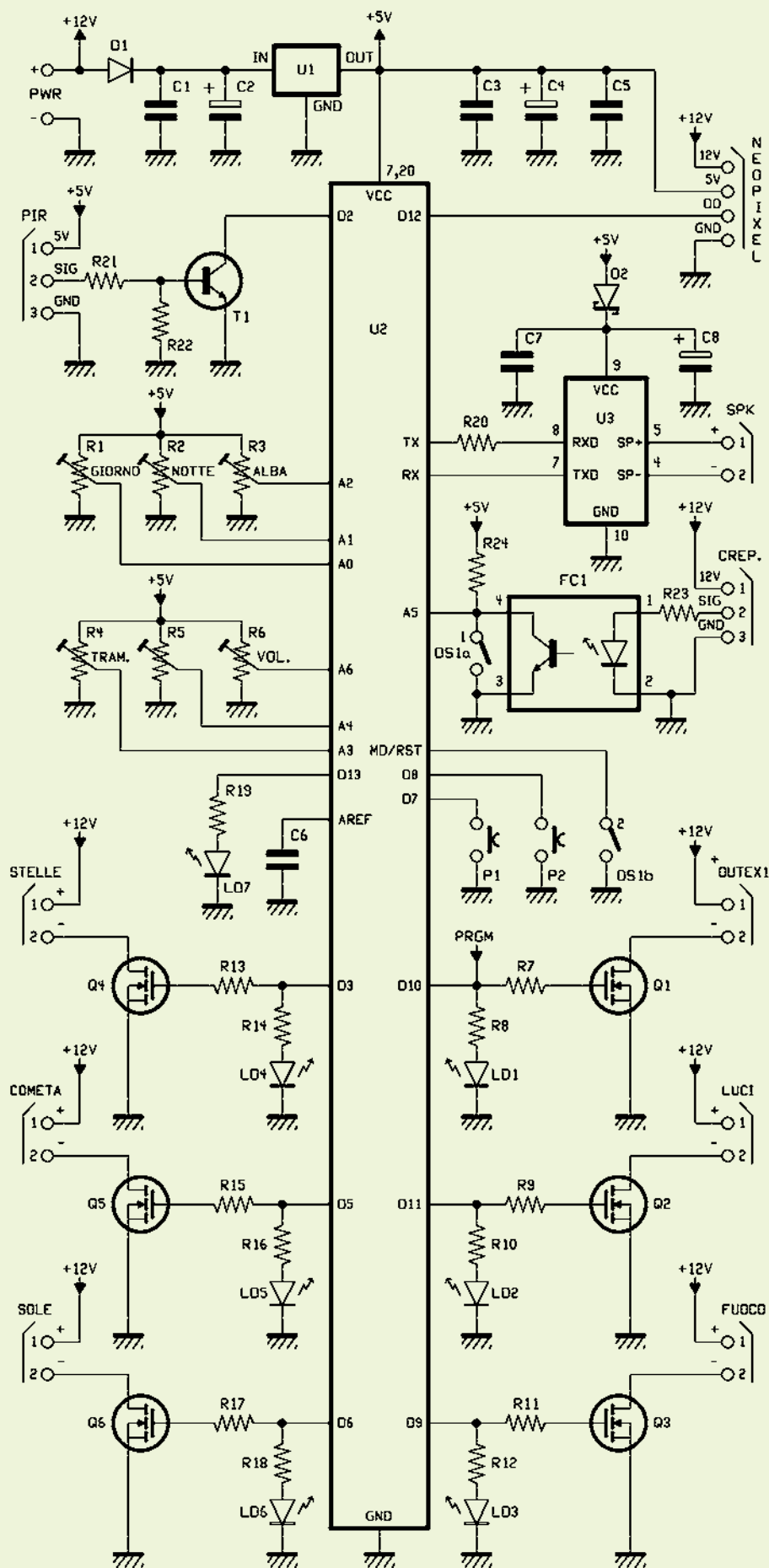
Quest'anno abbiamo deciso di riproporla in una versione aggiornata, gestita tramite hardware Arduino, come già avevamo fatto nel dicembre 2016 (fascicolo n. 201).

Tuttavia, questa nuova versione presenta alcune significative innovazioni: è stata realizzata principalmente con componenti SMD e utilizza un lettore MP3 più moderno (ed economico), oltre a una board Nano R4.

La particolarità di questa versione di Presepio risiede proprio nell'uso di questa scheda, progettata per funzionare con una speciale evoluzione della Arduino Nano R4 sviluppata da Futura Elettronica.

A questo proposito, rimandiamo al relativo articolo presente in questo fascicolo per ulteriori approfondimenti.





IL PROGETTO

Entriamo dunque nel vivo di questa scheda andando ad analizzarne le caratteristiche salienti a partire dalla CPU, che diversamente dalla precedente versione di Presepio (il cui nome si deve all'adozione di un'architettura Arduino basata su ATmega328) è contenuta in una scheda Futura Elettronica Nano R4. La sezione audio, ossia quella deputata a produrre le musiche del presepe, è contenuta in una breakout board riproduttore MP3, nella quale è possibile caricare file di pari formato attraverso una connessione USB, contando su uno spazio di memoria EEPROM di 4 MB; il modulo in questione è il JR6001 (cod. Futura Elettronica MP3PLAYERUSB) ed ha il pregio di non richiedere la SD-Card che invece era necessaria nella precedente versione di Presepio per caricare i file audio.

In questa versione abbiamo utilizzato 5 delle 6 uscite disponibili (con la possibilità di attivare anche la sesta uscita personalizzando il firmware, ad esempio per gestire i motori di un mulino o la pompa di una cascata). Il codice che comanda le uscite per le luci (sole, stelle, cometa, fuoco per i pastori e luci delle case) prevede un'accensione e uno spegnimento graduale, regolabili tramite i trimmer presenti sul circuito. Il box "Il funzionamento delle uscite" include un grafico esplicativo per chiarire il funzionamento descritto. Oltre alle uscite per le strip LED, abbiamo anche un'uscita Neopixel alla quale è possibile collegare catene di LED indirizzabili, per rendere il vostro presepe ancora più suggestivo e personalizzato. In aggiunta all'implementazione delle luci, a seguito di richieste sono stati inseriti due ingressi, dedicati rispettivamente ad un sensore crepuscolare e un sensore PIR, gestibili distintamente e secondo una gerarchia che verrà spiegata tra breve. Andiamo ora a vedere nel dettaglio la scheda di gestione del presepe, così da vedere in cosa consiste l'hardware.

SCHEMA ELETTRICO

Come accennato, il cuore del circuito è la scheda Nano R4, realizzata da Futura Elettronica. I suoi I/O D0 e D1 (corrispondenti alla seriale hardware dell'Arduino UNO R3) non sono condivisi con la seriale USB, permettendone così l'uso per altre funzionalità aggiuntive. In particolare, in questa configurazione, li abbiamo utilizzati per gestire il modulo MP3. Qui di seguito riportiamo i pin della board Nano R4 assegnati alle relative funzioni.

CARATTERISTICHE TECNICHE

 Tensione di alimentazione: 12÷14 Vcc	 Supporto strip NeoPixel
 Corrente assorbita: 15 A	 Riproduzione audio
 Uscite controllo luci: 6	 Controllo crepuscolare e con sensore PIR

```
P1 = 7;           //Pulsante Play
P2 = 8;           //Pulsante Stop
PIR = 2;          //Sensore PIR
TX = 1;           //Pin TX verso MP3
RX = 0;           //Pin RX da MP3

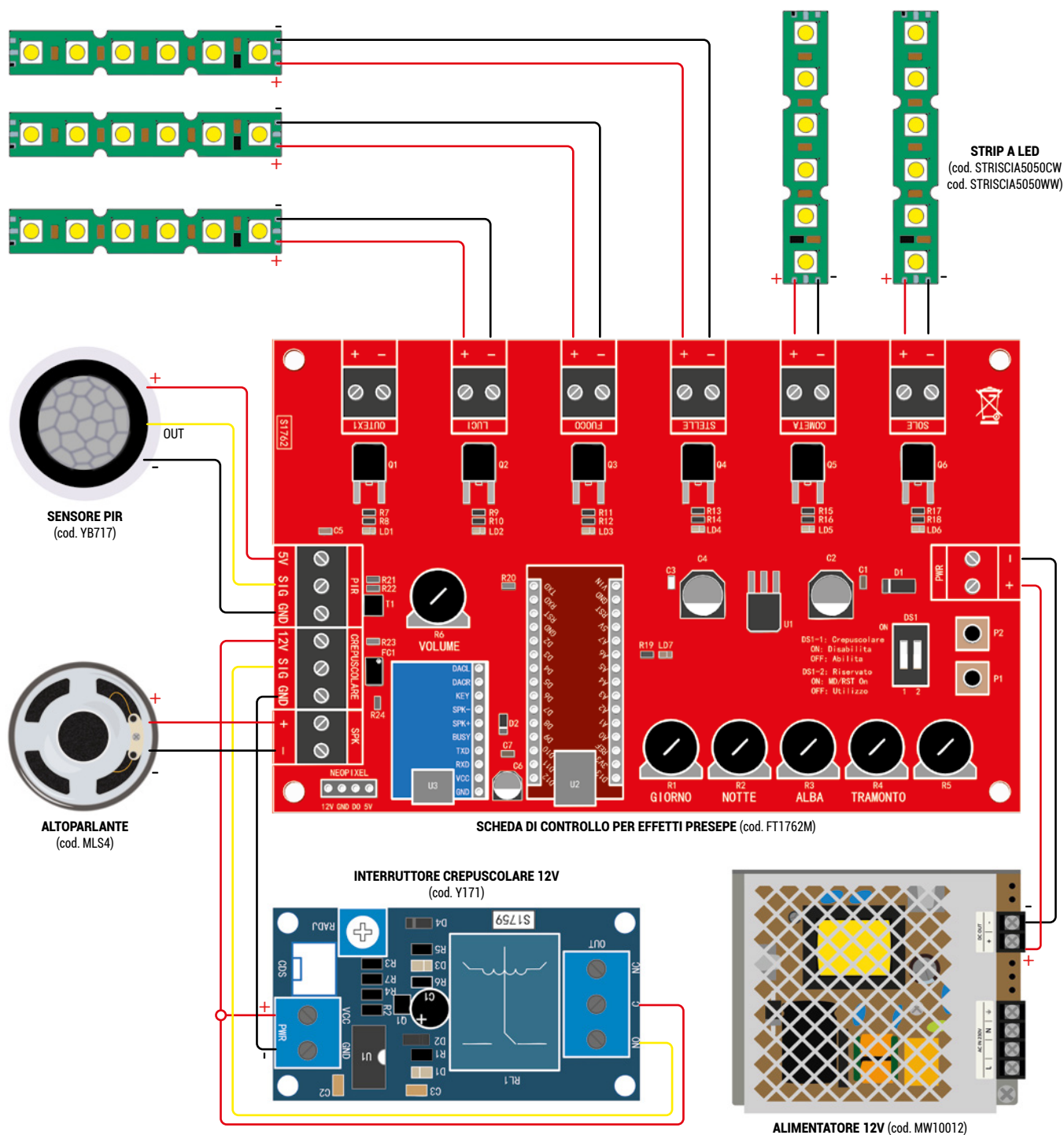
const int OUT1 = 6; //Sole
const int OUT2 = 5; //Cometa
const int OUT3 = 3; //Stelle
const int OUT4 = 9; //Fuoco
const int OUT5 = 11; //Luci
const int OUT6 = 10; //OUTEX1
CHANGESTATE = 13; //Led
NEOPIXEL = 12; //Neopixel
TRIM1 = A0;      //Giorno
TRIM2 = A1;      //Notte
TRIM3 = A2;      //Alba
TRIM4 = A3;      //Tramonto
TRIM5 = A4;      //Extra
CREPUSCOLARE = A5; //Ingresso Crepuscolare
TRIM_VOL = A6;   //Volume Audio
```

Guardando lo schema elettrico vediamo che la Nano R4 pilota i gate di sei MOSFET enhancement-mode a canale N (si tratta di DMN6068LK3-13, 60V di V_{ds} e 6A di corrente di drain) che materialmente commutano l'alimentazione sui LED o strip a LED corrispondenti, attraverso i pin digitali impostati come uscite, secondo questa corrispondenza:

- D3 pilota Q4 (tramite R14) il cui drain alimenta le luci che simuleranno le stelle;
- D5 pilota Q5 (tramite R15) il cui drain comanda le luci della cometa;
- D6 comanda Q6 (tramite R17) il cui drain alimenta le lampade che simulano la luce del sole e quindi le fasi di alba e tramonto;



schema di CABLAGGIO



- D9 pilota Q3 (tramite la R11) il cui drain alimenta le lampade che simuleranno, pulsando opportunamente, il crepitio della fiamma di un fuoco;
- D11 pilota Q2 (tramite la R9) il cui drain controlla l'uscita LUCI;
- D10 comanda Q1 (tramite R7) il cui drain gestisce l'uscita OUTEX1 (cui potete collegare altre strisce a LED, motori o altri carichi).

Tra il gate di ciascuno di questi MOSFET e massa è collegato un LED, con relativa resistenza di carico, che ripete il comportamento delle luci eventualmente collegate alla rispettiva uscita (ossia tra il drain e il positivo di alimentazione) tramite la morsettiera corrispondente. Questo permetterà di testare la scheda prima di collegare le lampade, ovvero di verificare l'attività delle uscite nel caso

qualcosa dovesse non funzionare nel circuito presepe, una volta cablato.

Ciascuna uscita può alimentare lampade comuni oppure a LED, purché funzionanti a 12 volt e che assorbano una corrente individuale di 5 ampere, tenendo però presente che la somma della corrente assorbita dalle uscite non deve superare i 15 ampere. Questa è anche la massima corrente che la scheda può gestire e può essere aumentata stagnando le piste che portano dalla morsettiera di alimentazione ai MOSFET e da questi alle relative morsettiere di uscita. Questi valori, considerato che il circuito nasce per pilotare strip a LED o comunque composizioni di LED, sono più che accettabili e per un presepe da casa sono sovrabbondanti, pur bastando anche per una struttura all'aperto, grazie all'efficienza luminosa dei LED. Ciascuna delle uscite a MOSFET fornisce impulsi rettangolari TTL-compatibili di frequenza costante e modulati in larghezza, secondo la tecnica PWM: ciò consente di variare il valore medio della tensione applicata dai MOSFET ai carichi loro collegati e di conseguenza la potenza e la luminosità, con un'elevatissima efficienza, in quanto la resistenza in stato di ON (piena conduzione) dei MOSFET è di (appena 68 milliohm) e quindi la potenza dissipata, a parità di potenza trasferita, è limitatissima se confrontata a quella che comporterebbe un pilotaggio serie a tensione continua, dove il transistor dovrebbe sobbarcarsi la differenza tra la tensione di alimentazione e quella applicata al carico. I segnali PWM vengono generati dai moduli PWM interni al microcontrollore della scheda Arduino, opportunamente impostati dallo sketch per ciascuna uscita.

L'attuale configurazione circuitale prevede di alimentare le strip di LED collegandone positivo e negativo rispettivamente a + e - della relativa uscita, tuttavia siccome i MOSFET possono "reggere" tensioni drain-source in interdizione pari a 60 volt, è anche pensabile partire da un'alimentazione esterna anche a 24+36 volt c.c. in modo da ridurre l'assorbimento a parità di potenza elettrica gestita e di rendere più versatile l'installazione sfruttando linee di alimentazione o alimentatori esistenti. Se l'alimentazione delle lampade non viene prelevata dal circuito e si utilizzano strip LED come corpi illuminanti, il positivo delle strip va collegato al polo positivo dell'alimentatore scelto, mentre il contatto negativo va collegato all'uscita della nostra centralina. È importante, inoltre, che la massa dell'alimentatore e quella della scheda siano collegate tra loro.

In tutti i casi le strip a LED devono essere composte da diodi puri e semplici e non devono avere a bordo un regolatore di corrente.

Le uscite della centralina sono sei, ma lo sketch che forniamo gestisce le prime cinque (ossia LUCI, FUOCO, SOLE, COMETA, STELLE), per simulare la luce del giorno, di quella delle stelle, dei fuochi della capanna/grotta e delle case dei pastori, ed infine di illuminare la stella cometa. Le luci si accendono e si spengono gradatamente seguendo un ciclo che simula un'intera giornata. A queste uscite di base potete aggiungere, intervenendo sullo sketch, quella siglata OUTEX1, configurandola affinché funzioni come meglio credete, ad esempio per il controllo di accessori del presepe come una ruota a pale, il movimento di personaggi animati e via di seguito. Per quanto riguarda le uscite di base, abbiamo suddiviso la sequenza completa in quattro fasi denominate giorno, tramonto, notte e alba, la durata delle quali può essere impostata mediante i trimmer, e l'escursione corrispondente si può reimpostare intervenendo sullo sketch (ma in maniera uguale per tutte); ciò vale dunque per il giorno e la notte, ma anche per le due fasi di transizione (tramonto e alba). Queste ultime due sono le fasi più suggestive: durante il tramonto, la luminosità del giorno (realizzata con i LED collegati all'uscita SOLE) diminuisce a poco a poco, mentre nel cielo iniziano a illuminarsi le stelle (si attiva l'uscita omonima); a un certo punto, prima che il ciclo si concluda, si accendono i fuochi delle case e della capanna (uscita FUOCO). Quando tutte le stelle in cielo sono completamente illuminate, appare anche la stella cometa (accesa dall'uscita COMETA); durante l'alba si spengono gradatamente tutte le luci luminosi, mentre aumenta lentamente sino a raggiungere la massima luminosità la luce del giorno. Oltre alle uscite a MOSFET già descritte, lo sketch che gira sulla board Nano R4 implementa anche il controllo delle altre uscite presenti sulla scheda e del modulo audio per il presepe. In particolare, all'avvio, lo sketch configura il pin D12 come uscita per l'invio dei comandi seriali a una eventuale strip Neopixel collegata al pin-strip NEOPIXEL, che fornisce alimentazione a 5V e 12V. Inoltre, attiva la seriale corrispondente alle linee D0 e D1 per comunicare con il modulo MP3 (U3). Notate che attualmente nello sketch non è stata prevista una gestione dell'uscita NeoPixel, ma abbiamo solo l'hardware; chi volesse potrà modificare lo sketch includendo innanzitutto la libreria FastLED scaricabile all'indirizzo futura.net.it/download/libreria-fastled-e-esempi-per-la-board-nano-r4.



Passiamo ora alle uscite a MOSFET per specificare che la sequenza di gestione delle luci, implementata dallo sketch, è strutturata in quattro fasi: giorno, tramonto, notte e alba. La durata di ciascuna fase può essere regolata indipendentemente tramite appositi trimmer, la cui tensione

del cursore è letta da un I/O di Arduino configurato come input analogico e acquisita tramite il convertitore A/D interno.

Più precisamente, l'ADC è assegnato alle linee A0, A1, A2, A3, A4 e A6 del Nano R4, utilizzate per scansionare le tensioni dei sei trimmer on-board: R1, R2, R3, R4, R5 e R6. I trimmer R1, R2, R3 e R4 permettono di variare la durata delle fasi di giorno, notte, alba e tramonto, rispettivamente.

IL FUNZIONAMENTO DELLE USCITE

Di giorno l'unica uscita attiva è quella che simula il sole e che illumina, appunto, "a giorno" il presepe. L'uscita sole resta attiva per un tempo compreso tra 0 e 102 secondi a seconda di come viene settato il trimmer R1. Scaduto il tempo ha inizio una nuova fase: il tramonto, la cui durata dipende dal trimmer R4, con il quale è possibile impostare un intervallo fra 0 e 102 s. A poco a poco la luce del giorno si attenua mentre aumenta la luminosità delle lampade che simulano le stelle, sino al completo spegnimento del sole che coincide con la massima illuminazione delle stelle.

Esattamente a metà di questa fase di transizione iniziano ad accendersi i fuochi delle case che raggiungono la massima luminosità al termine del ciclo. Questa uscita genera una luce tremolante che simula il bagliore del fuoco. Infine, al termine del ciclo, si illumina gradatamente anche la luce che simula la stella cometa. Il passaggio dalla minima alla massima luminosità avviene in tempo pari ad 1/4 di quello impostato per il TRAMONTO.

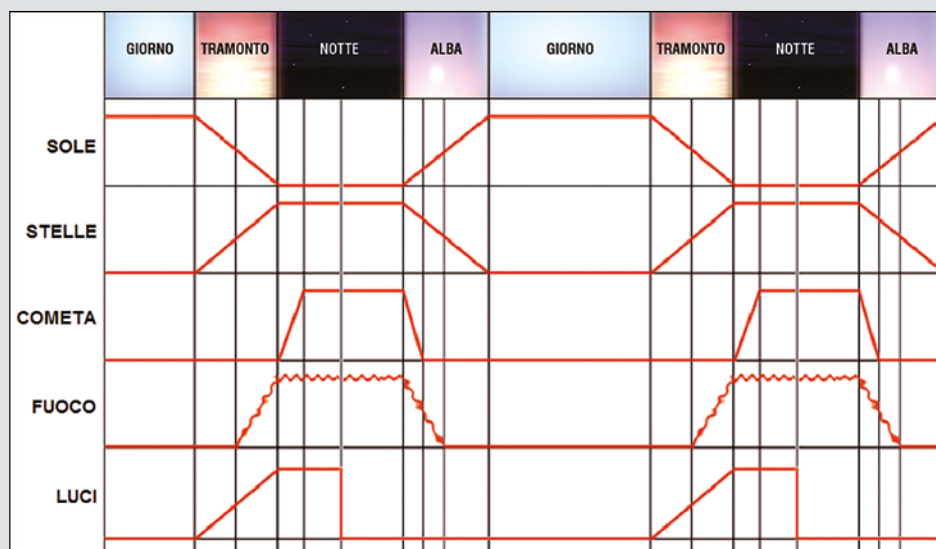
A questo punto ci troviamo in piena notte con le stelle in cielo che brillano, la cometa completamente illuminata ed i fuochi nelle case accesi col loro tipico tremolio. La durata di questa fase (compre-

sa, come per il giorno, tra 0 e 102 secondi circa), che è la NOTTE, viene impostata mediante il trimmer R2. Allo scadere del tempo impostato, ha inizio la quarta ed ultima fase: l'alba.

Gradualmente l'intensità luminosa delle stelle cala mentre aumenta quella del giorno sino al completo spegnimento delle stelle ed alla completa accensione delle lampade che simulano la luce del giorno. All'inizio di questa fase anche la cometa si spegne gradatamente ma molto più velocemente tanto che, trascorso un periodo pari ad 1/4 del tempo impostato per l'alba, la cometa risulta completamente spenta. Sempre all'inizio di questa fase di transizione, anche la luminosità dei fuochi inizia a calare sino allo spegnimento.

In questo caso il passaggio dalla massima luminosità allo spegnimento completo avviene in un tempo pari ad 1/2 di quello impostato per l'ALBA, mediante il trimmer R3. A questo punto è stato simulato un ciclo di 24 ore ed il sistema si appresta a ripetere all'infinito la sequenza programmata.

L'uscita LUCI si attiva gradualmente al tramonto per illuminare le case e la grotta, spegnendosi poi a metà della notte.



Andamento della tensione applicata alle luci delle uscite in base alle fasi di funzionamento.

Il trimmer R5 è associato all'uscita OUTEX1, ma la sua funzione è personalizzabile poiché tale uscita non ha una funzione assegnata dallo sketch. Infine, il trimmer R6 regola il volume di riproduzione del modulo MP3, intervenendo sulla routine dedicata che invia i relativi comandi tramite la seriale TX/RX. Per quanto riguarda le fasi di giorno, notte, alba e tramonto, la durata dei rispettivi cicli è direttamente proporzionale alla tensione applicata dal relativo trimmer, quindi portando il cursore di R1 verso massa, per esempio, si accorcia la durata del giorno, la quale invece aumenta portandolo verso il +5 V. Il duty-cycle dei segnali PWM alle uscite SOLE, STELLE, COMETA e FUOCO viene gestito secondo la periodicità prevista dallo sketch Arduino. Questo perché le fasi descritte coinvolgono la gestione di più di un'uscita e, nel caso della transizione giorno/notte (TRAMONTO), per esempio, riguardano l'uscita SOLE e quella STELLE, che vanno in dissolvenza, come avviene anche nel passaggio notte/giorno (ALBA), seppure nella successione opposta. Il passaggio da una fase alla successiva viene evidenziato dalla breve accensione del LED LD7. La durata di ogni singola fase viene impostata tramite l'apposito trimmer e la durata massima di ogni fase è di circa 100 secondi; per l'esattezza, la durata è ottenuta dalla lettura dell'ADC su 10 bit (tra 0 e 1.023) considerata in millisecondi, moltiplicata per 100 dal firmware. Possiamo adesso all'audio del presepe, che viene ottenuto grazie a un valido modulo siglato U3 nello schema elettrico; si tratta di una piccola scheda dotata di decoder MP3 gestito da microcontrollore, interfaccia USB 2.0 e amplificatore audio in classe D. Grazie alla sua memoria integrata da 4MB, contiene i file audio in formato MP3 o WAV caricabili da Personal Computer tramite la connessione micro USB (in modalità Device). Nel nostro circuito, U3 viene alimentato dalla linea a 5V attraverso un diodo Schottky (D2) che evita, quando il modulo è collegato al PC, di riversare i 5V nel circuito. Il suo assorbimento in assenza di segnale audio rimane nei 10 mA. Ai piedini 4 (SP-) e 5 (SP+) si collega, tramite il connettore di fissaggio alla scheda base e relativa morsettiera, un piccolo altoparlante da 8 ohm 3 watt, che riprodurrà i brani musicali, con una qualità che può spaziare da 8 a 320kbps. Se vi occorre un audio di maggior potenza, utilizzate l'uscita audio (+ e - SPK) per pilotare un amplificatore di potenza cui applicare una cassa acustica; in alternativa potete prendere l'audio dai contatti 9 e 10, che sono le uscite audio a basso livello dei

canali, rispettivamente, RIGHT (DACL, ossia canale destro) e LEFT (DACL, audio canale sinistro).

In questo caso potete contare su un audio stereo. La gestione del modulo MP3 nel nostro circuito si opera tramite la seriale software supportata dal microcontrollore della Nano R4; i comandi permettono di scegliere il file da riprodurre, regolare il volume d'ascolto ecc.

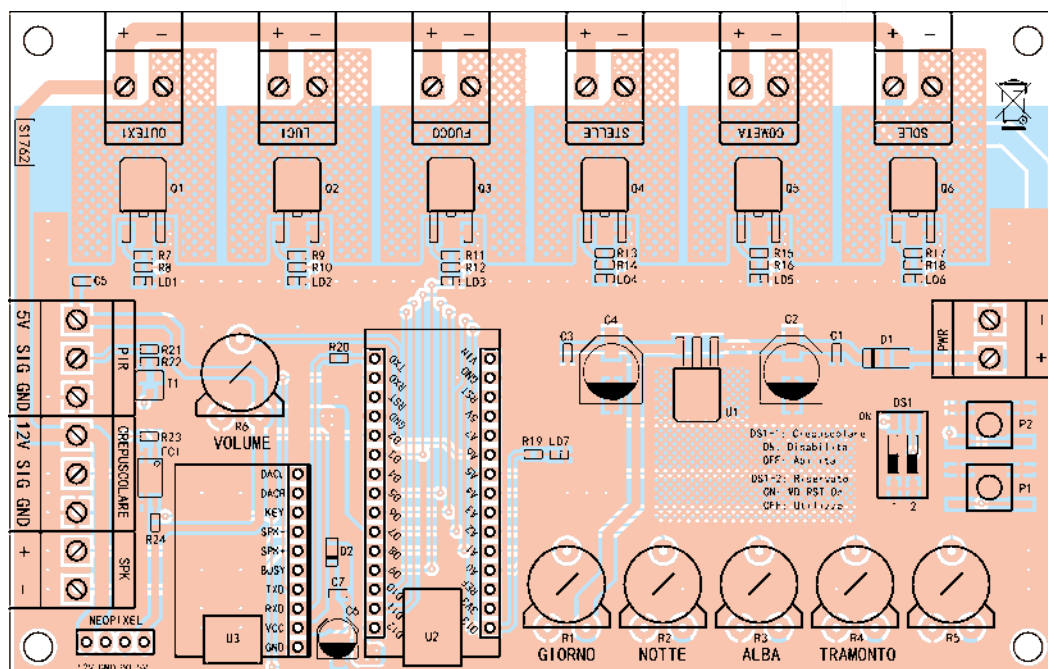
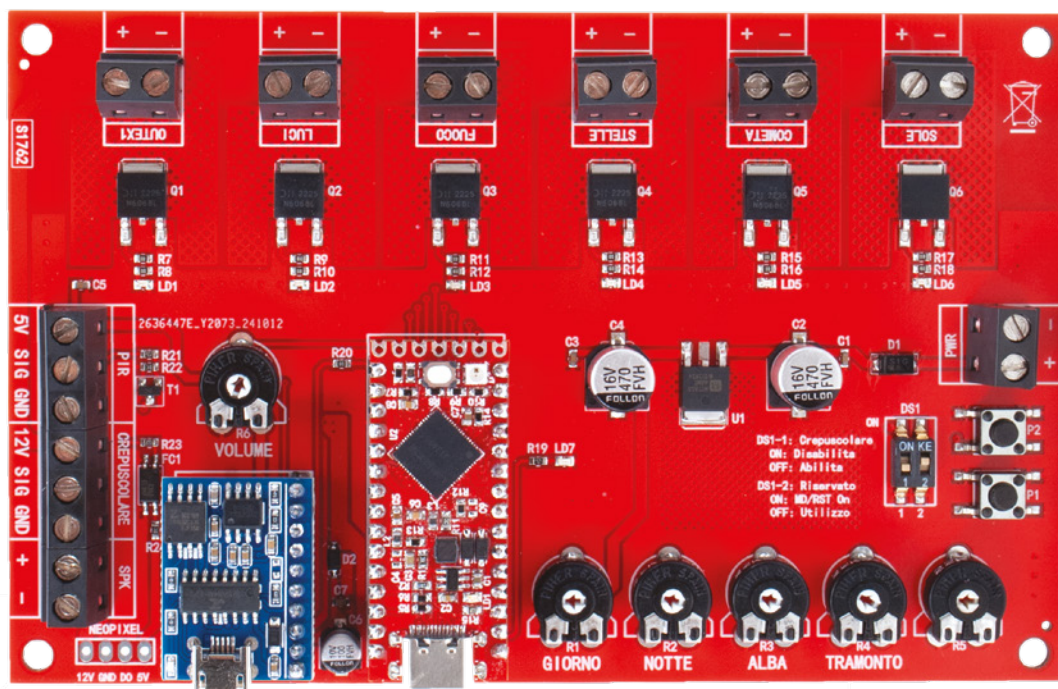
Quando l'audio è in riproduzione, il LED LD7 viene acceso dalla linea D13 dell'U2.

I file dell'audio che volete sia riprodotto devono essere creati sul PC e scritti nella memoria del modulo con nome nel formato 00001.mp3, 00002.mp3, 00003.mp3 e 0004.mp3, perché il primo file viene richiamato durante la fase GIORNO, il secondo durante il TRAMONTO, il terzo quando viene acceso il FUOCO (fuoco tremolante) e il file 0004.mp3 viene richiamato quando arriva l'ALBA. Durante il passaggio tra una fase e l'altra, il volume di riproduzione viene abbassato gradualmente fino al termine del brano e viene avviata, con volume in graduale aumento, la riproduzione del brano successivo (tipo ciò che avviene alla radio o negli album musicali); questo vale nel passaggio dal brano da una fase alla seguente.

Passiamo adesso alla sezione di controllo esterno del presepe, che consente di asservirne l'attività alle condizioni di illuminamento nell'ambiente (crepuscolare) ed alla presenza di persone di fronte allo stesso (PIR); nel primo caso si utilizza il connettore siglato CREP al cui pin 2 (SIG) si deve applicare una tensione, anche semplicemente ottenuta da un interruttore crepuscolare con uscita a transistor o a contatto pulito (collegato tra 12V del connettore stesso e SIG...) in modo che il fotoaccoppiatore FC1 vada in conduzione e porti a zero logico l'I/O A5 della Nano R4, inizializzato come input deputato alla lettura della condizione di illuminamento nell'ambiente. Per il test del sistema, abbiamo previsto il dip-switch DS1b, chiudendo il quale si forza l'avvio dei giochi di luce anche quando nulla è connesso al connettore CREP. Quindi il suddetto switch è utile anche per disabilitare la funzione crepuscolare.

Nel caso del nostro prototipo, abbiamo eseguito i test abbinando un sensore crepuscolare cod. Futura Elettronica DLS12V10A/Y171 che dispone di un'uscita a relé, della quale connettere il contatto C al SIG del connettore CREP della nostra scheda e l'N.O. al 12V della stessa. La funzione crepuscolare viene attivata portando ad OFF il dip DS1a; in caso contrario va portato ad ON in modo da attivare la scheda sempre.





Andiamo ora al controllo esterno con sensore PIR, che nel caso del nostro prototipo è il cod. Futura Elettronica YB717 (oppure il cod. MINIPIRMOD); la funzione di attivazione tramite PIR può essere gestita dal firmware solo se il sensore crepuscolare è attivo e ha dato il consenso. Il sensore deve essere collegato ai relativi morsetti.

Quando il sensore rileva il movimento di una persona di fronte ad esso, attiva l'esecuzione del ciclo completo dell'animazione del presepe; una volta terminato il ciclo, il tutto ritornerà in standby fino al prossimo rilevamento. Nel caso non si voglia usare il sensore PIR, sarà sufficiente fare un collegamento esterno, ovvero

Elenco Componenti:

R1, R2, R3, R4, R5, R6: Trimmer 10kohm M.O.
R7, R8, R9, R10, R11: 1 kohm (0603)
R12, R13, R14, R15, R16: 1 kohm (0603)
R17, R18, R20: 1 kohm (0603)
R19: 470 ohm (0603)
R21: 4,7 kohm (0603)
R22 : 10 kohm (0603)
R24: 10 kohm (0603)
R23: 2,2 kohm (0603)
C1, C3, C5, C7: 100 nF ceramico (0603)
C2: 470 µF 16 VL elettrolitico (ø 8mm)
C4: 470 µF 16 VL elettrolitico (ø 8mm)
C6: 100 µF 6.3 VL elettrolitico (ø 5mm)
D1: RGF1M
D2: MBR0540T1G
FC1: EL3H7(C)(TA)-G
LD1: LED Rosso (0603)
LD2: LED Rosso (0603)
LD3: LED Rosso (0603)
LD4: LED Rosso (0603)
LD5: LED Rosso (0603)
LD6: LED Rosso (0603)
LD7: LED verde (0603)
P1: Microswitch
P2: Microswitch
T1: BC817
DS1: Dip-Switch 2 vie
U1: HT7805ARMZ
U2: Nano R4 (Programmato con codice MF1762)
U3: Modulo MP3 4MB (MP3PLAYERUSB)
Q1, Q2, Q3 : DMN6068LK3-13
Q4, Q5, Q6: DMN6068LK3-13
CN1a, CN1b: Pin Strip Femmina 15 vie 2.54mm
CN2: Pin Strip Femmina 8 vie 2.54mm
CN3: Pin Strip Maschio 4 vie 2.54mm
CN4, CN5: Morsettiera 3 poli 5.08mm
CN6, CN7, CN8, CN9: Morsettiera 2 poli 5.08mm
CN10, CN11, CN12, CN13: Morsettiera 2 poli 5.08mm

Varie:

Circuito Stampato S1762 (139x88 mm)

un ponticello tra il pin 5V e SIG del morsetto "PIR". Riguardo al PIR e il Crepuscolare è stato detto tutto, ma a bordo del circuito sono presenti anche due pulsanti P1 e P2 che hanno rispettivamente funzione di riproduzione del ciclo e di stop. Il pulsante P1 può ad esempio essere utilizzato quando il ciclo non è già in riproduzione e il PIR

non sta rilevando movimento, per mandare in esecuzione manuale il ciclo. Il pulsante P2, se premuto durante la riproduzione, permette di fermare il ciclo di riproduzione. Bene, a questo punto non ci resta che descrivere la sezione di alimentazione: il circuito richiede da 12 a 14 Vcc (la corrente richiesta dipende da quanti utilizzatori collegate alle uscite, sempre restando nel limite già detto) tramite la morsettiera +/- PWR, la cui tensione giunge direttamente alle uscite a MOSFET e al contatto 12V del connettore NEOPIXEL. La tensione di ingresso giunge anche all'anodo del diodo D1, la cui funzione è proteggere il resto del circuito dall'inversione di polarità, quindi dal suo catodo viene applicata all'ingresso del regolatore di tensione U1, un 7805 in versione SMD (package in formato TO-252) utilizzato per ricavare i 5 volt stabilizzati con cui funzionano la scheda Arduino e il modulo audio U3.

I condensatori C1 e C2 servono a filtrare gli impulsi creati sulle piste per effetto dell'assorbimento dei LED e in generale dei carichi collegati alle uscite a MOSFET; ciò è necessario in quanto la pulsazione dell'alimentazione dei diodi è a frequenza elevata e altrimenti i disturbi (che poi sono abbassamenti di tensione pur lievi concomitanti con l'accensione dei singoli LED) potrebbero interferire con il corretto funzionamento del microcontrollore. Analoga è la funzione dei condensatori C3, C4 e C5 collocati all'uscita del 7805 e di quelli sparsi lungo la linea positiva a 5V, come anche C7 e C8, che filtrano l'alimentazione del modulo MP3.

REALIZZAZIONE PRATICA

Passiamo ora alle note costruttive, premettendo che tutti i componenti saranno montati su un circuito stampato a doppia faccia e che, a eccezione dei trimmer e delle morsettiere, tutti gli elementi sono a montaggio superficiale. Per questo motivo, è necessario avere almeno un minimo di esperienza con la tecnologia SMT e la relativa dotazione di laboratorio, che, nella sua forma più essenziale, comprende: un saldatore a punta fine di potenza massima 30W, filo di lega saldante con diametro massimo di 0,5 mm, una lente di ingrandimento e una pinzetta per il posizionamento dei componenti, oltre a del flussante a bassa densità.

Il circuito stampato può essere realizzato tramite il servizio PCBPRODUCTION di Futura Elettronica, che consente di ottenere a basso costo un PCB a doppia faccia con fori metallizzati. I file Gerber del PCB sono disponibili gratuitamente sulle pagine di presentazione di questo numero, sul sito



elettronica.in.it. Una volta ricevuto il PCB da Futura Elettronica, iniziate a posizionare i componenti più piccoli, come resistenze e condensatori non polarizzati, seguiti dai LED SMD, dai diodi e dai MOSFET di potenza (Q1-Q6). I MOSFET possono essere saldati lateralmente con la punta del saldatore e filo di stagno, dopo averli centrati accuratamente sulle rispettive piazzole, sulle quali avrete già applicato un velo di stagno e del flussante. Lo stesso metodo vale per il regolatore HT7805ARMZ, che ha un package identico a quello dei MOSFET. Proseguite quindi con i pulsanti in miniatura, il dip-switch e i condensatori elettrolitici, orientandoli come mostrato nel piano di montaggio presente in queste pagine, che servirà da guida nelle varie fasi di saldatura per i componenti polarizzati o con orientamento specifico (come diodi e LED).

Sistemati i componenti SMD si passa a quelli a foro passante, che sono i sei trimmer, i pin-strip per realizzare i jumper, gli strip femmina per ospitare la Nano R4 ed il modulo MP3, nonché le morsettiere a passo 5 mm: di queste ne occorrono 8 bipolari per le uscite, l'alimentazione e l'altoparlante (SPK) e due tripolari per gli ingressi PIR e crepuscolare. Ora bisogna introdurre la scheda Nano R4 e il modulo MP3 nei rispettivi strip, orientandoli come indicato (con le prese USB rivolte all'esterno del PCB...) e a quel punto potete già utilizzare il controller, premettendo che per un primo test con le funzioni essenziali, potete portare in ON SW1b e lasciare invece aperto SW1a, la cui funzione non è gestita nel normale impiego della centralina. Per utilizzare la scheda occorre prima caricare lo sketch (disponibile sul nostro sito web insieme ai

file di questo numero della rivista) ovvero aprire l'IDE di Arduino su un PC in cui è installato, quindi aprire il file dello sketch, caricarlo nella scheda che avrete collegato a una porta USB del computer mediante un cavo USB-A/USB-C. Ricordate che nell'IDE, dal menù Strumenti>Scheda, dovete selezionare Arduino Nano R4, ma ciò sarà possibile solo dopo aver scaricato il pacchetto "futura" dal nostro sito <https://futura.net.it/download/pacchetto-futura-per-la-board-nano-r4> ed aver scompattato il file .zip corrispondente, quindi copiato la cartella "futura" in "C:\Utenti\AppData\Local\Arduino15\packages". La scheda sarà visibile nel menu Strumenti, nel sottomenu Scheda, alla voce Futura Group Boards. Qui selezionate la voce Futura NANO R4 Minima e la porta seriale che viene installata dalla connessione USB.

IL FUNZIONAMENTO

A programmazione terminata la NANO R4 è pronta all'uso. Riguardo al modulo MP3, per copiare i file, è necessario collegare tramite cavo micro-USB, il modulo al computer e attendere che venga installato automaticamente il driver nel sistema operativo così da vedere lo stesso come drive disco. Aprendo il "Disco" si vedrà una serie di file che dovranno essere cancellati in modo da liberare la memoria; a quel punto potrete copiare i quattro file che abbiamo preparato, ma che potranno essere sostituiti con altri file audio a piacere, purché venga mantenuto il medesimo nome. Ricordate che i file da caricare devono avere bassa qualità, in modo da farli stare tutti nei 4MB di memoria dell'MP3. Alla prima accensione consigliamo di portare i cursori di tutti i trimmer a circa 1/4 della corsa in senso orario, in modo da verificare la sequenza, per poi impostare ogni singolo tempo a piacere. Ora non resta che fornire un'alimentazione 12V al morsetto PWR e la scheda sarà in funzione. Se tutto è funzionante, si potrà procedere a collegare alle uscite ciò che si vorrà a gestire. Il Dip1 di DS1 durante il normale funzionamento deve essere lasciato in posizione OFF. Ricordiamo che nel caso vogliate fare a meno dei sensori crepuscolare e PIR in modo da riprodurre ciclicamente quanto previsto, sarà quindi sufficiente portare il Dip2 su ON e fare un ponticello tra SIG e 5V del morsetto "PIR". Con questo articolo speriamo di aver offerto tutti gli strumenti e le indicazioni per dar vita a un presepe moderno, ricco di effetti luminosi e sonori, senza rinunciare al fascino della tradizione. Buone Feste! □



COSA OCCORRE ?

Codice	Descrizione	Prezzo € IVA inclusa
FT1762M	Scheda controllo luci e audio per presepe montata, con Nano R4. <i>Il modulo MP3, il crepuscolare, e il sensore PIR sono disponibili separatamente.</i>	65,00
MP3PLAYERUSB	Modulo MP3/WAV con Memoria da 4MB	5,50
YB717	Modulo ultracompatto con PIR	5,50
Y171	Interruttore crepuscolare 12V	5,50

IL MATERIALE VA RICHIESTO A:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - www.futurashop.it

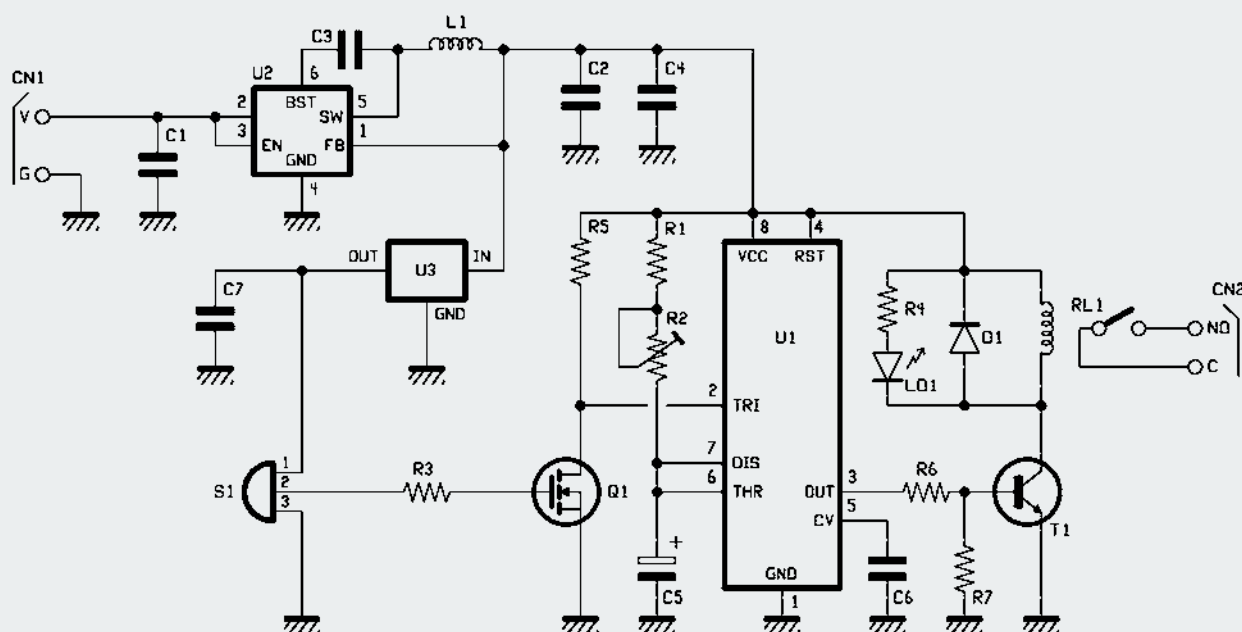
MODULO PIR A RELÉ



di **ALESSANDRO
SOTTOCORNOLA**

Perfettamente compatibile con i frutti per impianti elettrici BTicino Living Light e Vimar Arké, attiva un relé miniatura al suo interno quando rileva l'approssimarsi di persone nell'ambiente.

Da parecchi decenni gli impianti elettrici domestici sono modulari, ossia realizzati con scatole da incasso che ospitano gli elementi (prese, interruttori, deviatori ecc.) chiamati frutti, supportati da una placca che si fissa alla scatola da incasso e coperti, per ragioni estetiche e di protezione dall'introduzione di oggetti, mediante placche esterne variamente sagomate e rifinite. Da tempo, i "punti" degli impianti elettrici sono diventati il luogo ideale per ospitare dispositivi più avanzati rispetto ai comuni interruttori e deviatori. Questi nuovi dispositivi, che includono componenti elettronici, sono progettati per adattarsi alle scatole da incasso e alle placche di supporto dei più diffusi sistemi modulari, prodotti da aziende come Bticino, Vimar, Ave e Gewiss.



Tra questi troviamo sensori e attuatori per la domotica, spie e chiavi codificate per sistemi di allarme, oltre a dispositivi stand-alone, come quello descritto in queste pagine: un piccolo attivatore dotato di sensore a infrarossi passivi.

Più esattamente, il dispositivo integra un sensore PIR rivolto verso l'interno del locale, che quando rileva l'approssimarsi o il movimento di una o più persone comanda un'uscita a relé elettromeccanico tramite la quale, utilizzando i percorsi incassati dell'impianto elettrico, può triggerare l'ingresso di un impianto di allarme, accendere una lampadina, comandare un'elettroserratura, attivare un avvisatore acustico e fare altro ancora.

Il tutto temporizzato, nel senso che ad ogni evento prodotto dal sensore a infrarossi passivi l'uscita a relé viene attivata per un intervallo di tempo regolabile agevolmente tramite un trimmer.

SCHEMA ELETTRICO

Andiamo dunque a vedere in che cosa consiste questo speciale frutto, il cui circuito è tanto ridotto da poter essere montato in un tappo (copriforo) compatibile con i sistemi BTicino Living Light e Vimar Arké.

Lo schema elettrico del dispositivo è molto semplice, nonché basato su componentistica comune, perché si tratta di un multivibratore monostabile retriggerabile; il 555 è stato abbinato al sensore PIR AS312, che è uno dei più comuni sensori di

presenza a infrarossi utilizzato nelle lampade e/o nei faretto che si trovano in commercio. Il tutto viene alimentato a tensione continua di 12V tramite un alimentatore AC/DC da modulo frutto o da una linea a 12Vcc disponibile nell'impianto elettrico di destinazione; all'interno del circuito, un DC/DC converter switching abbassa i 12V a 5V per alimentare il 555 ed il relé di uscita, ed un ulteriore regolatore (stavolta LDO) ricava i 3,3V per il PIR. Andiamo dunque ad analizzare lo schema elettrico partendo dal sensore a infrarossi, siglato S1: si tratta di un dispositivo che incorpora un sensore piroelettrico preceduto da una lente di Fresnel, la cui prerogativa è dirigere opportunamente i raggi infrarossi corrispondenti al calore emanato da un corpo che si muove nel campo di intercettazione del sensore stesso, a prescindere dalla posizione. Per come funziona, il dispositivo PIR rileva lo spostamento di corpi capaci di irradiare calore e quindi persone e animali, ma anche veicoli a motore. La percezione del movimento si ottiene grazie alla disposizione in linea e a breve distanza di due sensori a infrarossi di cui si rileva la differenza di segnale dovuta allo "scorrimento" della fonte di calore su di essi.

Il sensore AS312 è un rilevatore di movimento digitale intelligente incapsulato in un package metallico tipo TO-18 ed integra tutti i circuiti elettronici, a partire da un A/D converter a 15 bit che campiona il segnale ottenuto dai trasduttori IR, un

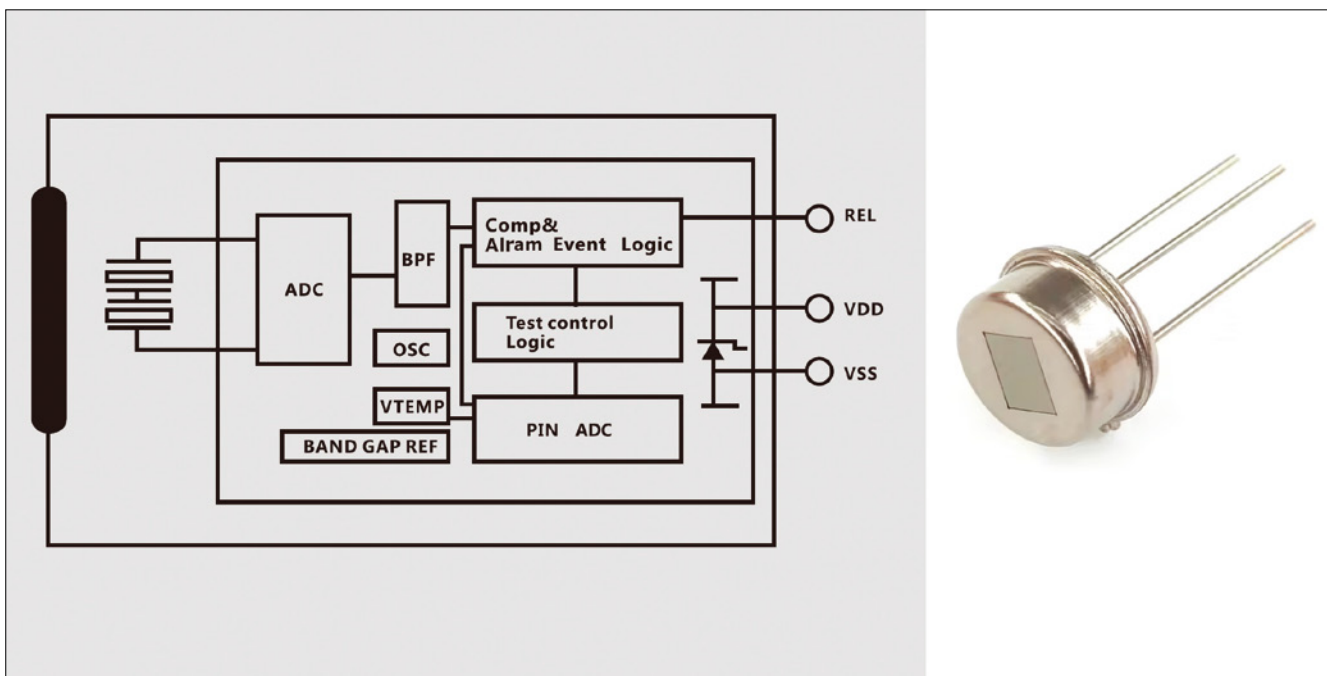


Fig. 1
Sensore PIR AS312:
schema
a blocchi
e aspetto.

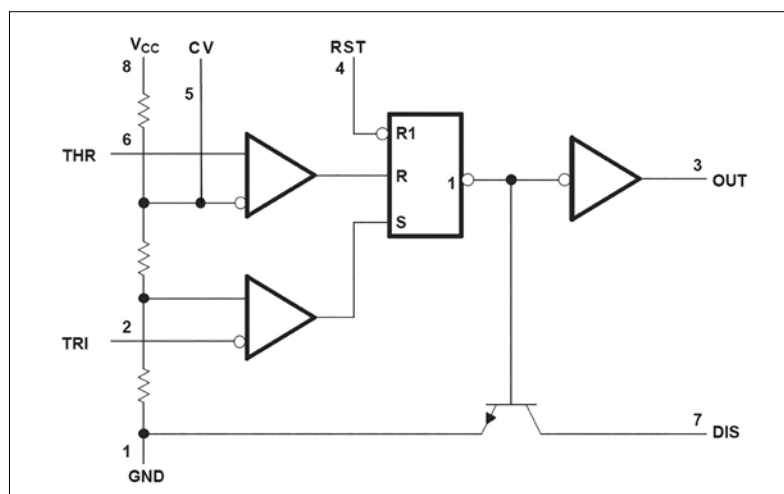
filtro passa-basso, fino al circuito di elaborazione e discriminazione; la logica interna determina la soglia oltre la quale il segnale digitalizzato è da considerarsi valido e tale da fornire un impulso all'uscita, con un tempo di ritardo sul ritorno a zero che può arrivare a circa 2 secondi, in base all'impostazione fatta in fabbrica. Nel nostro caso è di 2 secondi, che vanno a sommarsi alla durata del monostabile.

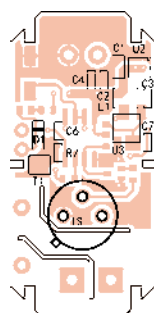
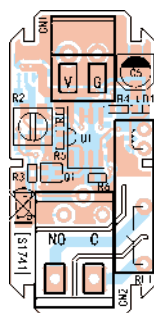
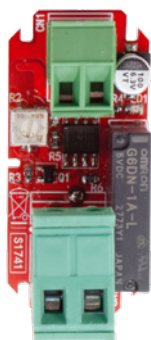
Ogniqualeva rileva il movimento di un corpo nel suo campo d'azione, l'AS312 fornisce un impulso di tensione positivo tramite il proprio piedino di uscita 2 (REL), la cui ampiezza è circa quella dell'alimentazione applicata tra i piedini 1 (positivo) e 3 (massa comune) che nel nostro caso ammonta a 3,3 volt in continua, ma in linea generale l'AS312 può funzionare a tensione compresa fra 2,7 e 3,3V. Per comprendere come funziona il sensore potete fare riferimento alla **Fig. 1**, che propone lo schema a blocchi dell'AS312 insieme all'aspetto esterno. Ogni impulso proveniente dal sensore raggiunge, attraverso la resistenza R3, il gate del MOSFET Q1, un dispositivo a canale N di tipo enhancement-mode. In condizioni di riposo, il MOSFET risulta interdetto tra drain e source, mentre quando riceve una tensione tra gate e source dell'ordine di 2-3V, diventa praticamente un cortocircuito. È importante notare che la funzione della resistenza R3 non è quella di limitare la corrente di gate poiché, almeno in condizioni stazionarie, non vi è assorbimento.

Piuttosto, essa serve a evitare picchi di corrente sull'uscita logica del sensore PIR nel momento in cui il segnale passa da livello basso a livello alto. In questo breve intervallo, infatti, la natura capacitiva del gate causa un temporaneo comportamento di cortocircuito.

Il MOSFET a canale N lo utilizziamo come inverter logico per comandare l'ingresso di trigger del timer 555: ogniqualvolta l'uscita del sensore si porta a livello alto, il suo drain va a zero logico e triggera il monostabile; vediamo subito perché ciò avviene, analizzando struttura interna e funzionamento del 555 con l'aiuto dello schema proposto dalla **Fig. 2**.

Fig. 2
Schema interno del
timer 555, che nel
circuito utilizziamo
come monostabile.





Elenco Componenti:

- R1: 10 kohm
- R2: Trimmer 500 kohm 4.5x4.5 mm
- R3: 10 kohm
- R4: 470 ohm
- R5: 100 kohm
- R6: 4,7 kohm
- R7: 10 kohm
- C1: 10 μ F 16V ceramico
- C2, C4: 22 μ F 6,3V ceramico
- C3, C7: 100 nF ceramico
- C5: 100 μ F 6,3 VL elettrolitico
- C6: 10 nF ceramico
- S1: AS312
- D1: MMSD4148
- LD1: LED rosso
- U1: NE555DR
- U2: AP63205WU
- U3: MCP1700T-3302E/TT
- L1: Bobina 4,7 μ H 1.6A AOTA-B252008S4R7MT
- Q1: BSS138W
- RL1: Relè SPST-NO 250Vac/5A G6DN-1A-L DC5
- T1: BC847BW

Varie:

- Lente Fresnell per AM312
- Morsetto Innesto 2 vie verticale passo 5.08 mm
- Connettore Innesto 2 vie verticale passo 5.08 mm
- Morsetto Innesto 3 vie verticale passo 3.81 mm
- Connettore Innesto 3 vie verticale passo 3.81 mm
- Circuito stampato S1741 (40x19 mm)

All'interno dell'integrato troviamo due operazionali configurati come comparatore di tensione ed aventi in comune un partitore di tensione multiplo che polarizza per l'operazionale superiore l'ingresso invertente e per quello inferiore, il non-invertente; dei comparatori vengono resi accessibili dall'esterno gli ingressi non-invertente di quello superiore (i termini superiore e inferiore si riferiscono al potenziale di riferimento che ricevono dal partitore multiplo, quindi quello superiore riceve la tensione più elevata), corrispondente al piedino 6 (THR, ossia Threshold) e l'invertente di quello superiore, corrispondente al piedino 2 (TRI, ossia Trigger). Il nodo tra la prima e la seconda (considerate dall'alimentazione positiva) resistenza del partitore multiplo di riferimento dei due comparatori (alimentato, come il resto degli stadi interni al 555, dal piedino 8) viene portato all'esterno tramite il piedino 5 (CV, Control Voltage) e ciò permette di alterare le tensioni di riferimento dei comparatori, così da controllare con una tensione esterna la frequenza di lavoro nella configurazione astabile (operando lo shift di frequenza come si fa nei VCO, ossia gli oscillatori controllati in tensione) oppure la durata degli impulsi in quella da timer (monostabile). Più esattamente, nella configurazione astabile, abbassando la tensione sul piedino 5 si accorcia il periodo della commutazione dell'uscita e quindi si incrementa la frequenza generata, viceversa lasciando salire la tensione fino al livello caratteristico (a vuoto, ossia senza alcun circuito collegato al piedino 5) o incrementandola si allunga il periodo e si riduce la frequenza prodotta. Nella configurazione monostabile, invece, la tensione altera la durata degli impulsi, il che permette, pilotando il trigger con impulsi a frequenza costante, di ottenere un modulatore PWM; nel nostro caso lasciamo CV filtrato dal condensatore C6, terminante a massa perché non ci interessa tale funzione.

Le uscite dei comparatori del 555 entrano una nel RESET (quella del comparatore superiore) e nel SET (comparatore inferiore) di un flip-flop di tipo RS, che è un circuito logico la cui uscita diretta (Q) va a livello alto (corrispondente a circa il potenziale del piedino 8) quando il SET è a livello alto, ovvero assume lo zero logico se SET è posto a circa zero volt; l'uscita presenta un comportamento esattamente opposto quando ad essere stimolato è l'ingresso RESET, il quale, posto a livello logico alto, resetta il flip-flop, intendendo con ciò che ne azzerà lo stato di Q. Il flip-flop ha anche un'uscita complementata ($\bar{Q}$), la cui condizione logica è sempre l'inverso di quella Q; nel 555 è collegata, tramite una resi-

stenza, alla base di un transistor NPN configurato ad open collector il cui emettitore è connesso a massa e il collettore al piedino DIS (Discharge, 7). Il transistor può commutare una corrente di collettore di 200 mA, utilizzato in modo sink (ad assorbimento di corrente). L'uscita diretta, Q, è invece collegata al piedino OUT (3) dell'integrato attraverso un buffer interno push-pull, capace di erogare un massimo di 200 mA.

Il negativo di alimentazione, vale a dire la massa di riferimento del 555 corrisponde al piedino 1, mentre al 4 è connesso il reset a logica invertita del flip-flop: questo piedino permette di resettare forzatamente il circuito dall'esterno applicando un livello logico basso, ma nel nostro caso lo poniamo fisso a livello alto (lo stesso potenziale dell'alimentazione fornito al piedino 8) perché non ci serve la funzione di reset esterno.

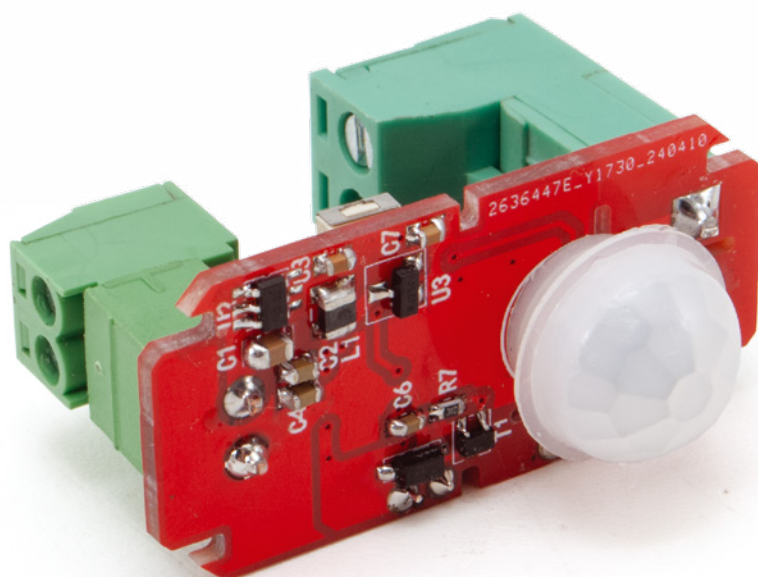
Ciò detto andiamo a vedere come tutto questo circuito funziona nella modalità monostabile, la quale consiste nell'unire i piedini 6 (THR) e 7 (DIS) portandoli a massa con un condensatore (nel nostro caso l'elettrolitico C5) ed al positivo di alimentazione con una resistenza, che è composta dalla serie di un trimmer (R2) e un resistore fisso (R1): il trimmer ci consentirà di variare a piacimento, entro certi limiti, il tempo del monostabile, essendo, quella appena descritta, la rete che determina la durata di ogni impulso all'uscita conseguente all'impulso negativo sul piedino di trigger.

Partendo dalla condizione in cui il condensatore C5 sia scarico e supponendo il MOSFET Q1 interdetto,

CARATTERISTICHE TECNICHE

❖ Tensione di alimentazione: 9÷30Vcc ❖ Assorbimento max: 30 mA ❖ Uscita a relé: 250Vca - 5A	❖ Modalità uscita: monostabile ❖ Tempo attivazione uscita: 4÷55 s ❖ Compatibile BTicino Living Light e Vimar Arké
---	---

il comparatore superiore ha l'ingresso non-invertente a un potenziale inferiore rispetto all'invertente, portando la sua uscita a livello basso e liberando così l'ingresso R del flip-flop. Il comparatore inferiore, al contrario, ha l'ingresso invertente a un potenziale positivo rispetto al non-invertente, risultando anch'esso con uscita a livello basso, e quindi non interviene sul flip-flop poiché l'ingresso S è a zero logico. In questa situazione, l'uscita del flip-flop rimane a zero e quella del 555 a livello alto, mentre il transistor di discharge è interdetto, consentendo a C5 di caricarsi. Questo processo continua finché il potenziale del piedino 6 supera quello dell'ingresso invertente del comparatore superiore, facendo sì che l'ingresso R del flip-flop

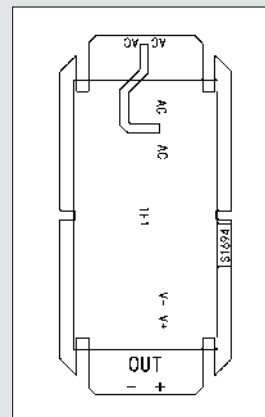
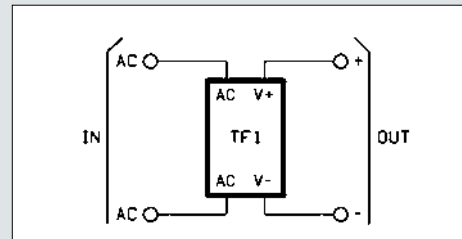


← Il sensore PIR va completo applicandogli la lente di Fresnel a cupola.

Fermo restando che il modulo PIR qui descritto e quelli che pubblicheremo della stessa serie, può essere alimentato da una qualsiasi fonte di tensione continua di valore compreso fra 9 e 30 volt, abbiamo pensato bene di creare un frutto dedicato all'alimentazione, che accetti in ingresso la tensione alternata fornita dalla rete di distribuzione domestica a 230 Vca ed eroghi dall'uscita 12Vcc. Questo modulo è sostanzialmente composto da un unico elemento, ossia un alimentatore switching AC/DC galvanicamente isolato, incapsulato in un contenitore plastico e dotato di due piedini per l'ingresso di rete e altrettanti (+ e -) per l'uscita a tensione continua; questo modulo si monta su un circuito stampato dedicato che riporta le piazzole cui collegare (saldare) i cavi di entrata ed uscita. Il componente che contiene l'intero alimentatore switching è l'HLK-2M12, di cui potete consultare il datasheet da Internet, consultando la pagina web <https://www.hlktech.net/index.php?id=1159>. Questo switching ultracompatto è in grado, quando alimentato a 230 Vca, di fornire in uscita una potenza di 2 W su 12 volt, corrispondenti a una corrente continua di 166 mA.

Lo schema elettrico del modulo frutto di alimentazione basato sull'HLK-2M12 è proposto in questo riquadro e in esso si nota l'estrema semplicità circuitale: in pratica c'è solo il modulo con le sue connessioni verso l'esterno.

Con la corrente erogabile dall'HLK-2M12 è possibile far funzionare tre moduli Vimar/BTicino senza particolari limitazioni. Chiaramente il form-factor (ossia forme e dimensioni...per usare un termine anglosassone...) di questo modulo è tale da poterlo introdurre all'interno del solito copriforo Vimar o BTicino, in maniera da collocare l'alimentatore all'interno della stessa scatola (o di una in collegamento tramite una guaina dell'impianto elettrico) dove si trovano gli altri moduli da alimentare, ovviamente anche quello PIR descritto in questo articolo.



passi a livello alto e resettando così il flip-flop RS. Ora l'uscita del 555 si porta a zero e il transistor di discharge scarica nuovamente il condensatore C5. Quando si trascina a massa (o comunque al disotto di 1/3 della tensione di alimentazione) per un istante il piedino 2, l'uscita del comparatore inferiore passa a livello alto e setta il flip-flop, la cui uscita va a livello basso lasciando interdire il transistor di discharge e facendo andare a livello alto il piedino 3 del timer; C5 riprende quindi a caricarsi con un tempo (t) pari a:

$$t = 1,1 (R1+R2) C5$$

In sostanza l'uscita OUT dell'integrato si mantiene a livello alto nel periodo che serve al C5 affinché la tensione ai suoi capi arrivi a 2/3 della tensione di alimentazione, ovvero a far commutare nuovamente il comparatore superiore e attivare così l'input R del flip-flop e a riportare a livello basso il

piedino 3 del 555; essendo R2 un resistore variabile, vediamo che è possibile modificare la durata dell'impulso all'uscita del monostabile. Più esattamente, ruotando il cursore del trimmer fino ad escluderne la resistenza il tempo t vale:

$$t_{min} = 1,1 \times R1 \times C5$$

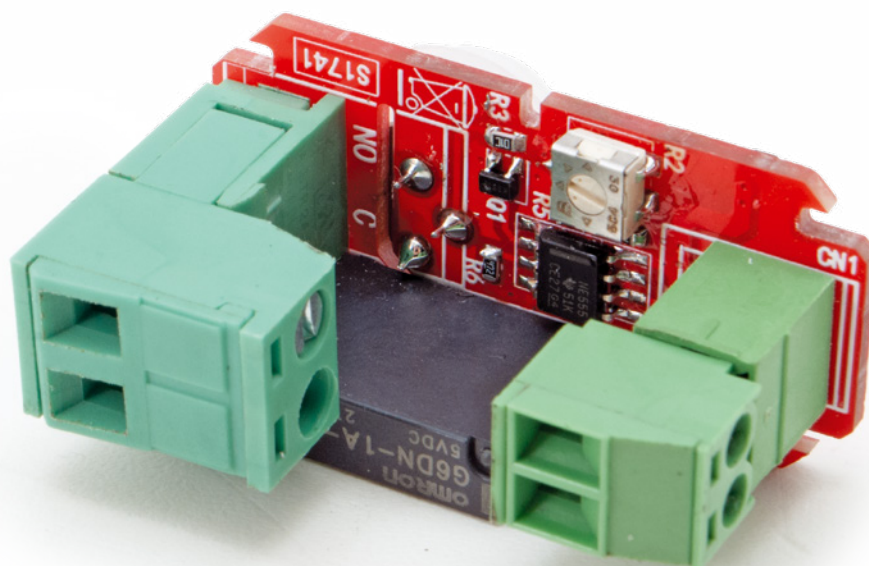
che sostituendo i valori dei componenti equivale a:

$$t = 1,1 \times 10 \text{ k}\Omega \times 100 \mu\text{F} = 1,1 \text{ s}$$

Invece inserendo la massima resistenza del trimmer otteniamo una durata dell'impulso pari a:

$$t_{max} = 1,1 \times 510 \text{ k}\Omega \times 100 \mu\text{F} = 56 \text{ s}$$

Gli intervalli suddetti si intendono considerati dal momento in cui il drain del MOSFET ritorna ad un potenziale oltre i 2/3 di quello del piedino 8 e con



Il prototipo dell'elettronica del modulo PIR.

esso il piedino 2, quindi gli va sommato il ritardo dell'uscita del PIR; ecco perché il tempo più corto (t_{min}) che è quello sul quale il ritardo nel ritorno a riposo dell'uscita del PIR ha maggiore rilievo, è poco meno di 4 secondi.

L'uscita del monostabile pilota un relé in miniatura con contatto di scambio (C e NO) attraverso il transistor NPN T1, il cui funzionamento è controllato dalla polarizzazione della giunzione base-emettitore mediante un partitore composto dalle resistenze R6 e R7. In antiparallelo alla bobina del relé è stato inserito un diodo che serve a dissipare le extratensioni inverse generate dall'induttanza della bobina stessa quando il transistor si interdice. Il diodo ha lo scopo di evitare che la tensione inversa si chiuda sulla giunzione base-emettitore, danneggiandola irrimediabilmente. Sempre in parallelo alla bobina del relé RL1 è stato posizionato un LED in serie a una resistenza limitatrice di corrente, che permette di visualizzare l'attivazione del relé quando viene attivato dal sensore PIR. Concludiamo l'analisi del circuito con la parte di alimentazione, che vede in campo un alimentatore DC/DC d'ingresso ed un regolatore lineare ad alimentare il PIR.

Il regolatore switching è basato sull'integrato AP63205 della DIODES, contenuto in un package per SMD a 3 pin per lato, di tipo TSOT-26. Questo componente, siglato U2 nello schema elettrico, fornisce una tensione d'uscita stabilizzata a 5V, accettando in ingresso da 3,8 a 32Vcc, erogando fino a 2A in regime continuativo, senza richiedere alcun dissipatore di calore; l'AP63205 è un rego-

latore buck operante a una frequenza di commutazione di 1 MHz che si accontenta di pochissimi componenti esterni, tra i quali un piccolo induttore, giacché integra una coppia di MOSFET per pilotare la bobina, uno high-side (sul lato di alimentazione) con $R_{ds(on)}$ di appena 125mΩ e uno low-side (che commuta l'uscita verso massa) da soli 68mΩ. Nonostante funzioni a una frequenza di commutazione molto elevata (allo scopo di minimizzare valori e dimensioni dell'induttanza e dei condensatori di filtro) irradia un bassissimo livello di interferenze, grazie a opportune strategie di riduzione delle EMI (Electromagnetic Interference) che comprendono la tecnica Frequency Spread Spectrum (FSS) con un jitter della frequenza del $\pm 6\%$.

Il nostro converter switching è basato sulla tecnica a carica d'induttanza, il cui principio di funzionamento si basa sulla ripartizione dell'energia immagazzinata da un induttore; più esattamente, un transistor alimenta una bobina, quindi interrompe la corrente e lascia che l'energia immagazzinata in essa si riversi sull'utilizzatore. Il livello della tensione di uscita si decide, non utilizzando il trasformatore ma giocando sulla larghezza o frequenza degli impulsi con i quali viene caricato l'induttore e sul valore di induttanza.

Nel caso dell'integrato della DIODES viene adottato uno schema di converter DC/DC molto utilizzato da parecchi anni che vede la bobina alimentata e chiusa alternativamente attraverso uno stadio a transistor a simmetria quasi complementare, dove i MOSFET, entrambi a canale N, vengono pilotati sul gate da impulsi in opposizione di fase in modo

che quando uno conduce l'altro è spento. Nella configurazione buck, il MOSFET che opera a drain comune alimenta la bobina per un semiperiodo e quello connesso a massa (funzionante a emettitore comune) la chiude in cortocircuito. Un diodo impedisce il flusso della corrente nel verso contrario, consentendo comunque la carica del condensatore, che nel nostro circuito è il parallelo di C2 e C4.

La struttura interna dell'AP63205 è meglio descritta dallo schema a blocchi dell'integrato proposto nella **Fig. 3**, dalla quale si vede, in basso a destra, la composizione dello stadio di commutazione che pilota l'induttore.

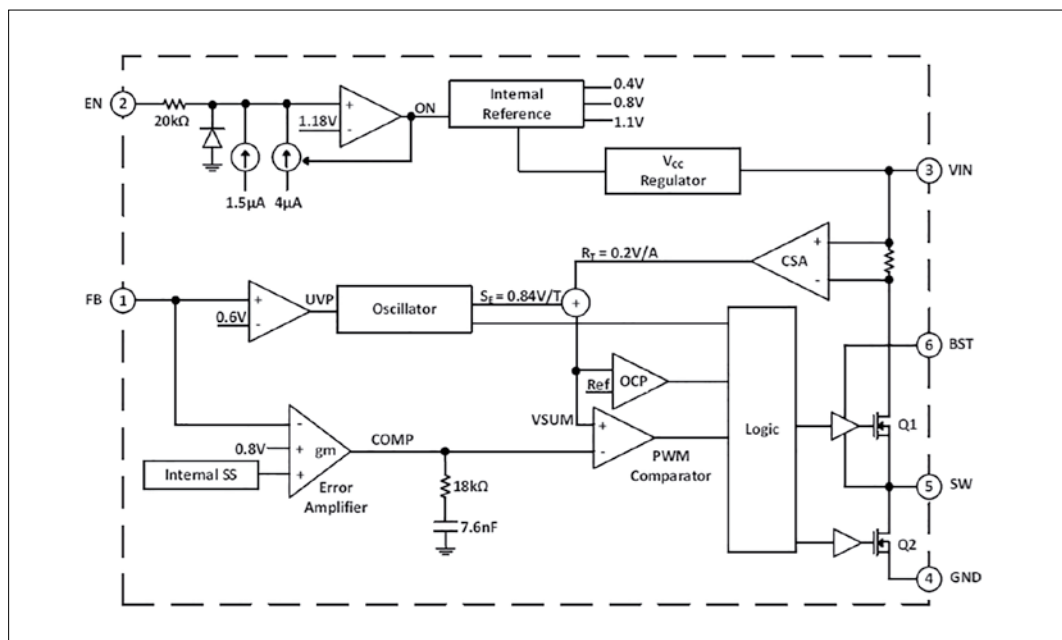
Nel circuito, l'integrato AP63205 opera in una configurazione classica, dove l'induttanza L1 viene alimentata e scaricata ciclicamente (fare riferimento allo schema a blocchi in **Fig. 3** per una rappresentazione dettagliata) tramite lo stadio a MOSFET a simmetria complementare, collegato al piedino SW e retroazionato dal condensatore C3, che invia il segnale di commutazione al piedino FB (feedback). Gli impulsi di corrente generati dall'induttanza durante la scarica attraverso il MOSFET low-side caricano i condensatori C2 e C4, che sono di piccola capacità, anziché elettrolitici, data l'elevata frequenza di commutazione in gioco.

La tensione di uscita si preleva proprio ai capi di detti condensatori, mentre quella di ingresso del regolatore viene applicata al piedino Vin ed all'EN: quest'ultimo, se portato a massa permetterebbe

di mantenere spento lo switching pur alimentato, ma nel nostro caso lo manteniamo al potenziale di alimentazione perché ci occorre un regolatore che funzioni permanentemente, ovvero non ci serve spegnerlo o porlo in standby.

Notate che la scelta di adottare uno switching come regolatore in entrata nasce dall'esigenza di supportare un ampio campo di tensione senza soffrire delle perdite di potenza caratteristiche di un regolatore lineare, il cui difetto è che all'aumentare della differenza tra la tensione d'ingresso e quella d'uscita (tecnicamente detta drop-out...) aumenta linearmente la potenza sprecata, che viene dissipata sotto forma di calore, con conseguente perdita di efficienza energetica del regolatore ed incremento dell'ingombro del circuito dovuto al dissipatore. Il bello dello switching è che opera sui parametri della potenza mantenendo, a prescindere dalla caduta di tensione ingresso-uscita, una dissipazione di potenza che dipende solo dalla corrente erogata. Quanto all'alimentazione del sensore a infrarossi passivi, la ricaviamo con un regolatore lineare LDO (Low Drop-Out) di tipo MCP1700T-3302E/TT e anch'esso in esecuzione SMT, collegato all'uscita dell'U2; il regolatore è a tre terminali e ricava i 3,3 volt occorrenti al PIR, opportunamente filtrati dal condensatore C7. Il regolatore sopperisce abbondantemente ai bisogni energetici del sensore PIR perché può erogare 250 milliampere, a fronte di un assorbimento tipico dell'AS312 che non supera la decina di mA.

Fig. 3
Schema a blocchi
del regolatore
DC/DC AP63205.



L'alimentazione dell'intero circuito si applica alla morsettiera siglata CN1, i cui contatti V e G corrispondono ai fili di uscita del modulo alimentatore a frutto che abbiamo pensato, ma che non l'unica soluzione per l'alimentazione; il condensatore C1 serve a filtrare eventuali spurie di commutazione sfuggite all'integrato switching AP63205 lungo la linea di output.

REALIZZAZIONE PRATICA

Ora che abbiamo descritto tutti gli aspetti necessari dello schema elettrico, possiamo dedicare qualche paragrafo alla spiegazione della realizzazione del sensore. A tal fine, abbiamo progettato un circuito stampato a doppia faccia, il cui file Gerber è disponibile per il download gratuito sul nostro sito (elettronica.in.it) e può essere realizzato tramite il servizio di produzione PCB low-cost offerto da ftrlink.it/pcb.

Naturalmente per poterlo miniaturizzare abbastanza da introdurlo in un coperchio per impianti elettrici modulari abbiamo dovuto utilizzare anche componenti SMD. Per il montaggio procuratevi del filo di lega saldante molto sottile (tipicamente 0,5 mm) e un saldatore a punta fine, oltre a una lente d'ingrandimento e una pinzetta per posizionare ben centrati nelle rispettive piazzole l'integrato, la bobina e i quattro condensatori; meglio è se avete della pasta fluxante di media densità da spalmare con un pennellino sulle piazzole prima di deporvi i componenti, perché agevolerà la saldatura consentendovi di limitare il tempo di permanenza della punta del saldatore e quindi il riscaldamento dei vari elementi.

Comunque sia, una volta ottenuto il PCB disponetevi i pochi componenti occorrenti, che sono tutti per montaggio superficiale ad eccezione delle

morsettiere, del relé e del sensore AS312; iniziate disponendo le resistenze fisse e il trimmer e ai condensatori, partendo dai non polarizzati e finendo con l'unico elettrolitico, per il quale ricordate di rispettare la polarità.

Montate quindi le morsettiere bipolari da c.s. (a passo 5 mm) e il relé miniatura, poi passate dal lato opposto, da dove introdurrete il sensore a infrarossi passivi saldandone i tre terminali dal lato componenti del circuito stampato, tagliandone l'eccedenza; al sensore dovette quindi applicare, premendola fino ad incastrarla, la cupolina che costituisce la lente di Fresnel.

Una volta completato il montaggio e verificato con attenzione che non vi siano saldature fredde o "baffi" di stagno a creare cortocircuiti tra piazzole attigue, il vostro modulo è pronto per l'uso, giacché non richiede taratura se non la regolazione del tempo di attivazione dell'uscita a relé mediante il trimmer R2, che farete dopo aver collegato la scheda all'alimentazione ed aver provato a passare la mano di fronte al PIR.

INSTALLAZIONE E UTILIZZO

Una volta completato, il modulo va inserito nel suo contenitore. L'ideale sarebbe utilizzare un copriforo per impianti elettrici modulari come quelli prodotti da Bticino o Vimar, per i quali il circuito stampato è stato progettato (come illustrato in **Fig. 4**). In alternativa, è possibile realizzare un contenitore apposito stampato in 3D, utilizzando materiale ABS. La parte anteriore di tale contenitore va forata con una punta da trapano (asportando poi eventuali sbavature) così da far uscire il sensore PIR affinché possa rilevare l'approssimarsi delle persone. Il circuito stampato, una volta completato, va quindi introdotto dalla parte della cupolina con la lente

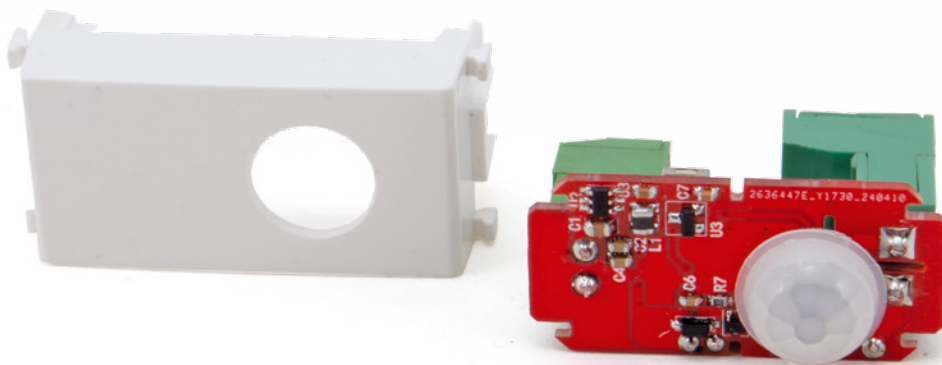


Fig. 4
Il copriforo Vimar forato per ospitare il circuito inserito dal lato del sensore.



Fig. 5
Il modulo così come
si presenta inserito
nel copriforo Vimar.

di Fresnel, in modo da ottenere un insieme come quello proposto nella **Fig. 5**.

Una volta fornita l'alimentazione, se viene rilevato dal sensore PIR l'approssimarsi di una persona o un animale in movimento nel locale dove si trova installato, viene attivata l'uscita a relé per il tempo impostato.

Rammentate che tale tempo è regolabile tramite il trimmer R2, che va registrato però prima di introdurre il modulo nella placca di supporto e di chiudere il punto luce corrispondente.

Durante il funzionamento, ogni volta che l'uscita si attiva, il LED rosso sul circuito stampato si illumina, indicando che lo scambio del relé miniatura si è chiuso e con esso il circuito elettrico che vi sarà collegato.

A proposito di alimentazione, il circuito è molto tollerante perché accetta in ingresso una tensione continua di valore compreso fra 9 e 30 Vcc (merito

del convertitore DC/DC posto in entrata) ed assorbe una corrente media di 30 milliampere, prelevabile quindi da fonti di alimentazione locali o di altri apparati o sistemi presenti nell'impianto elettrico in cui il modulo verrà installato.

Quanto al modulo alimentatore, è un frutto contenente un "trasformatore elettronico" vale a dire un alimentatore AC/DC switching capace di erogare in uscita 12Vcc e 166 mA di corrente, per un totale di 2W di potenza, ammettendo di applicare in entrata ad esso una tensione alternata di valore compreso fra 100 e 240 Vca.

L'uscita a relé del circuito si può utilizzare per una miriade di scopi: pilotare un ingresso dell'impianto di allarme di casa (in modo da farlo scattare in presenza di persone in un locale) o accendere una lampadina, far scattare l'elettroserratura di un cancelletto elettrico, per esempio; ma ci sono tante altre applicazioni che, siamo certi, saprete trovare. Le connessioni dello scambio del relé miniatura su circuito stampato e dell'alimentazione continua di ingresso sono implementate tramite due morsettiere con morsetti ad angolo retto, per potervi accedere da dietro e comunque per potervi introdurre e serrare i cavi anche quando il circuito stampato è già stato introdotto nel suo contenitore o copriforo che sia.

CONCLUSIONI

Bene, termina qui la descrizione di un progetto che riteniamo molto interessante e in linea con la tendenza ad automatizzare abitazioni ed edifici, integrando nel tradizionale impianto elettrico funzionalità e dispositivi che ne consentono l'implementazione.

In particolare, il modulo PIR qui descritto è un sensore multifunzione con uscita temporizzata a durata regolabile, che permette sia di attivare utilizzatori elettrici (elettroserrature, tornelli, attivatori acustici ed ottici, lampade ecc.) sia di inviare un segnale di comando a una centralina o ad un sistema elettronico remoto, quale ad esempio l'impianto di allarme o di videosorveglianza e videoregistrazione delle immagini dei locali in cui il modulo è stato installato.

Insomma, il nostro circuito è un generico sensore di movimento con uscita a relé temporizzata, che va inteso come dispositivo di input di vari apparati e sistemi, o anche impiegato come controllo autonomo e diretto di utilizzatori elettrici.

Lasciamo a voi la scelta su come e su dove impiegarlo, certi che lo farete al meglio.

Buon lavoro!



COSA OCCORRE ?

Codice	Descrizione	Prezzo € IVA inclusa
2WMA22012V	Modulo alimentatore 220VAC/12VDC	4,60

I componenti necessari per realizzare questo progetto sono disponibili nei principali negozi di componentistica elettronica.
Il circuito stampato può essere creato scaricando il file Gerber dalla pagina di presentazione di questo numero e utilizzando il servizio PCBPRODUCTION, disponibile all'indirizzo ftrlink.it/PCB.

IL MATERIALE VA RICHIESTO A:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - www.futurashop.it

Non lasciare la tua casa incustodita!

Allarme evoluto WiFi-GSM/GPRS controllabile da remoto



€119,00

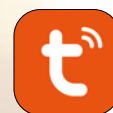
Cod. FR783



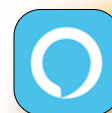
Controllabile con:



Smart Life



Tuya Smart



Amazon Alexa



Google Home

Centralina antifurto wireless programmabile, controllabile da remoto tramite Smartphone sfruttando la rete GSM/GPRS o Wi-Fi e direttamente con Amazon Alexa o Google Home. Le App "Smart Life" o "Tuya Smart" permettono di configurare e gestire il funzionamento dell'allarme in ogni suo dettaglio. È possibile aggiungere sensori ed accessori opzionali per coprire fino a 99 zone. La confezione comprende: la centralina, due radiocomandi, un sensore wireless per porte/finestre, un sensore PIR wireless, due chiavi TAG RFID, la batteria di backup e l'alimentatore di rete. Manuale di utilizzo in italiano.

Accessori opzionali:

Prezzi IVA inclusa



Tastiera wireless	Sensore PIR wireless da esterno	Sensore PIR wireless	Sensore porte/finestre wireless	Transponder a portachiavi	Sensore di fumo wireless	Radiocomando supplementare	Sirena wireless da interno	Sirena wireless da esterno da pannello solare
								
€32,00	€34,00	€14,00	€11,00	€5,00	€19,00	€9,00	€29,00	€72,00
Cod. FR783KEY	Cod. FR783PIREXT	Cod. FR783PIR	Cod. FR783MAGN	Cod. FR783TAG	Cod. FR783SMOKE	Cod. FR783RAD	Cod. FR783SIR	Cod. FR783SIREXT

INDISPENSABILI IN LABORATORIO!



Tappeto in silicone per saldatura 550x350 mm

PER SALDARE

Tappeto in silicone antistatico multifunzione per saldatura resistente al calore fino a una temperatura di 500°C. Estremamente flessibile, anche dopo piegato ritorna sempre alla sua forma originale. Dispone di 3 piccoli contenitori e di 5 aree magnetiche.



cod. AS12

€ 35,00



Tappetino antistatico 550x300x2 mm

Tappetino antistatico particolarmente indicato per posizionare e manipolare componenti o schede elettroniche. Viene fornito completo di cavo di collegamento messa terra. Resistività superficiale: $10^7 \sim 10^{10}$ ohm/m², tempo decadimento carica statica: <0,03 secondi.

cod. AS13

€ 13,50

ANTISTATICO



Tappetino da taglio 300x220x2,8 mm

PER TAGLIARE

Tappetino da taglio realizzato in PVC (Polivinil-cloruro) ideale come piano di lavoro per tagliare carta, cartoncino, cartone pressato, fogli di vinile, ecc. Dispone di griglia di misurazione 1x1 cm e goniometro che consente all'utente di eseguire segni e tagli precisi.



cod. CUTMAT30X22

€ 9,00



Scopri la gamma completa su www.futurashop.it

LC METER IN-CIRCUIT

Misuriamo la capacità e l'induttanza direttamente sul circuito.

di **MICHAEL MONKENBUSCH**

Quando si esegue una diagnostica su una scheda elettronica, la capacità di testare i componenti passivi senza dissaldarli dal PCB è estremamente utile, ma richiede attenzione per ridurre al minimo le interferenze.

Questo prototipo consente di misurare capacità in-circuit da 1 pF a 80 mF e induttanze da 1 mH a 40 H. Il dispositivo utilizza un metodo a bassa tensione, efficace anche in presenza di resistenze parassite sia in parallelo sia in serie.

L'elaborazione del segnale è principalmente analogica, mentre i calcoli sono eseguiti da un Arduino UNO.

Tempo fa ho utilizzato un generatore di funzioni per valutare rapidamente un condensatore, misurando la corrente che lo attraversava a partire dalla forma d'onda della tensione applicata, con l'ausilio di un oscilloscopio. L'installazione iniziale era piuttosto rudimentale,

poiché utilizzavo un piccolo resistore di shunt in serie con il condensatore per ricavare il valore della corrente. In questa configurazione, una forma d'onda triangolare di tensione presenta un vantaggio: durante le rampe costanti di salita e discesa della tensione applicata (idealmente), la corrente attraverso il condensatore assume un valore costante (positivo o negativo), proporzionale alla capacità C (vedi il riquadro **metodo di misurazione**).

Una corrente addizionale $I_p(t) = U(t)/R_p$, dovuta a una resistenza in parallelo (di perdita) R_p , risulterebbe semplicemente proporzionale alla tensione triangolare applicata $U(t)$.

Se la tensione applicata varia tra valori negativi e positivi, essa sarà zero in alcuni istanti t_0 durante le rampe costanti di salita e discesa della forma d'onda triangolare. In questi istanti, la corrente resistiva parassita $IR(t_0) = 0$, e la tensione

misurata in questi punti rappresenta solo C ed è indipendente da R_p . In pratica, la misura può essere eseguita campionando la corrente nei momenti in cui $U(t) = 0$.

Questa era l'idea iniziale per creare un circuito semplice per misurare la capacità, che fornisse una tensione proporzionale a C e indipendente da perdite o altre resistenze in parallelo. Un'analisi più approfondita del circuito ha evidenziato la possibilità di estendere la configurazione per stimare il valore della resistenza in parallelo R_p e per valutare la presenza di una resistenza in serie, il tutto in una singola operazione. Inoltre, scambiando il condensatore C con un induttore L , la tensione applicata $U(t)$ con una corrente applicata $I(t)$, e invertendo i ruoli delle resistenze in parallelo e in serie, uno schema simile può essere utilizzato per misurare gli induttori. La misurazione della corrente viene sostituita con una misurazione della tensione sull'induttore, anziché la misurazione della tensione su uno shunt di corrente come avviene nella modalità di misura della capacità.

Queste caratteristiche rendono allettante l'uso del metodo anche per componenti in-circuit, senza la necessità di dissaldarli. Per rendere ciò praticabile, si può osservare che, se non alimentati, quasi tutti i componenti a semiconduttore espon-

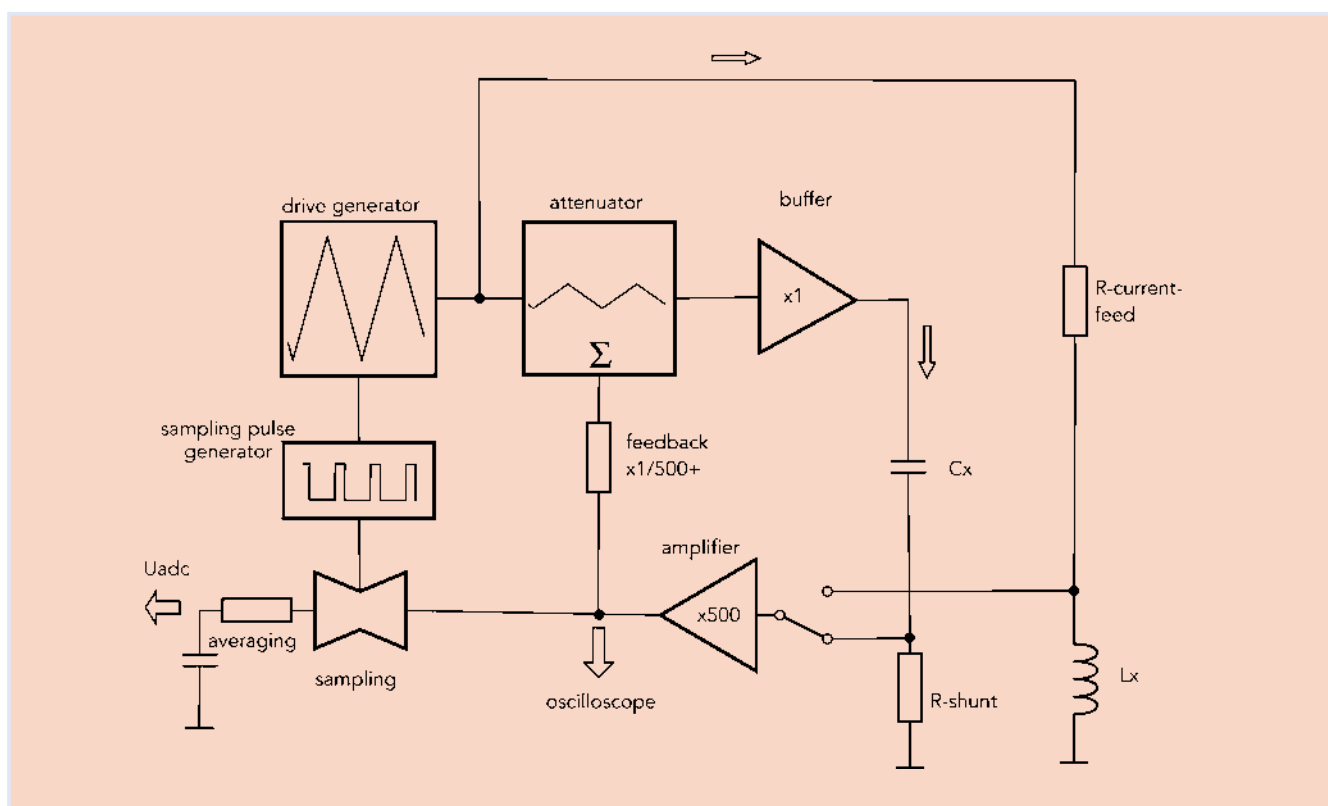


Fig. 1 Diagramma a blocchi del circuito.

gono, nel peggiore dei casi, giunzioni a diodo, con il rischio di generare percorsi di corrente parassita aggiuntivi. Se la tensione applicata rimane inferiore a circa 100 mV, le correnti nei diodi sono trascurabili. Nel caso degli induttori, la tensione risultante deve rimanere sotto questo valore. Tensioni così basse permettono anche di ignorare la polarità dei condensatori elettrolitici.

Nel circuito presentato, le tensioni su shunt e induttore restano inferiori a 10 mV. Il campionamento sincrono e l'averaging rendono la misura robusta contro le influenze di sorgenti di rumore interne ed esterne.

DIAGRAMMA A BLOCCHI

Il diagramma a blocchi del circuito è mostrato nella **Fig. 1**. Come elementi funzionali, il circuito contiene un generatore di una forma d'onda triangolare di tensione definita, comparatori e logica per generare impulsi, l'amplificatore di segnale A e lo stadio di campionamento con amplificatori buffer (non inseriti nel diagramma). Il collegamento con Arduino include un ADC a 16 bit a quattro canali ADS1115 della Texas Instruments. Gli ingressi analogici Ax del microcontrollore vengono utilizzati per determinare le posizioni degli switch di gamma, e le uscite digitali controllano un numero di interruttori

MOS per supportare un parziale auto-range. In questo modo, lo sketch Arduino può valutare la misurazione e visualizzare i valori di C o L e possibili resistenze parassite (effettive) con alcune ulteriori informazioni ausiliarie su un display LCD.

SCHEMA ELETTRICO

Il circuito (parte analogica) è mostrato nella **Fig. 2**. Contiene i seguenti gruppi:

- Tensioni di riferimento (U4, U11).
- Generatore di tensione a forma d'onda triangolare (U12A...D).
- Generazione della finestra di campionamento (U2A, U2B, U3A, U3B, U13A, U13B).
- Amplificatore di segnale (U16A, U16B).
- Inversione sincrona della polarità (Q8, U1B).
- Campionamento, averaging (U5B, U5C, U5D; C2, C3, C4; U5A) e buffering (U1C, U1D).

La tensione triangolare viene creata tramite integrazione (U12A), accumulando su C1 una corrente determinata dalla resistenza connessa tramite l'interruttore SW2 alla tensione di riferimento a polarità invertita (Q1, U12C). U12B confronta l'uscita triangolare con l'escursione di tensione dell'amplificatore operazionale prevista ($\pm 2,5$ V) per attivare l'inversione di polarità (Q1) della ten-

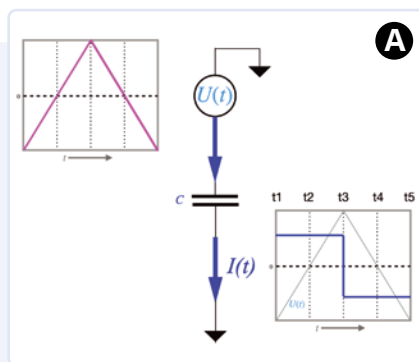


Metodo di Misurazione

La carica Q immagazzinata in un condensatore è proporzionale alla tensione U ; aumentando la tensione $U(t)$ su un condensatore aumenta la carica immagazzinata. Ciò implica che deve fluire una corrente ($I(t)$). Matematicamente, questo è espresso da $I(t) = C(dU/dt)$.

Se U aumenta con una pendenza costante, la corrente è costante. Un cambiamento da una pendenza crescente a una decrescente implica un cambio nella direzione della corrente. Questo è illustrato nell'**Esempio A**. Per un resistore, la corrente è proporzionale a U (**Esempio B**); la corrente attraverso un resistore in parallelo a C si somma semplicemente alla corrente del condensatore. In un qualsiasi istante t_0 con $U(t_0) = 0$, solo la corrente capacitiva prevale. Eseguire campionamenti in questi momenti (e moltiplicare per il segno della pendenza) produce valori di corrente che dipendono esclusivamente da C . Se la corrente $I(t)$ aumenta linearmente attorno a t_0 , l'intervallo di tempo di campionamento può avere un'estensione simmetrica finita prima e dopo t_0 il che aiuta a mediare i contributi da varie fonti di rumore.

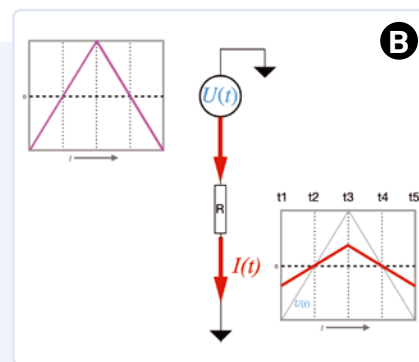
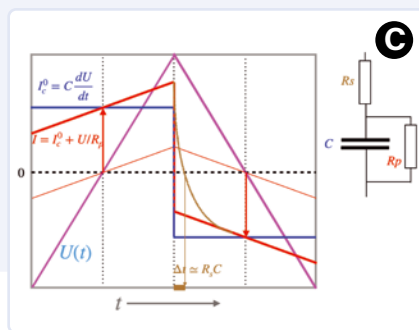
Ora, se abbiamo anche una resistenza in serie R_s , il suo effetto è che la transizione dai valori di corrente positivi a quelli negativi quando cambia la pendenza di $U(t)$ non è più brusca, ma segue un transitorio con una costante di tempo finita. Questa costante di tempo è $\tau_s = RC$, dove R è il valore di $R_s // R_p = R_s R_p / (R_s + R_p)$, che per R_p grande è molto vicino alla resistenza in serie R_s . Dopo alcuni cicli di τ_s , gli effetti



transitori sono terminati. Se l'intervallo centrale di campionamento inizia a questo punto, non ci sarà alcuna influenza residua di R_s sul valore di C inferito. Nel circuito presentato nell'**Esempio C**, il tempo $\Delta t \approx \tau_s$ è determinato dalla conversione del ritardo temporale tra il cambiamento di segno della pendenza e il passaggio per zero di $I(t)$ in una tensione.

Si noti che lo shunt utilizzato per la misurazione di $I(t)$ contribuisce anche alla resistenza in serie efficace R_s . Per compensare questo, la correzione del feedback su $U(t)$ è aggiunta al circuito (vedere **Feedback**).

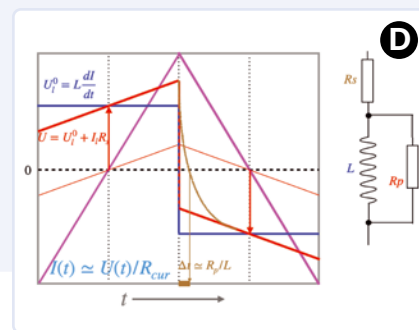
Per applicare lo schema agli induttori L invece che ai condensatori, è necessario scambiare i ruoli di corrente e tensione.



Pilotare l'elemento di prova (L) con una corrente crescente linearmente $I(t)$ produce una tensione indotta costante $U(t)$ sull'induttore $U = L(dI/dt)$.

L'effetto simile del resistore in parallelo nel caso dei condensatori è ora la tensione extra causata dalla resistenza (in serie) dell'induttore, mentre qualsiasi resistenza in parallelo all'induttore corrisponde all'effetto della resistenza in serie del condensatore (vedere **Esempio D**).

Mentre, per i "buoni" condensatori, la resistenza in parallelo è tipicamente così grande che il suo effetto è praticamente invisibile, la resistenza in serie degli induttori è quasi sempre chiaramente visibile come la pendenza della tensione rilevata attraverso l'induttore.



sione di riferimento ad alta precisione che controlla la pendenza determinante il valore. U12D bufferizza la tensione triangolare attenuata ($R3, R8$) per pilotare il componente di test C_x a bassa impedenza. La corrente indotta dal condensatore scorre attraverso il resistore di shunt, selezionato da SW2 tramite i MOSFET di auto-ranging Q2...Q7. La tensione risultante sullo shunt di pochi millivolt è amplificata (fattore totale ≈ 500) da U16A e U16B. Dopo il primo stadio di amplificazione, il segnale di retroazione positivo viene prelevato da R74 e inviato all'attenuatore di

pilotaggio ($R3, R8$) per incrementare la tensione di pilotaggio, includendo anche la caduta di tensione sullo shunt. Il segnale amplificato $I_x(t)$ può essere monitorato tramite oscilloscopio.

Nel circuito, questo segnale subisce inoltre un'inversione di polarità sincrona eseguita dai componenti Q8 e U1B. Per un condensatore ideale, l'uscita di U1B sarebbe quindi una tensione continua proporzionale al valore della sua capacità. Componenti non ideali, come resistenze in parallelo, inducono una pendenza sulla corrente attraverso R_{shunt} e quindi un ripple nel valore dell'uscita di

U1B. Di conseguenza, la tensione ripple a dente di sega deve essere campionata in finestre temporali appropriate e mediata, per isolare i valori corretti e stabili di C e R_p .

Le finestre di tempo di campionamento sono estratte dalla logica (U7, U18) guidata dal segnale proveniente dai comparatori (U2, U3, U13; nel prototipo ho usato operazionali audio MC33172 di STMicroelectronics) che rilevano le transizioni della tensione di pilotaggio (non attenuata) ai livelli forniti dal partitore di tensione tra R24 e R43. Il segnale logico ottenuto dall'operazione XOR delle transizioni successive è alto quando la tensione di pilotaggio è all'interno della finestra tra i livelli corrispondenti. Pertanto, una finestra centrale, simmetrica rispetto all'incrocio dello zero della tensione di pilotaggio, viene fornita per il campionamento principale che restituisce il valore di C e viene applicata all'interruttore MOSFET U5B, che campiona l'uscita dell'amplificatore attorno alla parte centrale della pendenza e la media in C2. Vengono selezionate finestre di campionamento decentrate in modo che solo le "finestre tardive" delle pendenze ascendenti o discendenti siano attivate tramite un'operazione AND con Sync1 o NOT Sync1. Applicando queste finestre rispettivamente a U5C e U5D, il segnale di tensione dell'amplificatore nelle finestre decentrate (sia in salita che in discesa) viene mediato da C3.

La media del campionamento del segnale principale e quella dei due periodi decentrati hanno costanti di tempo determinate da $R30 \times C2$ e $R39 \times C3$; tuttavia, questi tempi contengono un fattore aggiuntivo $1/f$ — circa 4 per il campionamento principale e 8 per quello decentrato — con f che rappresenta la frazione del periodo inclinato durante il quale la finestra di campionamento è aperta. Le larghezze relative delle finestre di campionamento (f) sono controllate dai livelli del partitore di tensione rilevati dalla fila di comparatori (U2, U3, U13). Infine, per ottenere una stima approssimativa della resistenza serie, il tempo di attraversamento dello zero (= metà del salto di tensione al rovesciamento della pendenza se $R_p = \infty$) del segnale viene rilevato da U1A (come comparatore) e "XORato" (U9D) con il segnale Sync1, che idealmente è alto durante il ritardo tra l'inversione di segno ("zero") e il momento in cui il salto di tensione (idealmente improvviso) della tensione di shunt amplificata è pari a $1/2$ della sua ampiezza. Questo ritardo temporale dipende dalla costante di tempo $\tau_s = R_{series}C$. Per convertirlo in una tensione, si apre l'interruttore U5A verso la tensione di

Feedback

In modalità capacitiva, la tensione della resistenza shunt viene aggiunta alla tensione di pilotaggio $U(t)$ in modo che la tensione netta applicata al condensatore in esame corrisponda alla forma e al valore triangolare non distorto. La retroazione utilizza la tensione shunt amplificata e tamponata ($\times 20$) e la inietta, tramite R74, nel divisore di tensione attenuante R3/R8. L'uscita del partitore di tensione è tamponata da U12D per fornire un'impedenza di sorgente molto bassa per la tensione di pilotaggio. La resistenza di retroazione R74 deve essere impostata su un valore tale che l'attenuazione di R74/R8 corrisponda all'amplificazione per ottenere un fattore di retroazione ≈ 1 . Per fattori maggiori, l'anello diventa instabile. In pratica, R74 deve essere regolato su un fattore prossimo a 1 utilizzando come condensatore di prova un condensatore di alta qualità nell'intervallo 100...1.000 nF. La retroazione serve a eliminare l'influenza della resistenza shunt sull'effetto della resistenza in serie. Non ha alcuna influenza sull'accuratezza del valore C dedotto dal campionamento centrale, purché la durata del transitorio dovuto a R_s sia sufficientemente inferiore al periodo dell'azionamento triangolare.

Per le misure di induttanza non viene implementata alcuna retroazione, poiché l'intervallo imposto dall'amplificatore limita la tensione utile sull'induttore a < 10 mV all'estremità bassa delle resistenze di regolazione della corrente R50, R55... che si confronta con i 2,5 V all'estremità "calda"; l'errore di corrente risultante è trascurabile (se si punta a una precisione dell'1% circa).

riferimento per la durata di quel periodo, mediando poi il segnale con i componenti R57, C4 e R62.

Da questa operazione è possibile stimare il ritardo temporale e, di conseguenza, la resistenza in serie.

MODALITÀ INDUTTIVITÀ

Il selettore SW3 consente di passare dalla modalità capacità a quella induttanza. In modalità induttanza, l'induttore in esame sostituisce il resistore di shunt usato in modalità capacità.

La corrente di pilotaggio necessaria è fornita tramite le resistenze R60, R55 e R46, con valori compresi tra 1 k Ω e 1 M Ω , e alimentate dalla tensione di pilotaggio triangolare non attenuata.

Poiché la caduta di tensione sull'induttore è limitata a 10 mV, molto inferiore rispetto ai 2,5 V di pilotaggio, la sorgente di corrente può essere considerata sufficientemente accurata senza necessità di ulteriori correzioni.

INTERVALLI E CODIFICA DEGLI INTERRUTTORI

Una volta impostata la pendenza, manualmente o tramite l'algoritmo di auto-range nel programma, viene selezionato un resistore di shunt (o resistore di alimentazione) in modo che la tensione risultante sia idealmente compresa tra 400 e 4000 mV per

garantire un'accuratezza ottimale. Le tensioni sono mostrate anche sul display LCD.

Per l'auto-range, la tensione grezza in uscita dall'unità di commutazione sincrona U1B viene inviata direttamente all'ingresso analogico A2 dell'Arduino per evitare un ritardo dovuto alla mediazione.

I resistori di shunt o i resistori di alimentazione sono selezionati manualmente da SW1, dove, con l'opzione auto-range, l'ultima posizione include, rispettivamente, i MOSFET (Q2...Q7 per la misurazione di Cx o i resistori commutati da U10 per Lx).

CONSIDERAZIONI SULL'ACCURATEZZA

Le incertezze sui valori del condensatore di integrazione C1, dei resistori di integrazione R9, R11, R12, R14, R15 e R16, così come quelle dei resistori di shunt o di alimentazione, influiscono sull'errore dei valori finali visualizzati di C o L. Inoltre, i resistori R5, R10 e R13, così come R66, R72 e R73 nelle fasi di inversione di segno, devono avere esattamente lo stesso valore. Anche l'attenuatore R3, R8 e il fattore di amplificazione determinato da R67, R68, R69 e R70 influiscono sul risultato.

Lo sketch Arduino consente di inserire i valori esatti (misurati) di C1, dei resistori di shunt e di alimentazione, nonché i fattori di attenuazione e amplificazione. Per il prototipo presentato qui, ho misurato questi resistori singolarmente utilizzando un multimetro digitale a 6 cifre e ho inserito i valori nello sketch. Per quanto riguarda i resistori, un'alternativa praticabile sarebbe utilizzare tipi con una tolleranza dello 0,1%. Per C1, invece, la misurazione rimane l'opzione più accessibile.

In ogni caso, se i valori differiscono dai componenti previsti, basta inserire il valore reale nello sketch! Per mantenere l'accuratezza della misurazione di C in presenza di una resistenza parallela relativamente bassa, o di L in presenza di una resistenza serie elevata, può essere utile regolare la simmetria della finestra di campionamento centrale rispetto al passaggio per lo zero di U(t) e, se necessario, anche la durata dell'intervallo di campionamento centrale.

Questa regolazione può essere eseguita tramite i resistori del partitore di tensione R33 e R38.

OSSERVAZIONI AGGIUNTIVE

La scelta di utilizzare il MOSFET BS170 per la commutazione del resistore di shunt è stata una prova di principio per il prototipo, realizzata con i transistor disponibili al momento. La resistenza ON misurata era di circa 1,7 Ω , valore che è stato

aggiunto ai valori di shunt nello sketch. Questo risulta un po' problematico per i due valori di shunt più bassi (nominalmente 1 Ω e 10 Ω): per il test, il resistore da 1 Ω è stato sostituito unicamente dalla resistenza del MOSFET.

Un miglioramento sarebbe quello di commutare almeno questi due valori di shunt più bassi utilizzando, ad esempio, un Si4136DY (0,0025 Ω a 4,5 V). Il diodo dei MOSFET di potenza a bassa resistenza non crea problemi per questa applicazione, poiché la tensione massima applicata resta inferiore a 0,2 V. La commutazione dei resistori di alimentazione della corrente per l'induttanza espone gli interruttori a tensioni maggiori, ma con un valore minimo previsto di 1 k Ω , la commutazione è compatibile con la resistenza ON di 10...20 Ω della serie DG411...13. Per gli interruttori CD4016 di campionamento e media, nonostante la loro resistenza ON di alcune centinaia di ohm, risultano perfettamente adatti e i loro livelli di tensione sono compatibili con gli operazionali.

Avvertimento: i resistori R31, R40, R58 e R64, che collegano l'uscita degli amplificatori buffer di tensione mediata all'ADC a 16 bit, così come R1 al riferimento digitale dell'interruttore DG411...13, non devono essere omessi.

Proteggono i chip dall'eccesso di corrente in ingresso dovuto alle differenze di intervallo di tensione tra la linea a 5 V dell'Arduino e l'alimentazione ± 9 V degli operazionali.

CONNESSIONE AI COMPONENTI IN TEST

Poiché la capacità viene misurata tramite la corrente che scorre attraverso il condensatore (C), è possibile collegare il condensatore da testare con due cavi schermati: uno per la tensione di pilotaggio e uno per l'ingresso dell'amplificatore/shunt. La capacità parassita residua è quindi limitata alle estremità non schermate dei cavi (una lunghezza di 0,5 m è perfettamente adeguata). Eventuali capacità parassite interne vengono sottratte tramite il programma Arduino.

I piccoli induttori da testare devono essere collegati con fili intrecciati molto corti. La misura di induttori più grandi può trarre beneficio dall'uso di cavi schermati (che possono essere più lunghi) per ridurre l'induzione di rumore (**Fig. 3**).

ESEMPI

Come esempio del funzionamento di base del circuito come misuratore di capacità, il display con un condensatore Styroflex da 470 nF è mostrato nelle **Fig. 4a** e **Fig. 4b**. Le due immagini mostra-

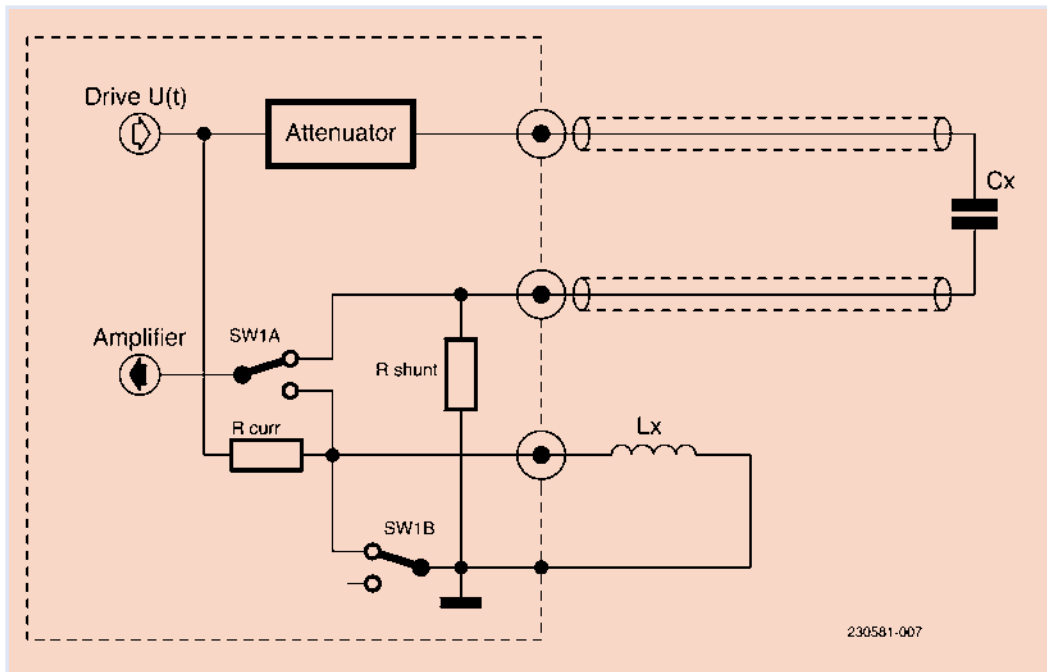


Fig. 3
Schema del front-end di uscita del misuratore LC, con cavi di sonda schermati da 0,5 m.

no i risultati ottenuti con diverse pendenze della tensione di pilotaggio $U(t)$. La pendenza è indicata dal valore della resistenza integratrice R_C nella riga finale del display, dove viene mostrato anche il valore effettivo della resistenza shunt $R = R_{shunt}$ determinato in modalità automatica. Il risultato principale C è visualizzato insieme alla tensione grezza su A0 dell'ADS1115. La seconda riga mostra il valore stimato, R_p , della resistenza parallela e la tensione relativa su A1 dell'ADS1115, dove è troppo elevata per essere rilevata e viene indicata dal display con "N.A." Infine, la terza riga fornisce una stima della resistenza in serie con la tensione su A3 dell'ADS1115. L'intervallo di tensione è 0...4,096 mV; maggiore è il valore all'interno dell'intervallo, migliore sarà la risoluzione/accuratezza. R_p è derivato dalla differenza tra le tensioni A0 e A1 ottenute per un intervallo di

campionamento decentrato e uno centrale. A3 è una conversione (non lineare) del ritardo temporale tra il cambio di pendenza e il passaggio per lo zero del segnale rispetto al periodo della tensione di pilotaggio. La tensione campionata grezza è mostrata in **Fig. 5**. La traccia rossa rappresenta $U(t)$ e la traccia blu la tensione al terminale $ICx(t)$. A parte i picchi, essa si conforma alla risposta ideale di un condensatore. I picchi visibili probabilmente derivano dalla sovrapposizione degli impulsi di campionamento con il cablaggio non ideale (caotico) del prototipo (**Fig. 6**). Qui, essi illustrano bene la posizione degli intervalli di campionamento. L'ampia lacuna al centro (intorno ai 25 ms) rappresenta la finestra principale di campionamento, i segni intorno ai 38 ms indicano la finestra di campionamento in cui viene rilevato l'aumento della tensione dovuto alla resistenza parallela.

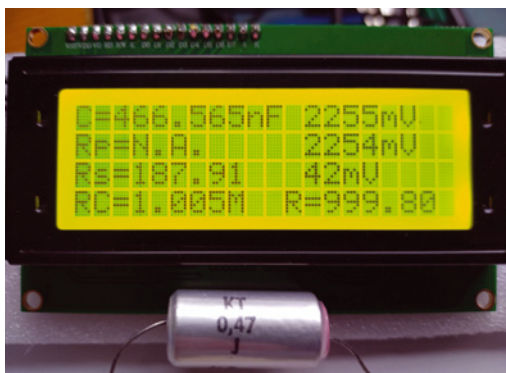


Fig. 4
Letture ottenute con diverse pendenze della tensione di alimentazione $U(t)$, con i valori del resistore di shunt R determinati automaticamente. Con stima di un resistore in serie $R_s = 187 \Omega$ (a sinistra), e senza di esso (a destra).

Fig. 5
La tensione grezza campionata.



Il secondo esempio simula un grande condensatore elettrolitico in-circuito da 1.000 μF in parallelo a due diodi in entrambe le direzioni e una resi-

Fig. 6
Il progetto in una versione assolutamente prototipale!

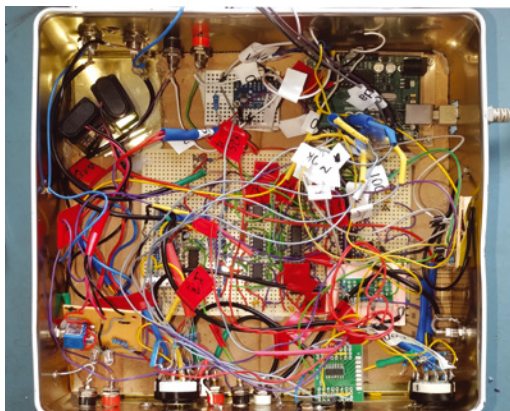
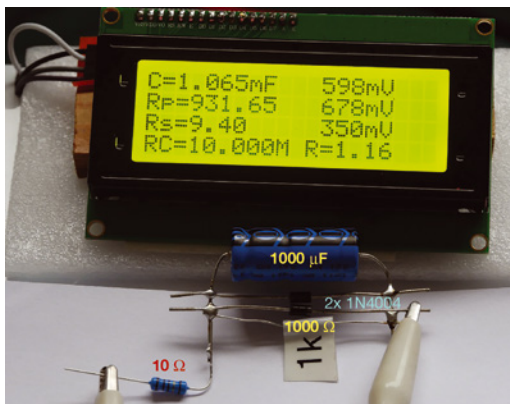


Fig. 7
Lettura con un condensatore elettrolitico da 1.000 μF , in parallelo a due diodi e una resistenza da 1 k Ω , oltre a una resistenza in serie da 10 Ω .



stenza da 1 k Ω con una resistenza in serie da 10 Ω (**Fig. 7**). La grande capacità richiede la scelta di una pendenza con un periodo di circa 1 s ($RC = 10 \text{ M}\Omega$). Come visibile in **Fig. 8**, la traccia della tensione grezza (blu) mostra ora il salto transitorio di tempo finito al cambio di pendenza, nonché la tensione inclinata $ICx(t)$ ai tempi successivi, che indicano la presenza della resistenza parallela. La tensione applicata è troppo bassa perché i diodi conducano una corrente significativa, quindi restano "invisibili". Questa figura mostra anche che i valori dedotti per C e R_p possono essere influenzati negativamente in situazioni estreme se il transitorio dovuto a R_s prevale nei momenti in cui inizia la finestra centrale di campionamento.

INDUTTORI

Una tipica traccia di un induttore è mostrata in **Fig. 9**, ottenuta con un nucleo ad anello con 29 spire, $L = 3,45 \text{ mH}$ ($Al \approx 4000 \text{ nH/n}^2$), mentre le **Fig. 10a** e **10b** mostrano le misurazioni ottenute usando questo design e un misuratore commerciale di induttanza, rispettivamente. Come accade per gli induttori tipici, le tracce mostrano una pendenza considerevole dovuta alla resistenza della bobina. Il rapporto tra la parte costante dipendente da L (ampiezza del salto), che dipende dalla velocità di variazione della corrente dI/dt , e la pendenza dipendente dalla resistenza (indipendente da dI/dt) può essere ottimizzato scegliendo una velocità adeguata.



Fig. 8
La traccia della tensione grezza (in blu) mostra il transitorio di salto a tempo finito nei punti di inversione della pendenza.

Esistono un paio di limiti: un'ampiezza di salto troppo ridotta a velocità basse da un lato, e oscillazioni spurie di lunga durata dopo il salto (dovute a capacità parassite) a velocità troppo elevate (cioè con un intervallo troppo breve tra il salto e la finestra di campionamento).
Una raccolta di esempi applicativi per diversi condensatori e induttori può essere trovata su <https://youtu.be/hfbUxPfHmeg>.

STRATEGIA DI MISURA E MONITORAGGIO

Le informazioni più complete e rapide sull'elemento in test, così come la rilevanza dei risultati, si ottengono monitorando il segnale ICx (t) con un oscilloscopio. In questo modo, qualsiasi violazione delle condizioni di validità diventa immediatamente evidente, facilitando anche la scelta del valore ottimale di pendenza. Inoltre, con o senza monitoraggio del segnale, la coerenza delle letture



Fig. 9
Traccia tipica di un induttore.

Elenco componenti

Resistenze (tutti 1%, film metallico, $\geq 1/8W$)

R1, R71 = 4,7 k Ω
 R2, R4, R34, R35, R36, R37 = 2,2 k Ω
 R3 = 1,0 k Ω (1)
 R5, R6, R7, R10, R13, R66, R72, R73 = 100,0 k Ω (2)
 R8 = 100,0 Ω (1)
 R9 = 100,0 M Ω (3)
 R11 = 10,0 M Ω (3)
 R12, R27 = 1,0 M Ω (3)
 R14, R26, R46, R54, R62 = 100,0 k Ω (3)
 R15, R28, R53, R55, R57 = 10,0 k Ω (3)
 R16, R25, R52, R60 = 1,0 k Ω (3)
 R17, R30, R39, R75 = 100,0 k Ω
 R18, R19, R20, R21, R22, R31, R40, R48, R58, R59, R63, R64 = 10,0 k Ω
 R23, R45, R65 = 1,0 k Ω
 R24, R29, R32, R33, R38, R41, R42, R43 = 1,0 k Ω (2)
 R44 = 1,0 M Ω
 R47, R51 = 100,0 Ω (3)
 R49 = 0,0 Ω (3)
 R50, R56 10,0 Ω (3)
 R61 = 1,0 Ω (3)
 R67, R69 = 4,7 k Ω (1)
 R68, R70 = 100,0 k Ω (1)
 R74 = 5 k Ω trimmer

Condensatori

C1 = 22 nF, a basso coefficiente di temperatura
 C2...C14 = 1 μ F, ceramica, multistrato

Semiconduttori

D1,D2 = 1N4007, diodi di protezione
 Q1, Q8 = J113, interruttore FET n-j

Q2, Q3, Q4...Q7 = BS170, interruttore MOSFET

Q2,Q3 = Si4136DY, MOSFET (4)

U1, U12 = TL084, operazionale quadruplo

U2, U3, U13 = MC34072PG, amplificatore operazionale doppio

U2, U3, U13 = LM393B+1k collettore R, doppio comparatore (5)

U16 = OPA2182, operazionale veloce a zero offset

U4, U11 = TL431D, riferimento a 2,5 V

U10 = DG413Y, interruttore quadruplo MOSFET (azionamento di corrente)

U5 = CD4016B, interruttore quadruplo a MOSFET (campionamento)

U7 = CD4001B, quadruplo NOR a 2 ingressi

U9 = CD4070B, quadruplo XOR a 2 ingressi

U18 = CD4093B, quadruplo NAND a 2 ingressi

Varie

SW1 = interruttore rotante, 6 $\times$ 2

SW2 = interruttore rotante, 4 $\times$ 3

SW3 = interruttore a levetta, 2 $\times$ 2

Note:

(1): Le tolleranze di questi componenti possono essere compensate attraverso i valori di valori di amplificazione e attenuazione inseriti nel programma Arduino.

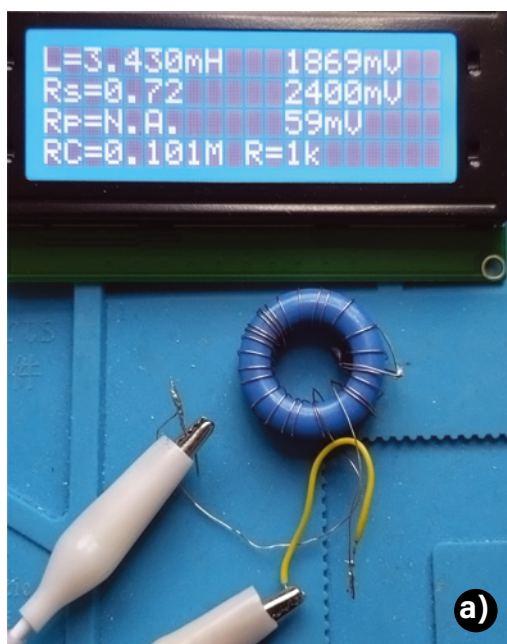
(2): L'uguaglianza dei valori di queste resistenze influisce sulla precisione.

(3): Queste resistenze sono fondamentali per la precisione. Utilizzare resistenze di precisione (a bassa tolleranza).

(4): Alternativa piú precisa al BS170, con un Ron inferiore.

(5): Alternativa a MC33172.

Fig. 10
 Misure ottenute con il nostro progetto (a) e utilizzando un misuratore di induttanza commerciale (b).



Sketch di Arduino

La comunicazione con il microcontrollore comprende l'uso di uno shield ADC a 4 canali e 16 bit basato su ADS1115 e l'uscita su un LCD 4x20, entrambi collegati via I2C. Altre tensioni ausiliarie vengono lette direttamente tramite gli ingressi analogici diretti A0...A2 di Arduino, utilizzati per decodificare le posizioni degli interruttori di gamma e supportare la funzione di (semi) auto-ranging. Altri GPIO digitali sono utilizzati per azionare gli interruttori MOSFET che selezionano le resistenze di shunt o di alimentazione della corrente. I valori effettivi delle resistenze corrispondenti devono essere aggiornati nel codice sorgente per ottenere una precisione ottimale. La funzione principale del micro è interpretare le tensioni ADC misurate e convertirle nelle informazioni desiderate su C/L, R_p e R_s.

$$C_x = \frac{U_{adc0}}{U_{ref}} \cdot \frac{R_c \cdot C_1}{R_{shunt} \cdot A} - C_{offset}$$

e analogamente

$$L_x = \frac{U_{adc0} \cdot R_c \cdot C_1}{\alpha \cdot U_{ref}} \cdot R_{current-feed}$$

con A che è il fattore che combina l'attenuazione applicata alla tensione di pilotaggio $\approx 1/10$ con l'amplificazione $\alpha \approx 500$, cioè $A \approx 50$. La resistenza in parallelo (resistenza in serie per gli induttori) è la differenza di tensioni dalla finestra di campionamento "tardiva" decentrata U_{adc1} e dalla finestra centrale U_{adc0}; U_{ref} è la differenza di tensione di pilotaggio tra i centri delle due finestre di campionamento

$$R_p = \frac{U_{ref} \cdot f_p \cdot R_{shunt} \cdot A}{U_{adc1} - U_{adc0}}$$

e per gli induttori

$$R_s = \frac{U_{adc1} - U_{adc0}}{\alpha \cdot U_{ref} \cdot f_p} \cdot R_{current-feed}$$

L'analisi del transitorio che contiene informazioni sulla resistenza in serie di C o sulla resistenza in parallelo di L (o altre perdite) è più complessa. Nella prima fase, il ritardo tra il cambiamento di pendenza e l'attraversamento dello zero del segnale τ_{ESR} deve essere dedotto dalla lettura della tensione nel terzo canale ADC U_{adc2}:

$$\tau_{ESR} = 2 \cdot R_c \cdot C_1 \cdot \frac{R_{44} \cdot U_{adc2}}{R_{45} \cdot (U_{ref} - U_{adc2})} - \tau_{offset}$$

dove τ_{offset} (pochi μs) è un offset tecnico che corregge i ritardi nei rilevamenti dello zero.

Nel caso in cui la resistenza parallela di C possa essere trascurata (cioè nessuna pendenza nel segnale di corrente), la resistenza in serie può essere facilmente ottenuta da $R_s = \tau_{ESR} / (\ln(2) \cdot C_x)$. Nel caso di un effetto sostanziale della resistenza parallela, è necessario tenere conto della pendenza extra e l'espressione diventa leggermente più complessa.

Se la pendenza legata a R_p domina l'attraversamento dello zero, non è possibile ottenere una stima affidabile di R_s e il display mostrerà N.A.

Le espressioni effettive possono essere visualizzate nel sorgente del programma Arduino.

al variare della pendenza di pilotaggio (impostata tramite R_c) indica valori affidabili, mentre l'ispezione delle tensioni ADC grezze consente di valutare l'accuratezza delle misurazioni. In ogni caso, la resistenza serie equivalente (ESR) deve essere considerata solo come una stima approssimativa.

POSSIBILI MODIFICHE

- Per estendere l'intervallo della resistenza parallela ammessa, si può ridurre la finestra centrale di campionamento (riducendo R33 e R38).
- Questo restringimento della finestra di campionamento potrebbe richiedere una riduzione del valore di R30 per mantenere costante il prodotto $f \times R_{30}$.
- Le tensioni C/L e R_p possono anche essere misurate utilizzando gli ingressi analogici A2 e A3 di

Arduino, se si accetta una minore accuratezza.

- La tensione ± 250 mV di U(t) può essere ulteriormente ridotta per evitare completamente errori in R_p in presenza di diodi Schottky riducendo R8. Il fattore di attenuazione reale deve essere modificato nel programma Arduino.

La parte analogica può anche essere progettata in modo tale da richiedere solo l'alimentazione a 5 V (di Arduino). Questo implica l'uso di opamp adeguati (ad esempio OPAx388 per tutti), e la divisione della tensione per creare una massa virtuale (ad esempio usando un TLE2426). Tuttavia, per il commutatore di gamma automatica della resistenza di corrente dell'induttore, deve essere trovato un sostituto adeguato per DG411...13.



Per maggiori informazioni visita la pagina elektormagazine.com/magazine/elektor-341/62884

Fai il pieno di energia!

BATTERIA RICARICABILE ULTRACELL AL PIOMBO SENZA MANUTENZIONE

Batterie al piombo senza manutenzione, ad alta densità d'energia e a bassissima autoscarica, prodotte dalla ULTRACELL. Per uso generico, la loro robustezza e affidabilità le consentono un ampio campo di applicazioni: solare, antifurto, backup, UPS, illuminazione d'emergenza ecc. La modalità di ricarica consigliata è quella a corrente costante.



BATTERIE RICARICABILI AL NI-MH VARTA "READY TO USE"



€ 14,50
Cod. HR03AAA800



€ 15,80
Cod. HR6AA2100



€ 13,80
Cod. HR14C3000

Prezzi IVA inclusa.

**FUTURA
ELETTRONICA®**

www.futurashop.it

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche
di questi prodotti e acquisti on-line
su www.futurashop.it

KiCad

CONOSCERE E USARE

di CRISTIANO RUGGERI



Impariamo ad utilizzare uno strumento di progettazione elettronica tanto potente quanto versatile, che consente di partire dall'idea ed arrivare alla scheda finita. Prima Puntata.

La progettazione dei circuiti stampati per applicazioni elettroniche è una pratica che esiste da oltre sessant'anni e che si è evoluta nel tempo. Partendo da processi rudimentali e relativamente imprecisi, è arrivata a cicli produttivi completamente automatizzati, con tolleranze di realizzazione incredibilmente basse.

In origine, i circuiti stampati erano poco più che prototipi di architettura semplice e venivano disegnati direttamente sulle piastre di rame, partendo da uno schema spesso tracciato a mano. Nei casi più professionali, si utilizzava un tecnigrafo. La piastra veniva poi immersa in una soluzione che eliminava il rame non protetto dal disegno, lasciando solo le piste di collegamento in rame.

Con il miglioramento delle tecniche fotografiche e l'aumento della complessità dei circuiti, si è passati alla fotoincisione: un processo in cui il disegno del circuito stampato veniva trasferito sulla scheda di rame per effetto fotografico. In pratica, il circuito era disegnato su una lastra trasparente, interposta

tra la superficie in rame della piastra (verniciata con un'apposita vernice sensibilizzata) e una sorgente di raggi ultravioletti. I raggi UV "dissolvevano" la vernice non protetta dal disegno, permettendo poi di sviluppare la lastra rimuovendo il rame non protetto e lasciando solo i percorsi elettrici.

Negli ultimi 20-30 anni, sia per l'aumentata complessità dei circuiti stampati e dei sistemi elettronici, sia per il naturale corso evolutivo delle tecnologie, anche gli strumenti di progettazione al computer hanno subito una straordinaria evoluzione. Oggi è possibile passare dall'idea iniziale alla visualizzazione in 3D di un progetto ancora prima di realizzare il primo prototipo.

In generale, questi sistemi di progettazione vengono chiamati CAD (Computer Advanced Design), mentre nello specifico, in ambito elettronico, vengono chiamati ECAD (Electronic CAD) o EDA (Electronic Design Automation).

Per anni, le piattaforme più performanti e diffuse in ambito professionale sono state Mentor Graphics, Orcad, Altium, EasyPC, tutte molto potenti ma piut-

tosto costose, quindi accessibili principalmente alle aziende e non agli hobbisti.

Oggi si nota una convergenza verso piattaforme di sviluppo potenti, complete, intuitive, versatili e soprattutto open source, spesso gratuite o quasi gratuite, come Eagle e KiCad (ma anche altre). Questi strumenti permettono a chiunque operi nel settore dell'elettronica di realizzare circuiti stampati in modo preciso, rapido, efficiente e a costi contenuti. Abbiamo già trattato di Eagle in un corso specifico e di KiCad circa dieci anni fa. Nel frattempo, KiCad ha fatto notevoli progressi e oggi iniziamo a parlare di KiCad 8, un ambiente di sviluppo integrato (IDE) per la progettazione completa di "progetti" di circuiti stampati.

INTRODUZIONE A KICAD

Tratto e tradotto direttamente dal sito www.kicad.org: "KiCad è una suite software open source per l'Electronic Design Automation (EDA). I programmi gestiscono l'acquisizione schematica e il layout PCB con output Gerber e IPC-2581. La suite funziona su Windows, Linux e macOS ed è concessa in licenza sotto GNU GPL v3.

L'obiettivo del progetto KiCad è quello di fornire la migliore applicazione possibile per la progettazione elettronica multipiattaforma per i progettisti di elettronica professionisti. Viene fatto ogni sforzo per nascondere la complessità delle funzionalità di progettazione avanzate in modo che KiCad possa rimanere fruibile anche da utenti nuovi e inesperti, pur dando però priorità alle esigenze degli utenti professionali quando si determina la direzione del progetto e la priorità delle nuove funzionalità

Il progetto KiCad è gestito da un comitato tecnico, composto dai membri del team di sviluppo principale. La maggior parte delle decisioni viene presa in accordo con il comitato tecnico, ma in caso di disaccordo, la decisione finale spetta al responsabile del progetto.

KiCad è stato pubblicato per la prima volta nel 1992 dal suo autore originale, Jean-Pierre Charras. Da allora è stato in continuo sviluppo ed è ora gestito dal team di sviluppo di KiCad. Il nome di KiCad deriva dalle prime lettere di una società dell'amico di Jean-Pierre Charras "Ki" che viene combinato con "Cad". Ma ora non ha altro significato se non quello di essere il nome della suite software..."

Ossia...

Come Eagle, EasyPC, EasyEDA, anche KiCAD è una "suite" di applicazioni specifica per il progetto elettronico che, però non si limita alla sola progettazione di uno schema e del relativo circuito stampato: essa integra altre funzioni molto utili quali

- la possibilità di convertire immagini bitmap in simboli elettrici;
- un calcolatore specifico per i circuiti stampati, in grado di calcolare combinazioni di resistenze e condensatori per raggiungere il valore più vicino possibile a quello desiderato,

impedenze, correnti, bande di isolamento, formule specifiche per la radiofrequenza e altro ancora;

- tantissimi plugin di produzione sia KiCad che di terze parti in grado di eseguire le più disparate operazioni come l'arrotondamento degli spigoli delle piste, la generazione dell'elenco componenti virtuale, l'auto sbrogliatura e l'auto posizionamento dei componenti in modo ottimizzato;
- un efficiente sistema di archiviazione dei progetti completi e dei backup automatici di ogni progetto.

Il tutto attraverso un'interfaccia grafica intuitiva ed una struttura di lavoro ben ottimizzata poiché, pur essendo molto potente e particolarmente adatto per impieghi di alto livello professionale, è in realtà stato pensato per essere "user friendly".

Potremmo spendere tante parole per cercare di convincervi che KiCad è la panacea di ogni progettazione elettronica, ma siamo convinti che iniziare subito a vedere il programma possa essere molto più efficace, soprattutto in considerazione del fatto che non è nostra intenzione mettere KiCad in concorrenza con altri sistemi di sviluppo simili; quello cui noi miriamo, come sempre, è fare didattica e divulgazione al meglio delle nostre capacità. Se poi dopo averlo provato insieme a noi, questo tool vi sarà piaciuto al punto da voler continuare ad usarlo, beh, in questo caso vi auguriamo buon lavoro. Come ultima nota introduttiva, sappiate che quando è stato scritto questo corso, la versione più recente di KiCad era la V8.0.5 e che pertanto tutto il corso fa riferimento ad essa: ciò significa che eventuali versioni successive potrebbero presentare variazioni rispetto a quanto descritto qui.

DOWNLOAD E INSTALLAZIONE

Per prima cosa occorre recuperare il pacchetto completo per l'installazione, cosa questa che si fa accedendo al sito www.kicad.org e cliccando il tasto blu "Download" in centro schermo (**Fig. 1**). Volendo potete cliccare anche i tasti a sinistra e a destra di esso per scaricare documentazione e novità. Una volta cliccato il pulsante download compare la pagina relativa alla scelta del sistema operativo: ovviamente ognuno di voi deciderà in base al proprio OS; noi abbiamo optato per Windows cliccando sulla icona relativa (**Fig. 2**). A questo punto compare la finestra di scelta del server da cui scaricare. Si entra così nell'ultima pagina del sito, quella in cui inizia il download vero e proprio e l'entità KiCad vi chiede se volete effettuare una donazione per far sì che KiCad possa rimanere open source e freeware. Non è un obbligo bensì una scelta: se volete, potete fare una donazione, ma potreste persino diventare collaboratori nello sviluppo.

Per quanto riguarda il download, il pacchetto completo è piuttosto voluminoso: attualmente occupa circa 1,2 GB compresi. Assicuratevi quindi di avere abbastanza spazio libero sul vostro HDD, poiché, una volta installato, occuperà circa 6 GB. Passiamo ora all'installazione: dalla cartella di download, o da quella in cui avete scelto di salvare la suite, eseguite il

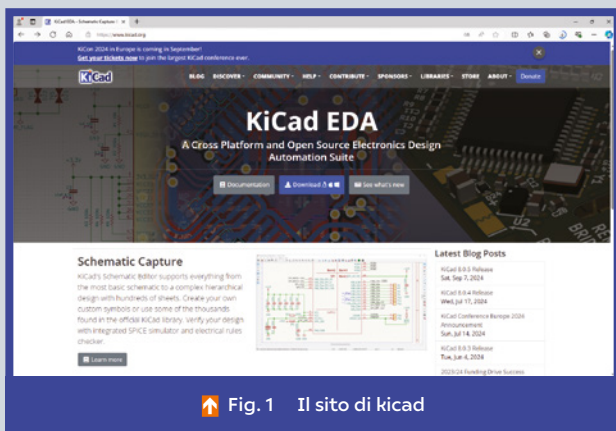


Fig. 1 Il sito di kicad



Fig. 2 La scelta del sistema operativo

pacchetto *kicad-8.0.5-x86_64.exe*. Fate attenzione, poiché l'antivirus potrebbe bloccare l'installazione o eseguirla in una zona sicura e protetta dell'HDD. Le procedure da adottare in questi casi variano in base all'antivirus utilizzato. In alcuni casi, invece, l'installazione potrebbe avviarsi senza alcun ostacolo. Presumiamo che sappiate come gestire eventuali richieste

del vostro antivirus.

In ogni caso, l'installazione vi mostrerà una serie di schermate piuttosto tipiche di qualsiasi installazione, dove è necessario confermare cliccando sul pulsante "Avanti", fino all'ultima schermata, in cui invece si clicca su installa.

Il gestore dell'installazione vi chiederà di scegliere se installare per l'utente corrente o per tutti gli utenti, di selezionare i componenti da installare ed eventualmente di variare la cartella di destinazione, come praticamente avviene in tutte le installazioni Windows.

Di tutta questa sequenza noi riportiamo solo la **Fig. 3**, che mostra la possibilità di selezionare i componenti da installare, e la **Fig. 4** che mostra la cartella di installazione predefinita e lo spazio richiesto.

Arrivati alla schermata dell'ultimo passaggio, in cui viene segnalato il buon esito, è previsto l'avvio automatico della suite, ma potete evitarlo semplicemente togliendo la spunta dall'opzione relativa ben visibile. Sappiate che è probabile che Windows vi chieda l'autorizzazione affinché l'installazione apporti modifiche nella configurazione di macchina (*regedit.exe*). Si deve accettare obbligatoriamente altrimenti l'installazione non procede. E' anche probabile che venga installata la suite Microsoft Visual C++, necessaria al funzionamento in ambito Windows. Qui non vi è opzione di scelta: il sistema la installa in automatico.

L'INTERFACCIA GRAFICA PRINCIPALE

Al primo avvio dell'interfaccia grafica, se sapete dove si trova l'icona di avvio, di solito creata sul desktop del PC, potete avviare KiCad con un doppio clic su di essa o utilizzare un qualsiasi altro metodo di avvio di Windows.

Se invece non sapete dove si trova l'icona, potete eseguire una ricerca in Windows digitando "kicad" nella casella di ricerca: il sistema operativo mostrerà tutte le occorrenze di KiCad, con la migliore corrispondenza in alto a sinistra nella finestra di ricerca. Indipendentemente dal metodo di avvio scelto, la



Fig. 3 Selezione dei componenti da installare

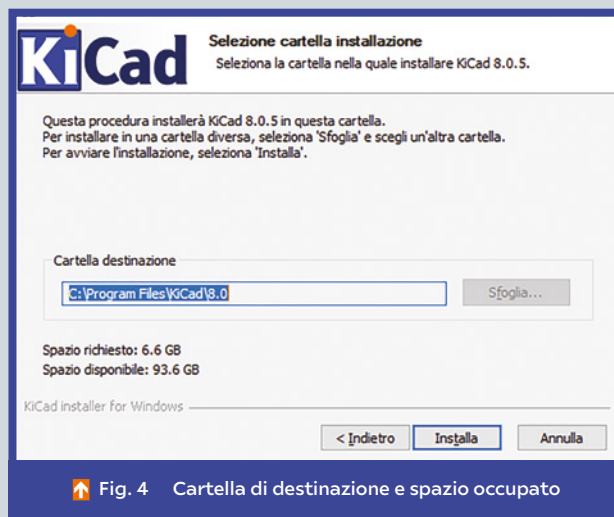


Fig. 4 Cartella di destinazione e spazio occupato

schermata principale sarà quella mostrata in **Fig. 5**.

Sono chiaramente distinguibili 4 sezioni:

- un menu a tendina, in alto a sinistra;
- un menu a icone, verticale, sul lato sinistro della pagina;
- un'area bianca destinata ai file di progetto (in questo momento vuota);
- una sezione molto ampia che ospita tutte le applicazioni che fanno parte dell'ambiente di sviluppo.

IL MENU A TENDINA

Presenta sei opzioni di scelta e precisamente File, Edit, View, Tools, Preferences, Help, che tra poco vedremo nel dettaglio; tenete presente che, in linea di massima, ogni opzione svolge le stesse operazioni svolte in ambito Windows con la stessa dicitura e questo vi dovrebbe in qualche modo essere di aiuto durante il corso. Sappiate inoltre che alcune icone legate ai menu a tendina sono presenti anche sul lato sinistro in alto dell'interfaccia grafica principale (**Fig. 5**), pertanto "stessa icona" equivale a "stessa funzione" quindi una volta vista la funzione di quell'icona nei menu a tendina, essa farà esattamente la stessa cosa quando invocata non da menu ma direttamente da interfaccia grafica in alto a sinistra.

MENU FILE

Visibile in **Fig. 6**, comprende a sua volta undici opzioni di scelta che ora vediamo.

New Project

Crea un nuovo progetto vuoto, assegnandogli una cartella di lavoro e una di backup. Tutti i file creati o aggiunti durante la fase di progettazione dovranno essere contenuti in questa cartella.

Cliccando su New Project compare la classica finestra di navigazione file di Windows (Explorer), tramite la quale do-



Fig. 5 La pagina principale di controllo

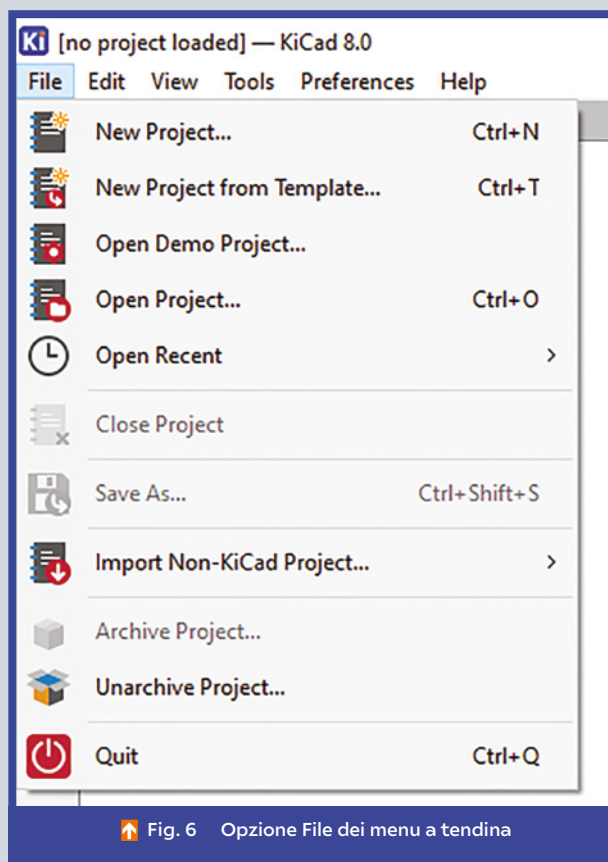


Fig. 6 Opzione File dei menu a tendina

vremo cercare la cartella in cui collocheremo il nostro nuovo progetto.

In questo caso è stata creata sul desktop una cartella di comodo chiamata MyKiCadPRJ, cartella in cui, come visibile in **Fig. 7**, non è ancora presente alcun file.

Nel campo File Name andiamo a scrivere il nome di nostro interesse (PrimoProgetto) quindi confermiamo cliccando Save e lasciando che il programma crei in modo autonomo la nuova sottocartella dedicata al progetto: notate infatti la spunta sull'opzione "Create a new folder for the project" vicino al pulsante Save.

Abbiamo riportato questa finestra di navigazione perché il suo uso sarà molto frequente in tutto l'ambito del progetto. Non appena salvato il nuovo progetto, la finestra di navigazione si chiude e il programma torna alla schermata principale, che ora mostra il nuovo progetto appena creato nella sezione Project Files, in precedenza vuota.

Se volessimo, a scopo didattico, andare a vedere cosa contiene la nuova cartella, dovremmo semplicemente cliccare (per ora, poiché esistono altre vie) ancora su "New Project" così da far comparire la stessa finestra di navigazione file di prima (**Fig. 7**), che noterete ora contiene un file progetto chiamato PrimoProgetto.kicad_pro.

Da questa verifica ricaviamo un'informazione molto utile in futuro: tutti i file progetto hanno estensione kicad_pro.

Dalla **Fig. 8** notiamo che il sistema mostra due file aggiun-

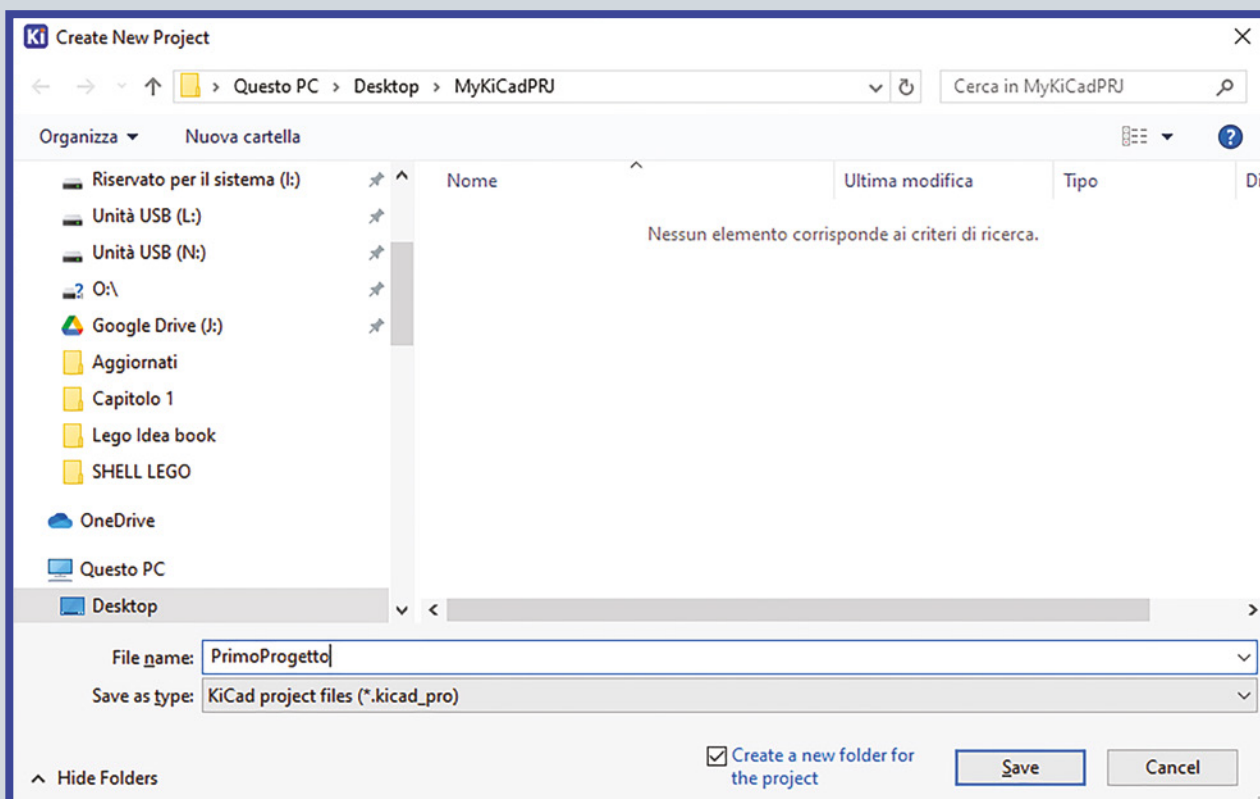


Fig. 7 Creazione di un progetto nuovo tramite la finestra di navigazione generale di Windows

tivi rispetto al nostro "PrimoProgetto": uno con estensione .kicad_sch e uno con estensione .kicad_pcb. Tuttavia, durante la verifica del contenuto della cartella, non li abbiamo visti. Dove si trovano?

La risposta è semplice: entrambi sono nella stessa cartella, ma non vengono visualizzati perché il filtro "Save as type" è impostato su .kicad_pro, quindi mostra solo i file di tipo progetto. Se fossimo nell'editor di schemi o PCB, vedremmo solo i file relativi agli schemi o ai circuiti stampati.

In realtà, la cartella del nuovo progetto, vista tramite il file

manager di Windows (Fig. 9), contiene esattamente tre file: progetto, schema e PCB.

New Project from Template

Funzionalmente esegue la stessa operazione vista per New Project, con la sola differenza che in questo caso consente di scegliere la base progetto da un template, ossia da una struttura preconfezionata, così da semplificare la progettazione sia dello schema elettrico che del circuito stampato.

Vediamo un esempio, cliccando su New Project from Template. Compare la finestra di Fig. 10, con la parte inferiore inizialmente completamente bianca (c'è solo la scritta TEMPLATE SELECTOR); nella parte superiore della finestra sono riportati tutti i template disponibili, ossia tutte quelle strutture precompilate che assomigliano a "qualcosa di esistente e comodo da copiare". Noi abbiamo cliccato su Arduino Pro mini: come da indicazioni nuove apparse, questo template consente di generare un progetto che contiene:

- un PCB già strutturato meccanicamente identico ad un Arduino Pro Mini
- la relativa parte di schema elettrico.

Attenzione ad un dettaglio importante: non si tratta di uno shield per Arduino Pro Mini, bensì di una scheda vuota avente esattamente la geometria di quel dispositivo. Viene da sé che Template differenti avranno funzioni e descrizioni differenti. Scelto il template si clicca su OK e quindi riprende esatta-

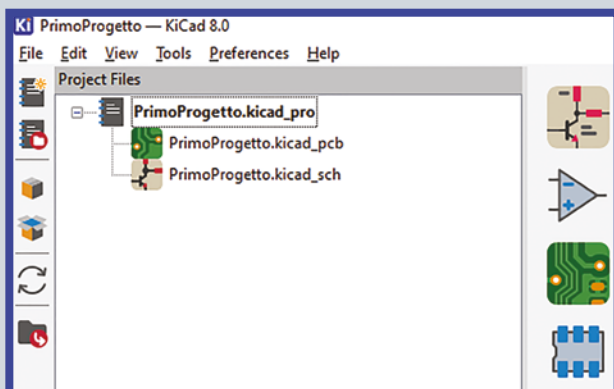
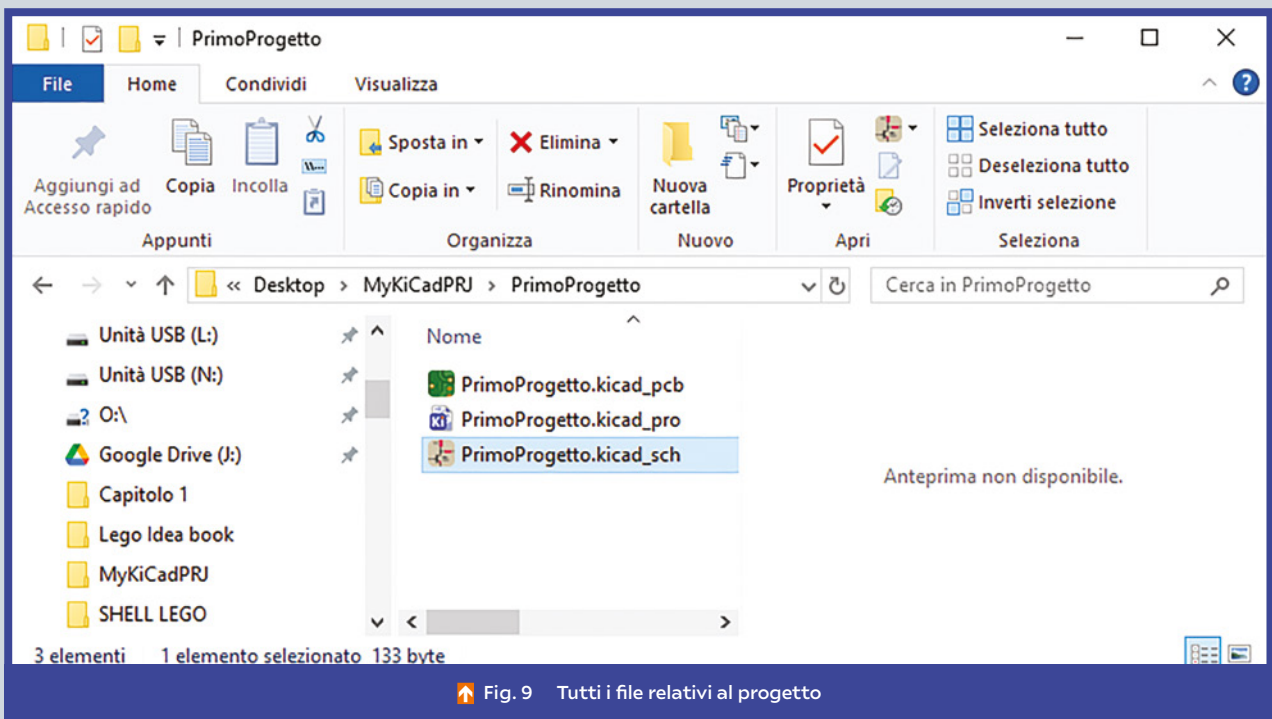


Fig. 8 La finestra principale mostra il nuovo progetto



mente passo per passo la stessa procedura vista al paragrafo precedente relativo al comando New Project, ossia si sceglie la cartella, si assegna il nome e si fa generare al sistema la nuova sottocartella, terminando il tutto con salva. Giusto per conferma a quanto descritto, andate ad esplorare con file manager di Windows la cartella di lavoro MyKiCadPRJ: noterete che ora sono presenti due cartelle ossia la “Primo-Progetto” fatta in precedenza e la “PrimoDaTemplate” appena creata. Logicamente anche quest’ultima conterrà i tre file di progetto e la cartella di backup.



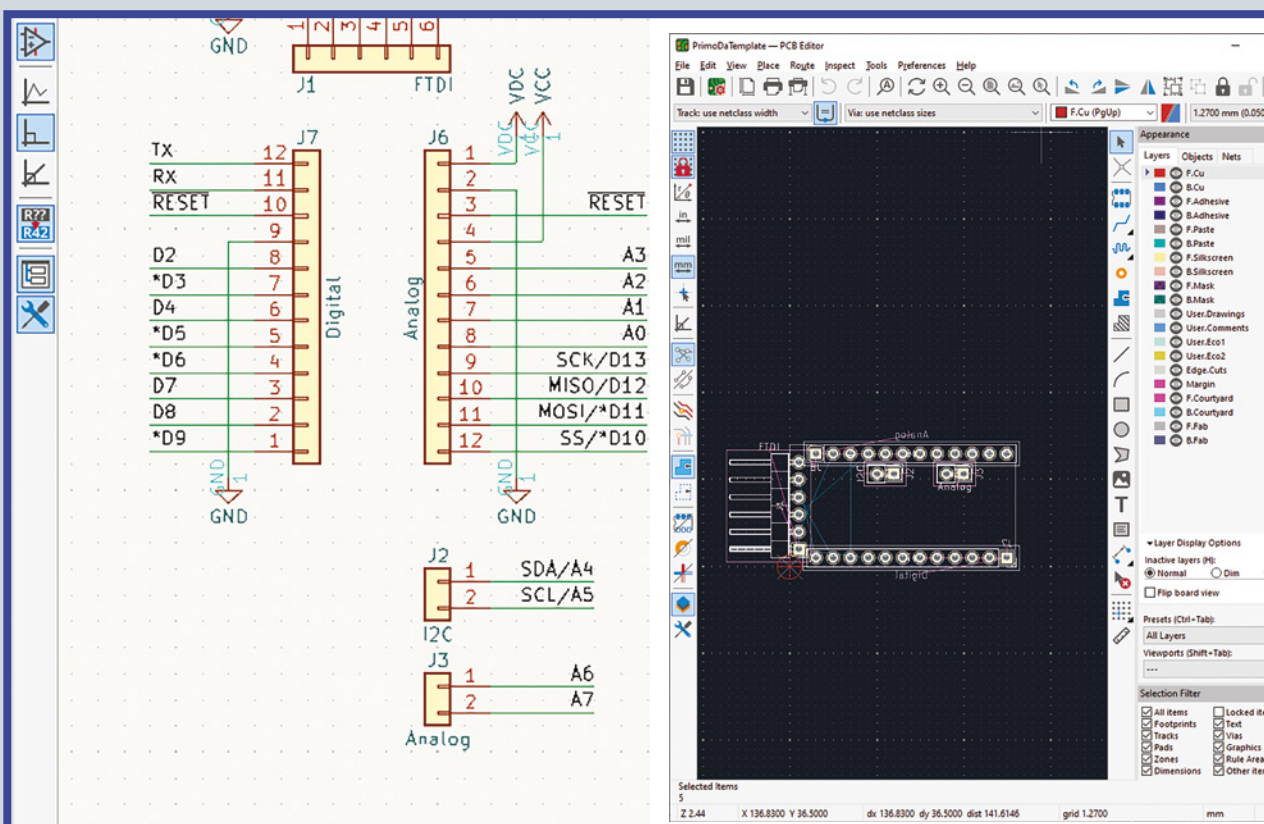


Fig. 11 Quando si crea un progetto da template, sia lo schema che il PCB contengono già i simboli di schema e i simboli di circuito stampato relativi ai componenti del template scelto

Close Project

Come intuibile dal comando stesso, chiude il progetto attualmente aperto e lo toglie dalla visualizzazione nell'area Project Files. Chiaramente il progetto è semplicemente stato chiuso

e può essere ricaricato in qualsiasi momento andando a selezionare i comandi Open Project oppure Open Recent. Volendo, potrebbe diventare un "modus operandi" quello di chiudere il progetto in corso prima di uscire completamente dall'ambien-

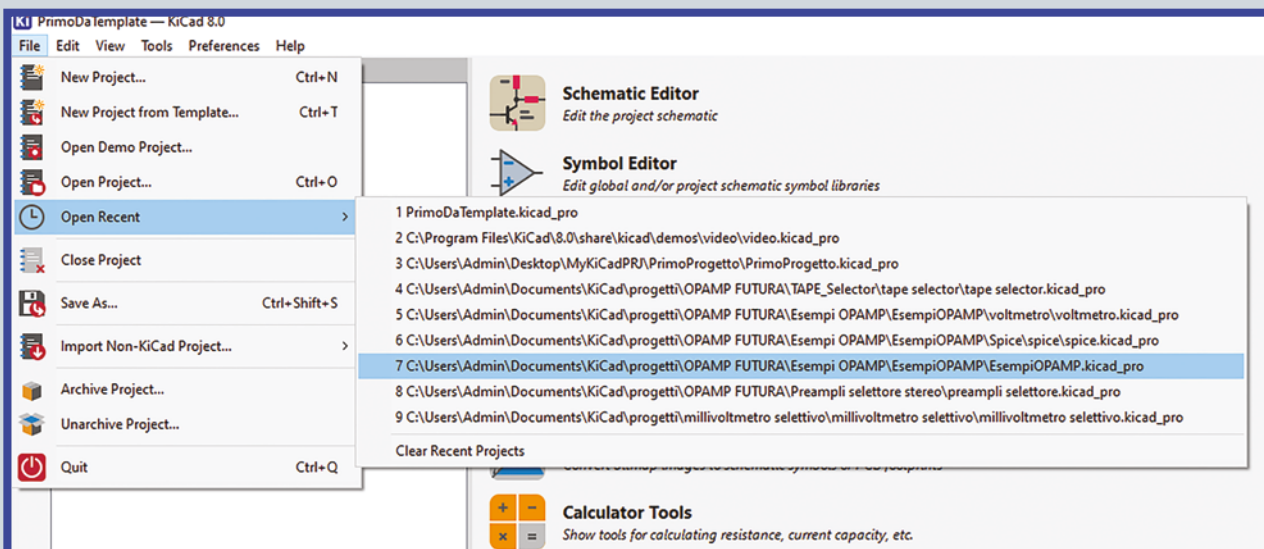


Fig. 12 Apertura progetto recente

te di sviluppo KiCad. Naturalmente non è una regola e non è obbligatorio farlo, poiché ad ogni chiusura volontaria di KiCad il progetto viene chiuso ed archiviato automaticamente nel modo corretto.

Save As

Come in tutte le applicazioni basate su OS Windows, consente di salvare il progetto corrente con un altro nome e in un'altra cartella. Quando invocato, richiama la solita finestra di navigazione file già vista più volte in precedenza, tramite la quale possiamo andare a cercare una locazione specifica, crearne una nuova e salvare tutto il progetto con un nuovo nome.

Import Non-KiCad project

Tramite questa opzione è possibile caricare e convertire in formato KiCad un progetto sviluppato con piattaforme EDA differenti. Ad oggi sono supportati quattro EDA e per la precisione CADSTAR, EAGLE, EasyEDASTandard e EasyEDAprofessional: possono essere selezionate tramite il sottomenu a tendina che compare quando si posiziona il cursore del mouse sull'opzione Import (**Fig. 13**). Quando si richiama questa funzione, l'EDA apre la solita finestra di navigazione file, con la quale si cerca il progetto da importare; essendo però un'operazione un po' evoluta, non la vedremo in questo momento, bensì quando ci saremo meglio impraticati con l'uso dell'interfaccia grafica principale e dell'editor di schemi poiché, praticamente certo, la conversione da un EDA a KiCad richiederà la conferma manuale di alcune operazioni, come per esempio il "match delle ratsnet". Per ora basti sapere che è possibile importare un progetto e convertirlo completamente.

Archive Project

Quando il progetto è stato completato, verificato in ogni sua parte anche a livello pratico e pertanto confermato e chiuso, è possibile creare un file compresso che contiene tutti i file di progetto della versione ultima, inclusi gerber, PDF, note,

datasheets e tutto quanto possa essere stato aggiunto al progetto nel corso della sua evoluzione. Come sempre il comando richiama la finestra di navigazione file e cartelle di Windows Explorer, con la quale si può scegliere dove salvare l'archivio e con che nome salvarlo; se non si apportano cambiamenti, il default è "nome_del_progetto.zip" e la cartella quella in cui è contenuto il progetto. Spendiamo qualche parola per i backup automatici che, di fatto, rappresentano una sorta di archiviazione: essendo pensato per applicazioni professionali, KiCad cerca di offrire tutte le garanzie possibili relativamente alla perdita accidentale di dati e conseguentemente ore di lavoro. Per questa ragione, infatti, dispone di un efficiente sistema di salvataggio e di backup automatici del progetto in corso in cui l'utente può configurare a proprio piacimento molti parametri di salvataggio quali frequenza giornaliera, livello di profondità dei backup, dimensioni massime dei file eccetera. Per quanto tutti i backup siano in formato compresso (.zip) essi comunque occupano spazio, per cui sarà compito dell'utente stabilire se i parametri predefiniti possano esser mantenuti oppure debbano essere adattati alle proprie esigenze. Attualmente la frequenza di backup è di 25 minuti, mentre quella di auto-salvataggio è di 5 minuti. Quanto alla gestione dei parametri, la vedremo più avanti quando in un capitolo dedicato del corso parleremo del menu a tendina Preferences.

Unarchive Project

E' l'operazione complementare dell'archiviazione: essa consente di selezionare un file progetto archiviato (zippato) e di estrarlo in una cartella a scelta.

Quit

Termina il programma e si comporta esattamente come la X di chiusura in alto a destra sulla schermata principale. Quando viene invocato, l'EDA provvede a salvare tutto il necessario oppure ad avvisare l'utente che qualcosa di aperto richiede l'intervento umano. Non è possibile chiudere la suite perdendo dati e ore di lavoro a meno che non lo si dica esplicitamente (per errore umano!) tramite messagebox da cliccare.

MENU EDIT

Visibile in **Fig. 14**, sono presenti le classiche opzioni di copia, incolla e taglia tipiche dell'editing in Windows, ma al momento sono tutte disattivate. Questi comandi potranno essere utili per funzioni future.

MENU VIEW

Come potete verificare in **Fig. 15**, è abbastanza scarso: contiene solamente tre opzioni di scelta che di fatto non fanno grandi cose. Vediamole una per una.

Refresh

Serve per aggiornare la lista dei file di progetto visibile in struttura a tendina nell'area Project Files.

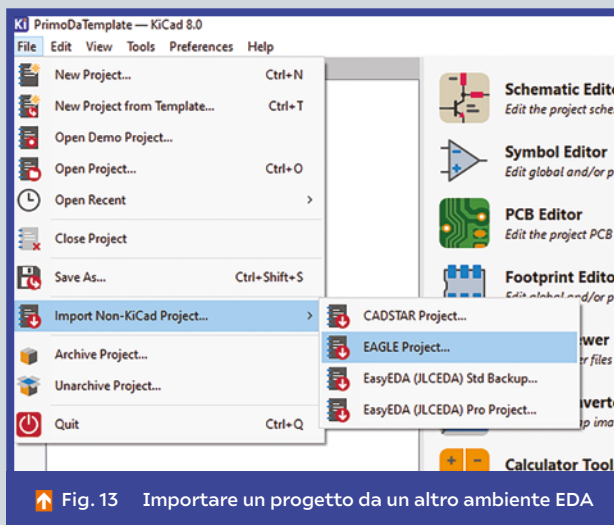


Fig. 13 Importare un progetto da un altro ambiente EDA

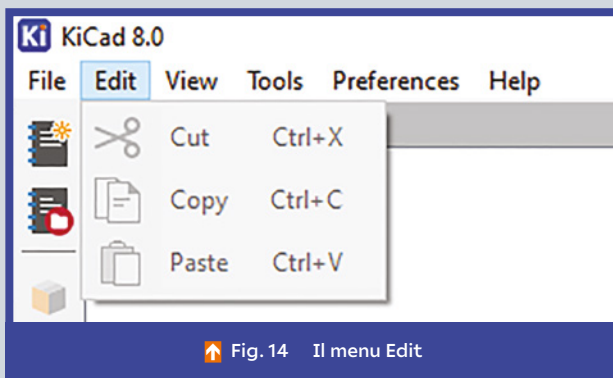


Fig. 14 Il menu Edit

Quando si lavora interamente all'interno dell'EDA, l'invocazione di questo comando (equivalente a F5 in Windows) non provoca alcun cambiamento nella visualizzazione. Tuttavia, se apriamo la cartella del progetto "PrimoDaTemplate" direttamente dal file manager di Windows, e non dall'EDA, ciò è normale: a volte è più pratico aggiungere o rimuovere file non strettamente legati all'EDA, come PDF di componenti specifici a completamento del progetto, direttamente tramite il sistema operativo.

Ora, all'interno di questa cartella, vogliamo aggiungere una sotto-cartella chiamata "Documenti PDF". Tuttavia, avendo effettuato questa modifica da Windows, KiCad non è a conoscenza del cambiamento finché non glielo comunichiamo. Per aggiornare la visualizzazione in KiCad, possiamo utilizzare il comando Refresh, equivalente al tasto F5.

Open Text Editor

Durante un progetto, capita spesso di dover prendere appunti, annotazioni o promemoria che non sono necessariamente legati al progetto, ma piuttosto al modo di lavorare personale di ciascun utente.

Alcuni preferiscono i "cheese-paper" (come vengono chiamati i foglietti volanti nei paesi anglofoni), altri usano file TXT sul PC, o un blocco appunti di carta riciclata (scelta apprezzata dal pianeta!).

Qualunque sia la preferenza, KiCad consente comunque di aprire un editor di testo a scelta direttamente dal PC.

La prima volta che questo comando verrà chiamato, KiCad

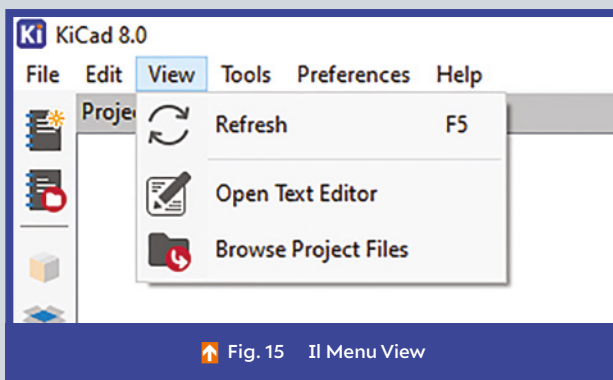


Fig. 15 Il Menu View

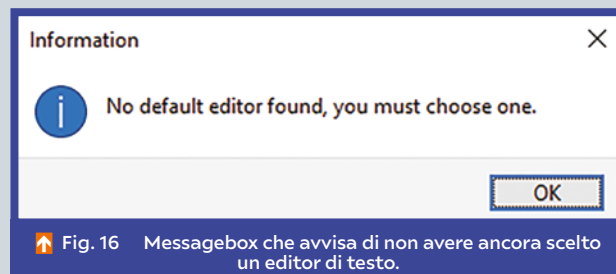


Fig. 16 MessageBox che avvisa di non avere ancora scelto un editor di testo.

vi comunicherà tramite la messagebox di **Fig. 16** che non ne avete ancora scelto uno per cui cliccando su OK aprirete sempre la solita finestra di navigazione file con cui sceglierete una tantum l'editor TXT che preferite.

Noi abbiamo scelto di usare Notepad.exe, localizzato nella cartella di sistema C:\Windows\system32.

Dopo aver confermato cliccando sul tasto Open, si avvierà notepad.exe e così ogni volta che utilizzerete il comando Open Text Editor.

A titolo di esercizio, provate ad aprire il text editor, create il nuovo documento "NoteDiLavorazione.TXT" e salvatelo nella cartella del progetto corrente (PrimoDaTemplate): noterete che KiCad ha tenuto traccia di questo nuovo file e lo ha aggiunto in automatico ai file di progetto visibili nell'area Project files.

Avreste potuto creare il documento fuori da KiCad, in ambito Windows, ma così facendo avreste dovuto usare F5/Refresh per aggiornare la vista file.

Browse Project Files

E' molto simile al comando Open Project visto nel menu File: la differenza sostanziale è che non ci sono filtri applicati per la scelta dei file quindi la finestra di esplorazione mostra tutti i file contenuti nella cartella relativa al progetto.

Vista così potrebbe sembrare una funzione piuttosto inutile, ma immaginate di dover andare a "pescare" un file già usato in un altro progetto solo per rivedere una certa cosa: tramite questa finestra lo potete cercare direttamente in ambiente KiCad e il sistema EDA lo tratterà in maniera autonoma, indipendente dal progetto, proprio come se lo steste facendo tramite file manager di Windows.

Ovviamente potreste fare la stessa cosa in ambito Windows; se vi va, questa è un'altra strada percorribile.

MENU TOOLS

Prima di parlare del menu, è opportuno precisare cosa si intende per "Tools" in ambito "KiCad". La traduzione letterale, in Italiano, avrebbe il significato di "attrezzi, utensili" e in qualche modo la traduzione è appropriata.

In questo EDA, come in altri del resto, gli attrezzi sono dei programmi, delle applicazioni, ognuno specifico per una certa funzione. Per esempio l'edizione degli schemi elettrici ha un suo tool chiamato Schematic Editor, così come la creazione dei circuiti stampati ne ha uno chiamato PCB editor.

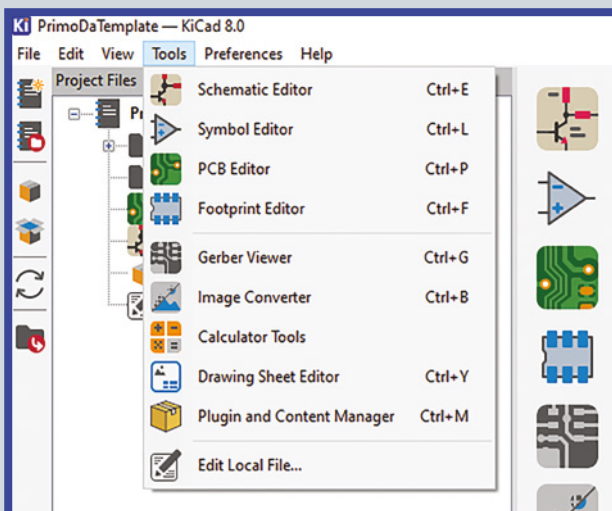


Fig. 17 Il menu Tools ripropone esattamente le icone dei tools visibili nella pagina principale. La sola differenza è il comando Edit Local File

In pratica, ogni volta che si invoca un tool, non si fa altro che aprire un programma specializzato.

Nell'attuale versione di KiCad sono presenti nove strumenti predefiniti, accessibili sia dal menu Tools che nella parte destra dell'interfaccia principale (Fig. 17).

Descrivendo il menu Tools, copriamo quindi anche l'area di destra dell'interfaccia grafica.

In questa parte del corso, ci limiteremo a mostrare un'immagine della schermata principale; approfondiremo ogni tool nei prossimi capitoli.

Schematic Editor

Avvia il programma per la generazione degli schemi (Fig. 18): si tratta di un programma autonomo che potrebbe vivere di vita propria, ma viene richiamato tramite la schermata principale. In linea di massima, quasi tutti i tools potrebbero vivere di vita propria, ma invocarli da interfaccia comune è sempre la strada più veloce.

Qui abbiamo aperto lo schema del nostro primo progetto da template: notate come esista già una parte di schema. Fossimo partiti da uno schema nuovo, ovviamente esso sarebbe vuoto. Nota di interesse: quanto più è ampia la finestra di visualizzazione dell'editor, tanto maggiore è il numero di icone mostrate.

Symbol Editor

Avvia il programma per la generazione dei simboli elettrici (Fig. 19). Accede a tutte le librerie esistenti in lettura e scrittura con controllo totale e consente di crearne di nuove. Occorre conoscere molto bene l'applicazione se si vuole evitare di danneggiare le librerie esistenti.

PCB Editor

Avvia il programma per la creazione dei circuiti stampati (Fig. 20). Schematic e PCB editor sono interagenti, ossia il PCB editor tiene conto dei collegamenti presenti nello schematic editor e viceversa. E' pertanto sempre possibile modificare uno schema elettrico ed aggiornare il PCB, ma anche viceversa.

In pratica, un progetto rappresenta uno schema ed il suo PCB, sempre relazionati l'un l'altro.

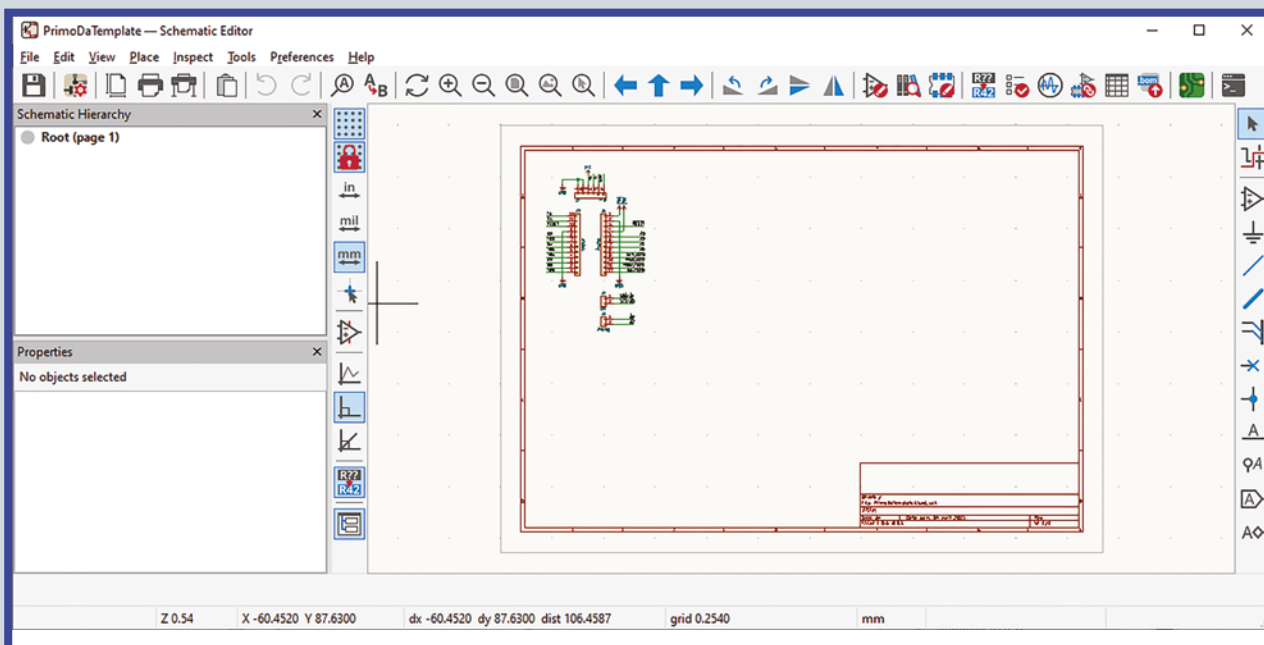


Fig. 18 Tool Schematic Editor

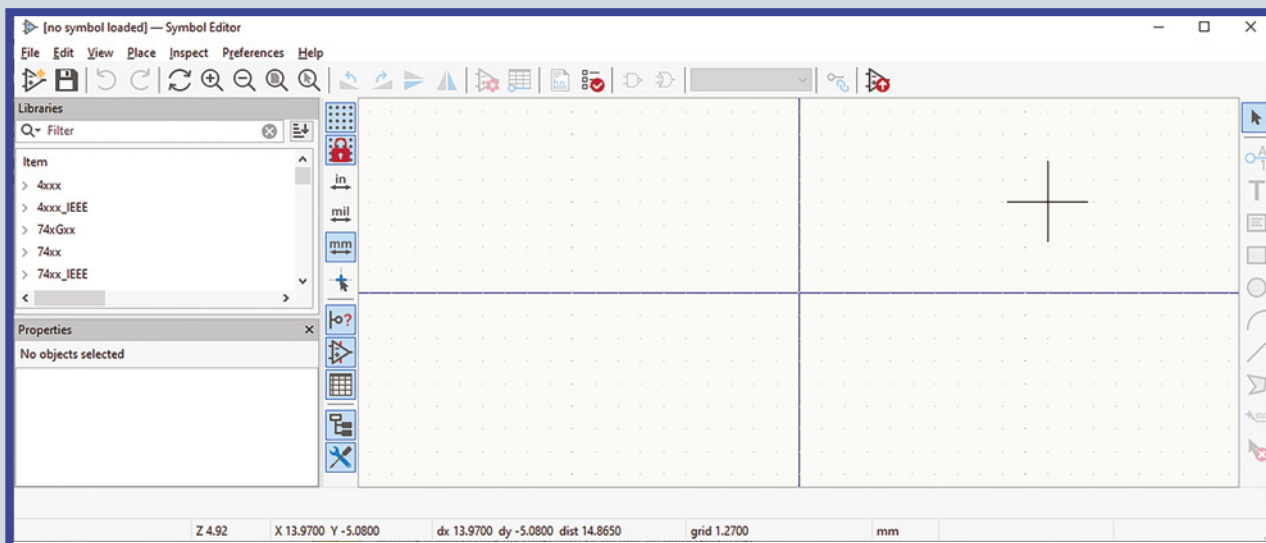


Fig. 19 Tool Symbol Editor

Come per lo schematic, le icone mostrate sono proporzionate alle dimensioni della finestra e questa regola vale sempre per "ogni tool di editing KiCad".

Footprint Editor

Avvia il programma per la creazione dei Footprint, ossia delle geometrie meccaniche specifiche per ogni componente elettrico / elettronico (**Fig. 21**).
Accede a tutte le librerie esistenti in lettura e scrittura e consente di crearne di nuove.
Come nel caso dell'editor di simboli, occorre attenzione perché l'inesperienza e le disattenzioni possono causare danni.

Gerber Viewer

Avvia il programma che permette di controllare i file Gerber (**Fig. 22**) che, per quanti non lo sapessero ancora, descrivono il PCB in modo interpretabile dalle macchine che lo producono e montano.

Image Converter

Avvia il programma che consente di realizzare simboli elettrici nuovi partendo da immagini grafiche.
Questo tool, visibile in **Fig. 23**, è particolarmente utile quando si devono utilizzare componenti molto nuovi, non ancora inseriti nelle librerie fornite, per i quali occorre creare sia il simbolo elettrico che il footprint.

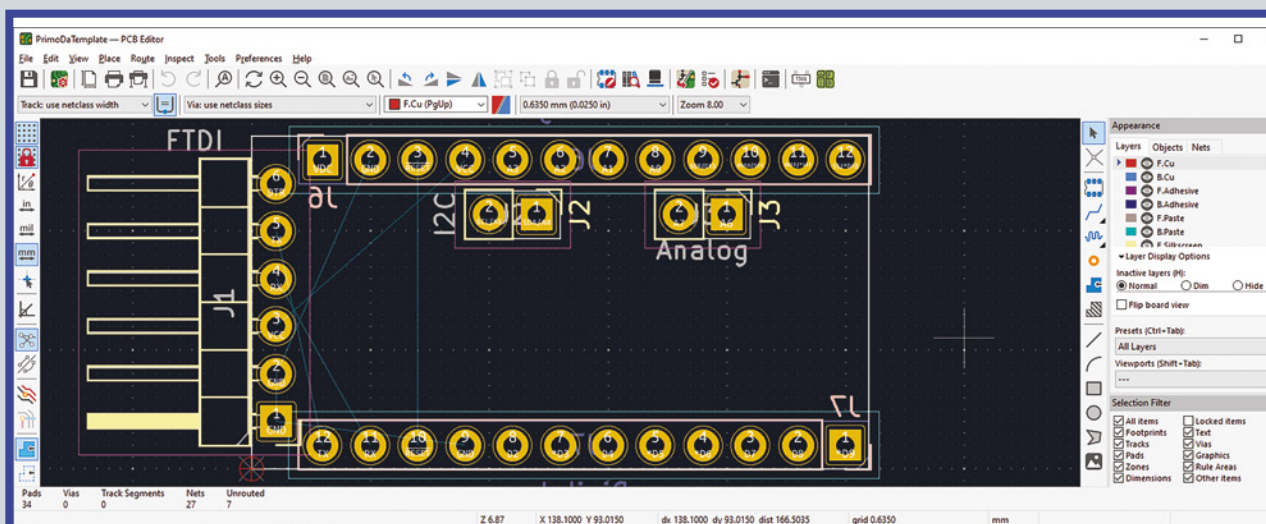


Fig. 20 Tool PCB Editor

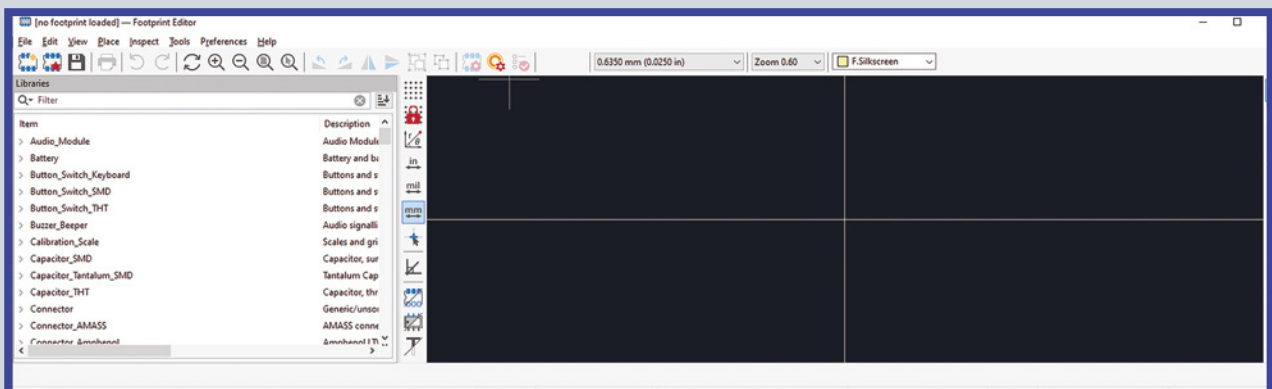


Fig. 21 Tool Footprint Editor

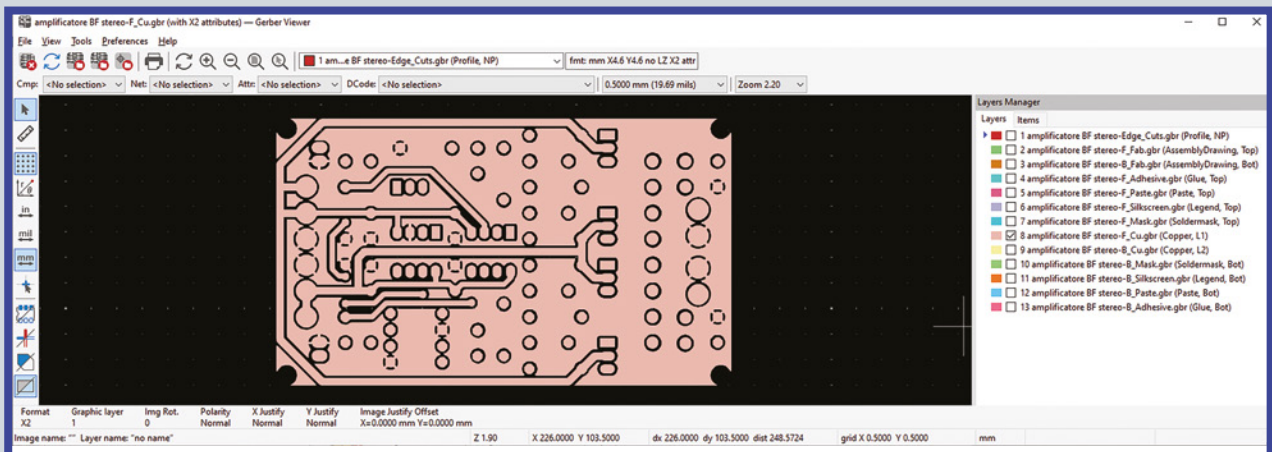


Fig. 22 Tool Gerber Viewer

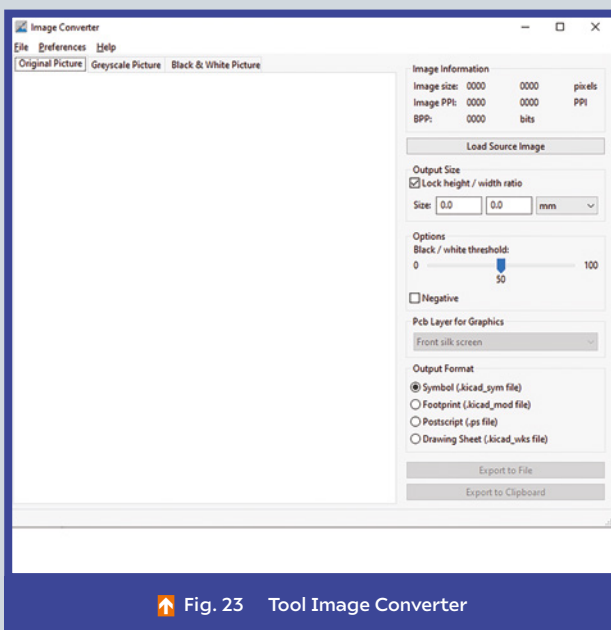


Fig. 23 Tool Image Converter

Calculator Tools

Si tratta di un programma di aiuto, che contiene alcune formule di calcolo piuttosto ricorrenti in progettazione.

Si presenta come in **Fig. 24**, dove noi abbiamo scelto di mostrare una formula che probabilmente è nota alla maggior parte dei progettisti, ossia quella per il calcolo della tensione di uscita nei regolatori di tensione a tre pin, noti il valore della tensione di riferimento e una delle resistenze di retroazione.

Questo programma offre molte altre facilitazioni come il calcolo di valori non standard di resistenze sulla base di combinazioni serie e parallelo di valori standard, il codice colori delle resistenze, il calcolo di attenuatori, eccetera.

Tutto ciò verrà approfondito a tempo debito.

Drawing Sheet Editor

Avvia il programma visibile in **Fig. 25**, che consente di disegnare e/o personalizzare lo sfondo del foglio su cui verrà disegnato lo schema elettrico.

E' possibile modificare quelli esistenti così come di crearne di nuovi.

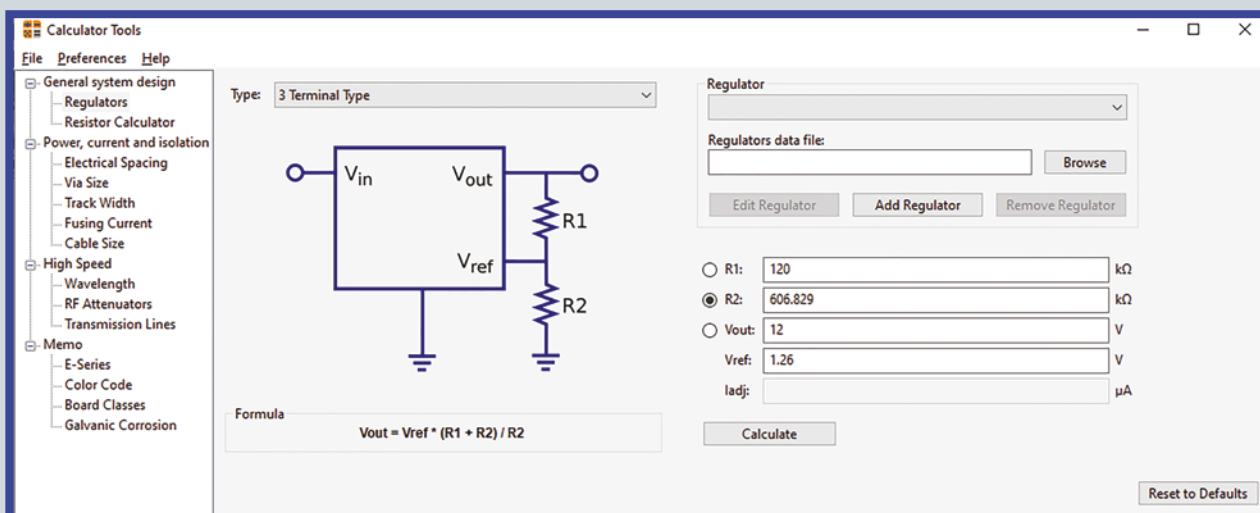


Fig. 24 Tool Calculator Tools

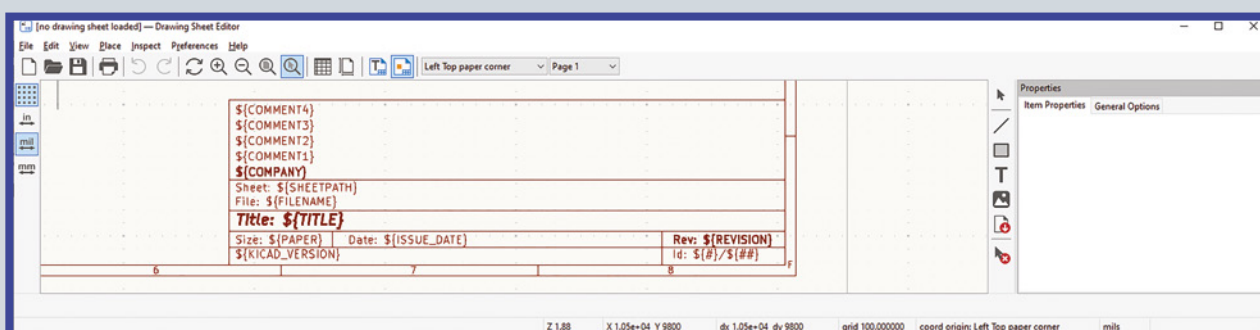


Fig. 25 Tool Drawing Schematic Sheet Editor

Plugin And Content Manager

Probabilmente questo è l'unico tool a non essere un programma indipendente, o forse lo è anche, ma di fatto si comporta come se fosse una finestra prioritaria o modale del programma di base stesso, in quanto una volta chiamato, la finestra che compare diventa prioritaria rispetto all'interfaccia principale che non può essere più raggiunta se non chiudendo questa finestra.

Indipendentemente dal suo funzionamento, questa funzione consente di caricare plugin, ossia componenti aggiuntivi complementari o ausiliari per gli altri strumenti.

Questi plugin, sviluppati sia da KiCad che da terze parti, possono essere gratuiti o a pagamento e fornire funzioni utili o semplici miglioramenti grafici.

Per esempio, nella **Fig. 26** abbiamo scelto di installare il plugin Interactive HTML BOM, che, quando si lavora in PCB Editor, consente di generare una pagina web locale. Questa mostra il punto esatto del PCB in cui si trova un componente specifico, semplicemente selezionandolo dall'elenco componenti. È una funzione particolarmente utile quando la lettura del cartaceo diventa più impegnativa con il passare degli anni.

Ovviamente esistono molti plugin, alcuni utili altri "belli". Li vedremo più avanti dato che, per vederne il funzionamento, occorre spesso essere all'interno di altri tools che al momento non conosciamo.

Edit Local File

Tutti i file generati da KiCad sono molto complessi, ma sono in

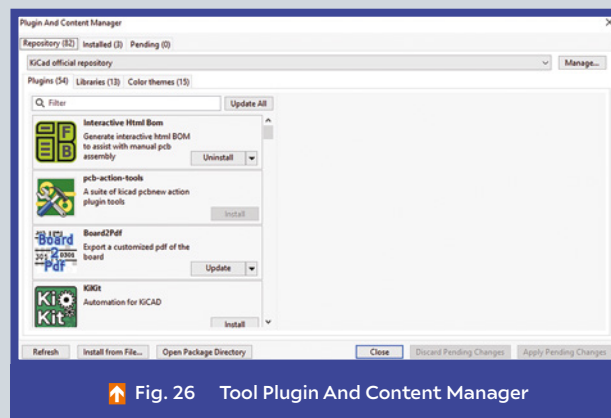


Fig. 26 Tool Plugin And Content Manager

formato testo. Con questo tool, è possibile aprirli utilizzando il text editor selezionato in precedenza per visualizzarli. Con una buona conoscenza del linguaggio in cui sono scritti, è anche possibile modificarli manualmente. Per chi fosse interessato a esplorare questo argomento, saremmo lieti di dedicare uno spazio apposito. Ci limiteremo a dire che questo tool consente di aprire e, a proprio rischio, modificare ogni singolo file di progetto generato da KiCad. Quando il comando viene invocato, si apre la finestra di navigazione file di Explorer, con cui abbiamo aperto il file PrimoDaTemplate.kicad_pro **Fig. 27**, che ricorda una sorta di listato in linguaggio C.

MENU PREFERENCES

Visibile in **Fig. 28**, dentro questa voce sono state raggruppate alcune funzioni un po' a sé stanti, ossia in qualche modo legate in modo trasversale ai vari tools piuttosto che alla stessa interfaccia grafica principale. Vediamole una per una.

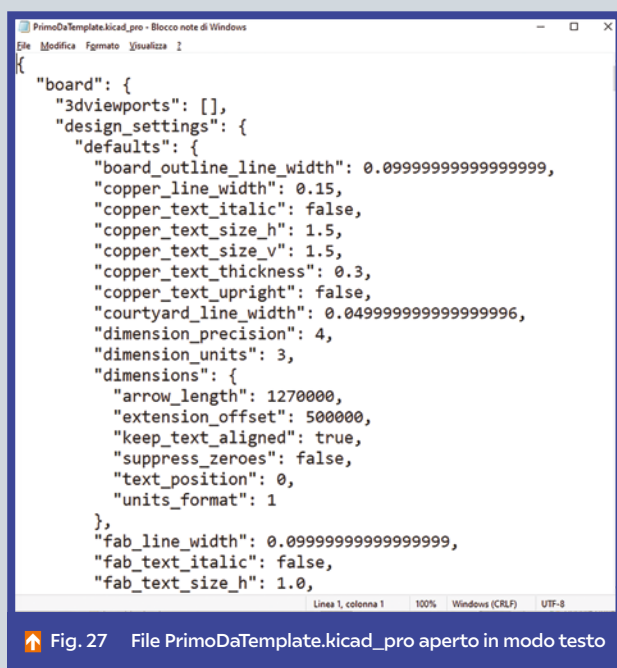


Fig. 27 File PrimoDaTemplate.kicad_pro aperto in modo testo

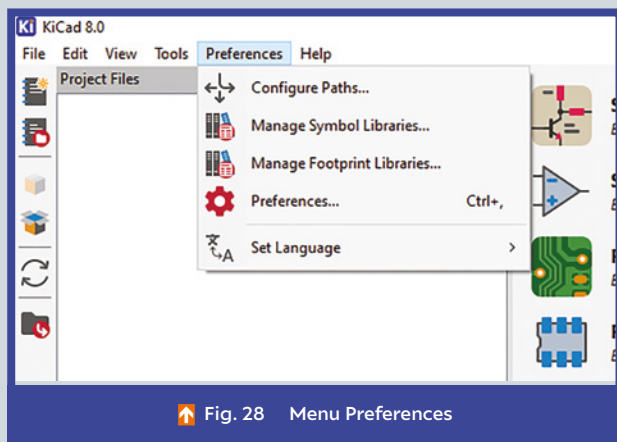


Fig. 28 Menu Preferences

Configure Path

Aprire la finestra di **Fig. 29**, in cui si impostano tutti i percorsi dove sono immagazzinati i file necessari alla suite. Se volete potete anche modificarli; noi abbiamo scelto di lasciarli così come sono.

Manage Symbol Libraries

Aprire la finestra di **Fig. 30**, dove è possibile visualizzare e modificare ogni singolo parametro descrittivo di ciascuna libreria di simboli, dal nickname al percorso di archiviazione fino al formato. KiCad ha scelto nomi piuttosto intuitivi, sebbene siano in inglese e potrebbero non essere di gradimento per tutti. In questo caso, tuttavia, non abbiamo apportato modifiche.

Manage Footprint Libraries

Aprire la finestra di **Fig. 31**, che a prescindere dai parametri, è praticamente identica alla precedente. Mentre la prima consente di gestire le librerie di simboli elettrici, questa agisce sulle librerie di footprint, ossia le geometrie fisiche dei componenti. Anche qui vale quanto sopra e noi abbiamo scelto di lasciare tutto così com'è.

Preferences

Richiama una piccola finestra che apre invece un intero mondo in cui è possibile andare a modificare i parametri di funzionamento di ogni singolo tool.

In **Fig. 32** abbiamo visualizzato la prima schermata di preferenze, quella in cui sono presenti i tempi per il salvataggio e backup automatici, cui abbiamo già accennato nel corso di questo primo capitolo.

Poiché esistono diverse centinaia di parametri personalizzabili, a tutto ciò dedicheremo un capitolo specifico a fine corso, dato che per sapere "cosa andare a modificare" occorre prima sapere dove si usa e cosa comporta. Pertanto qui ci limiteremo a dire che ogni singolo tool di lavoro è personalizzabile.

Set Language

Aprire il menù a tendina di **Fig. 33**, dove si può scegliere il linguaggio di tutta la suite.

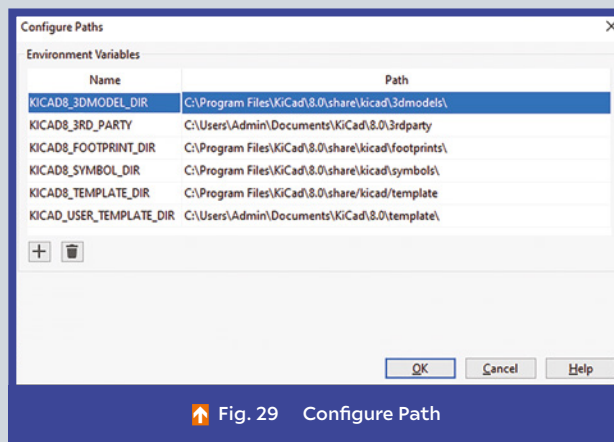


Fig. 29 Configure Path

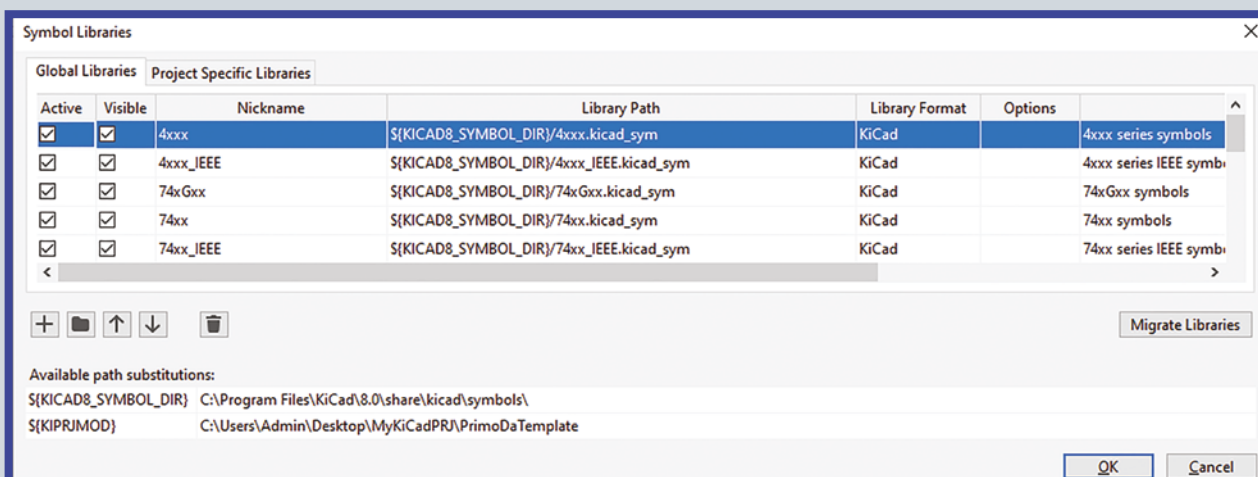


Fig. 30 Manage Symbol Libraries

Avremmo anche potuto scegliere l'italiano, ma secondo noi è meglio lasciare l'Inglese per una serie di ragioni:

- tutta la suite è stata pensata in inglese poiché la maggior parte dei CAD è ed è sempre stata in inglese;
- le traduzioni spesso sono fuorvianti e talvolta contengono errori di concetto, per quanto abbiamo verificato che quelle in Italiano siano piuttosto ben fatte;
- imparando ad usare direttamente l'inglese si prende subito confidenza con terminologie internazionali utilizzate in ambito produttivo.

Il nostro suggerimento spassionato, quindi, è quello di imparare ad usare in inglese qualsiasi Suite tecnica perché nel lungo termine mantiene il significato e semplifica la vita.

MENU HELP

Siamo giunti all'ultimo menu, quello di HELP (Fig. 34), che ovviamente include tutta una serie di Aiuti e di "suggerimenti".

Help

Questo comando tenta di aprire direttamente da PC il Manuale della Suite, ma, non essendo al momento fornito col pacchetto di installazione, vi comunicherà di non averlo trovato e vi chiederà se volete accedere al manuale online al collegamento <https://docs.kicad.org/8.0/it/kicad/kicad.html>. Al momento è questo, ma versioni successive potrebbero trovarsi in altro luogo.

Il manuale online, a nostro avviso, è fatto piuttosto bene, ma indirizzato a tecnici che già lavorano con gli ECAD e che pertanto partono già con delle solide basi. Se volete potete anche dargli un'occhiata, visto che esiste anche in Italiano e ben tradotto.

Getting Started With KiCad

Anche questo comando tenta di aprire direttamente da PC un tutorial che nuovamente non troverà, per cui KiCad vi comuni-

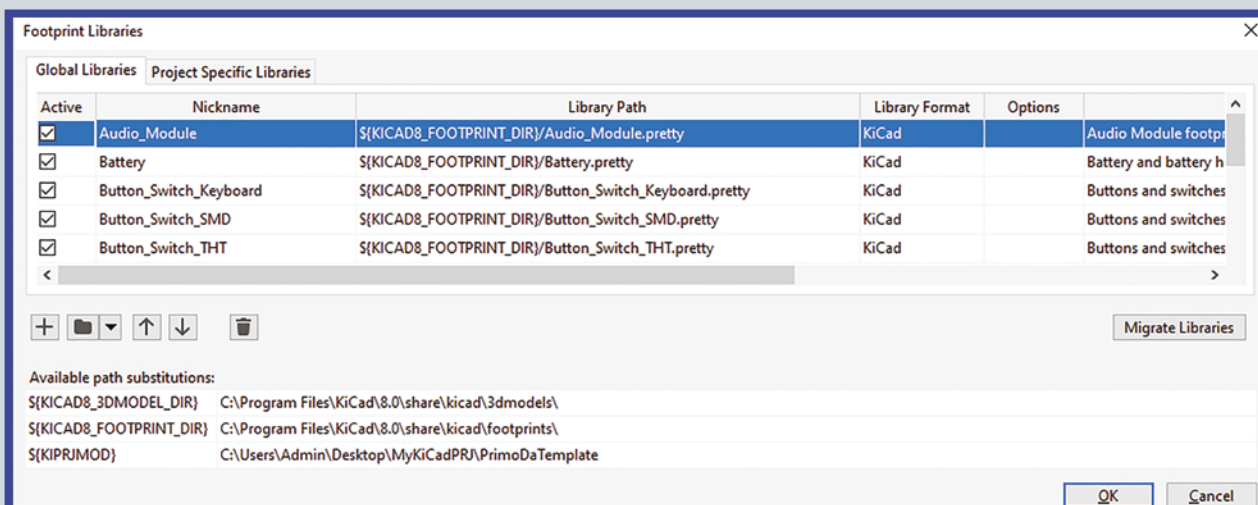


Fig. 31 Manage Footprint Libraries

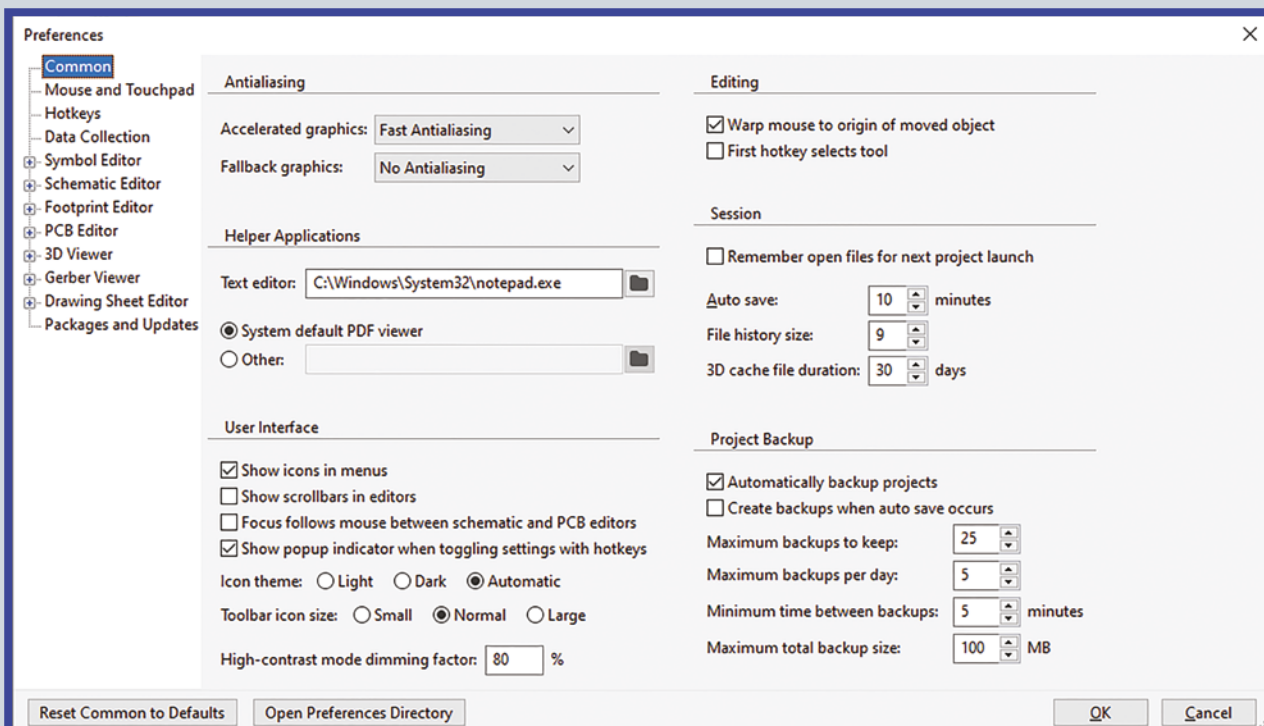


Fig. 32 Preferences

cherà di non averlo trovato e vi chiederà se volete accedere al documento online al collegamento https://docs.kicad.org/8.0/it/getting_started_in_kicad/getting_started_in_kicad.html.

Si tratta di un tutorial con cui potreste iniziare a fare qualcosa fin da subito, sempre nell'ipotesi che abbiate già delle solide basi ECAD.

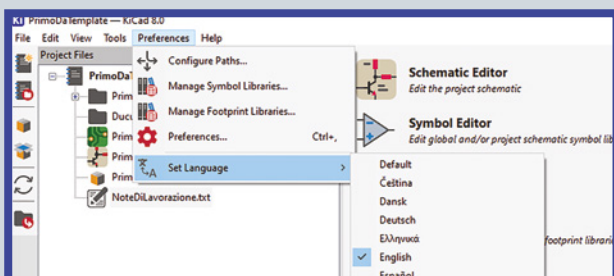


Fig. 33 Set Language

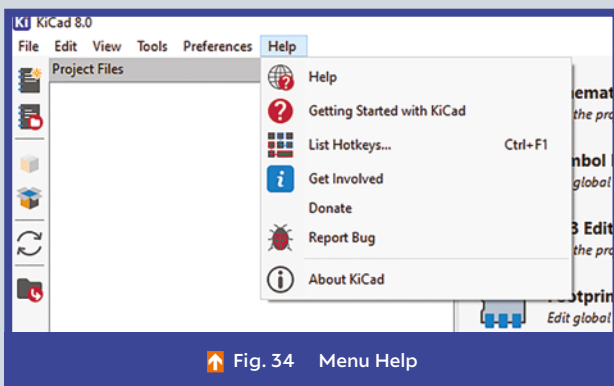


Fig. 34 Menu Help

List HotKeys

Apri la finestra di **Fig. 35**, che riassume tutti i collegamenti rapidi, noti come hotkeys, cioè combinazioni di tasti che eseguono specifici comandi nel programma.

Di solito, anche se non sempre, ogni hotkey è associato a un comando disponibile tramite icona, menu, pulsante o altra opzione. In KiCad ce ne sono moltissimi, con funzioni che variano a seconda dello strumento attivo.

Servono memoria e pratica per padroneggiarli, ma con il tempo diventano più familiari.

Get Involved

Come abbiamo già detto in precedenza, KiCad è una Suite open source al cui sviluppo può partecipare chiunque sia in grado di farlo.

Cliccando su questa opzione viene aperta una pagina del Sito in cui trovate spiegato cosa occorre fare per partecipare allo sviluppo in maniera attiva.

Donate

Beh, abbastanza chiaro: tutto ciò ha un costo di sviluppo che immaginiamo sia parecchio pesante.

Però il pacchetto di installazione viene fornito a titolo gratuito e senza vincoli troppo restrittivi.

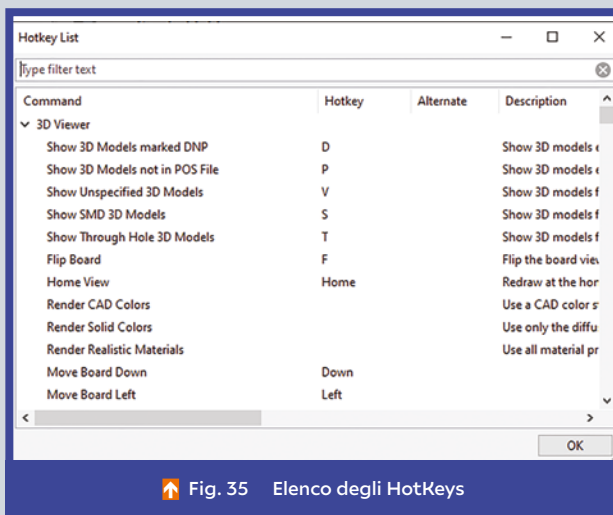


Fig. 35 Elenco degli HotKeys

Giustamente loro chiedono aiuto economico sotto forma di donazioni.

Noi non ci schieriamo da nessuna parte: riteniamo che ognuno abbia in diritto di poter scegliere.

Sappiamo di persone che non hanno fatto donazioni così come di persone che hanno scelto di farne.

Report Bug

Pur non facendo donazioni o non partecipando in modo attivo allo sviluppo, è possibile comunque fornire il proprio contributo segnalando anomalie di programma riscontrate durante il normale utilizzo.

A tal proposito, questo comando apre un collegamento internet in cui, dopo essersi registrati, potete segnalare l'anomalia riscontrata e ciò che avete fatto per causarla.

About KiCad

Aprire la finestra informativa della suite in uso (Fig. 36).

Contrariamente ad altre suite, questa vi dice praticamente

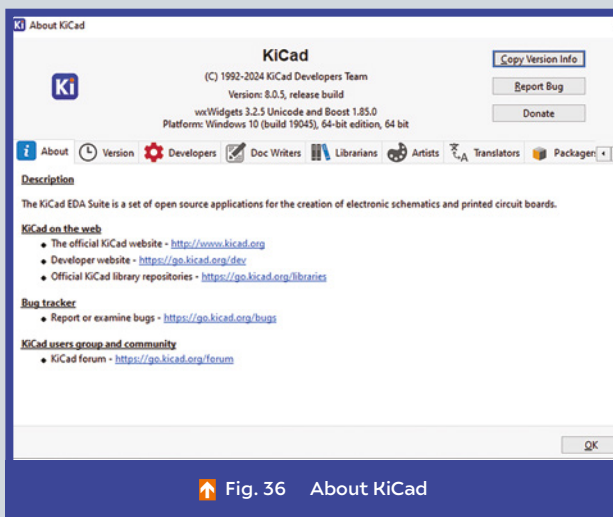


Fig. 36 About KiCad

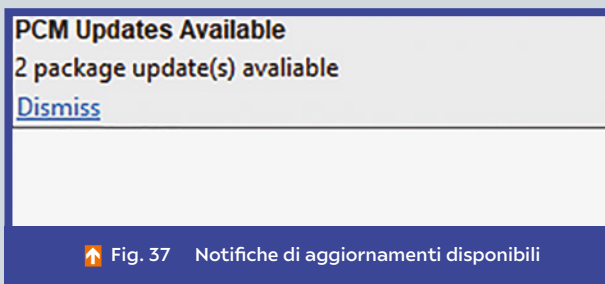


Fig. 37 Notifiche di aggiornamenti disponibili

tutto: chi sviluppa, chi scrive i documenti, chi fornisce le librerie, chi le grafiche, chi le traduzioni, le varie versioni di tutti i tool, eccetera.

ULTIMO MA NON ULTIMO...

Fate riferimento alla Fig. 5: a volte può apparire un numero bianco su sfondo rosso in basso a destra nella finestra principale, che indica la presenza di aggiornamenti disponibili per i plugin installati. Cliccando su questo numero, si apre una finestra che mostra i dettagli degli aggiornamenti disponibili (Fig. 37).

In generale, quando questa notifica appare, è visibile anche in altre parti dell'interfaccia grafica principale.

Come mostrato in Fig. 5, un numero rosso appare anche accanto all'icona dei plugin, segnalando che uno o più plugin installati hanno aggiornamenti online disponibili.

Aprendo la finestra dei plugin, troverete il plugin con l'aggiornamento. In questo caso, come illustrato in Fig. 38, si tratta di Board2Pdf, un plugin che consente di generare PDF di alta qualità a partire dal PCB.

NOTE CONCLUSIVE

Siamo giunti così al termine di questo primo capitolo dedicato alla suite KiCad. Abbiamo esplorato le funzionalità principali e i vari menu, gettando le basi per un utilizzo più approfondito del software.

Con il prossimo capitolo inizieremo a vedere nel dettaglio i vari tools, partendo dallo Schematic Editor, per comprendere come creare e gestire progetti di circuiti stampati in modo efficace. Continuate a seguire il corso per scoprire tutte le potenzialità di KiCad e per sviluppare una conoscenza pratica e completa dell'ambiente di progettazione. □

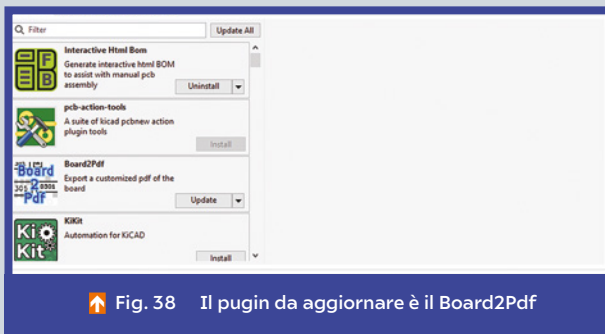


Fig. 38 Il pugin da aggiornare è il Board2Pdf

Entra nel mondo



cod. KSR19
€ 44,00

Robot con codifica meccanica 5 in 1 - in kit

Robot didattico in scatola di montaggio con codifica STEM (Science, Technology, Engineering, Mathematics) particolarmente adatto per principianti, pensato per stimolare l'utente ad avvicinarsi all'ingegneria di costruzione delle macchine con codifica manuale.

cod. SOLARDINO
€ 15,00



Robot Solare 3 in 1 - in kit

Particolarmente indicato come gioco didattico per bambini a partire dagli 8 anni e non solo, questo kit permette di assemblare ed ottenere 3 diversi robot alimentati ad energia solare, tra cui un dinosauro, un insetto e una trivella. Quando il sole non è disponibile è possibile utilizzare una luce alogena come alimentazione.



cod. KSR10N
€ 58,00

Interfaccia USB
per braccio robotico

cod. KSR10/USB | € 53,00

Braccio robotico

Braccio meccanico, dotato di cinque motori, cinque articolazioni con unità di controllo via cavo a cinque comandi.

cod. KSR18
€ 45,00



Robot TOBBIE II - in kit

Può funzionare in due differenti modalità: "Seguimi" e "Esplora". Nella modalità "Seguimi" è in grado di seguire qualsiasi cosa gli si avvicini. Quando funziona in modalità "Esplora" è in grado di evitare gli ostacoli che trova davanti a sé trovando sempre un nuovo percorso alternativo.

della robotica

cod. KSR20
€ 92,00



Robot TOBBIE II con MICRO:BIT - in kit

Robot a 6 gambe con corpo libero di ruotare, dotato di tutti i componenti elettronici necessari per renderlo attivo e interattivo. Ideale in combinazione con Micro:bit. È possibile programmarlo in tutta semplicità con Javascript Blocks Editor o Python Editor sul PC portatile o smartphone.

cod. KSR14
€ 24,50



Robot SOLAR ROVER - in kit

Particolarmente indicato come gioco didattico per bambini (da 8 anni in su). Questo kit permette di assemblare un mini rover solare con 6 ruote. Dispone di un sistema di trasmissione a ingranaggi planetari, piccoli bracci robotici, un pannello solare e un piccolo motorino.

cod. KSR17
€ 38,00



Robot solare e idraulico 12 in 1 - in kit

Permette di realizzare fino a 12 differenti modelli di robot con movimento idraulico e alimentati tramite energia solare. Il kit viene fornito completo di un manuale ricchissimo di illustrazioni.

cod. SOLARDOG
€ 23,00



Cane Robotico in kit

Particolarmente indicato come gioco didattico (da 8 anni in su), il kit permette di assemblare un cane robotico alimentato da 2 batterie o da una cella solare. Un motore DC permette al robot di muoversi.

cod. KSR16
€ 32,00



Kit per auto con motore ad aria

Kit di montaggio per auto con motore alimentato ad aria compressa. È composto da 102 parti di montaggio estremamente facili da assemblare tra di loro. La macchinina è in grado di percorrere fino a 50 metri di distanza.

Airbus e Toshiba studiano la superconduttività

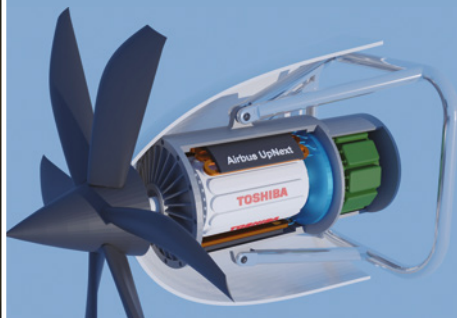
Airbus UpNext e Toshiba Energy Systems & Solutions Corporation coopereranno nello studio e la messa a punto di tecnologie superconduttive per i futuri aeromobili alimentati a idrogeno; scopo del progetto è sviluppare un motore superconduttore da due megawatt.

Nella ricerca per decarbonizzare il settore dell'aviazione, gli aeromobili alimentati a idrogeno sono una delle soluzioni promettenti per raggiungere emissioni nette pari a zero entro il 2050. Le tecnologie superconduttive offrono un vantaggio unico per questi aeromobili, utilizzando idrogeno liquido a -253°C come carburante ma anche come refrigerante per smaltire in maniera efficiente i sistemi di propulsione elettrica. La tecnologia criogenica potrebbe consentire una trasmissione di potenza pressoché inalterata all'interno dei sistemi elettrici dell'aeromobile, migliorandone significativamente l'efficienza energetica e le prestazioni.

La partnership con Toshiba rappresenta un'opportunità per spingersi oltre i limiti dei motori elettrici convenzionali o parzialmente superconduttori, in modo da rendere una realtà la propulsione elettrica alimentata da fuel-cell a idrogeno per i futuri aerei Airbus e contribuire al progresso della tecnologia dei motori superconduttori per soddisfare le esigenze dell'industria aerospaziale.

L'esperienza di Toshiba nella tecnologia superconduttiva per flussi di corrente elevati, nella tecnologia di azionamento del motore per un controllo preciso della corrente e nella tecnologia avanzata dei macchinari rotanti per un funzionamento stabile e ad alta velocità, costituisce una solida base di partenza.

www.airbus.com



L'orologio atomico in una valigetta 24ore

Presso l'università del Colorado Boulder si trova uno degli orologi più precisi e accurati al mondo, la cui particolarità è di essere ottico e non a microonde, come invece è la maggior parte degli orologi atomici. È basato sull'atomo di stronzio, che pulsa a una frequenza di 429 terahertz, la stessa della luce nella parte inferiore della regione rossa dello spettro visibile; questa frequenza è alla base dell'incredibile precisione dell'orologio (gli orologi atomici a microonde operano nell'ordine dei gigahertz).

Vector Atomic di recente ha lavorato su questa tecnologia per sviluppare un orologio abbastanza piccolo da stare in una scatola delle dimensioni di una 24ore, ma abbastanza preciso che potrebbe perdere un secondo in 32 milioni di anni.

Nell'ultimo anno, tre diverse aziende hanno sviluppato le proprie versioni di orologi atomici ottici compatti: oltre a Vector Atomic, ci sono anche Infleqion, a Boulder, in Colorado, e QuantX Labs, con sede ad Adelaide, in Australia. Fuori dal laboratorio, questi orologi promettono una maggiore resilienza e un backup del GPS per applicazioni militari, nonché per data center, istituzioni finanziarie e reti elettriche e potrebbero consentire un futuro di GPS più precisi, con una risoluzione di posizionamento centimetrica, abbastanza esatta da mantenere le auto a guida autonoma nelle loro corsie, consentire ai droni di far arrivare le consegne sui balconi ed altro ancora.

Scalare un orologio delle dimensioni di una stanza ad un prodotto affidabile e portatile pone una serie di problemi, innanzitutto perché in un orologio atomico, gli atomi si accorgono se la frequenza della radiazione elettromagnetica a cui sono esposti è corretta e purtroppo le sorgenti non hanno una frequenza certa e stabile.

A differenza di queste sorgenti di radiazioni, tutti gli atomi di un certo isotopo di una specie (rubidio, cesio, stronzio o qualsiasi altro) sono esattamente identici tra loro e ognuno ha una serie di livelli energetici discreti

che può occupare. Ogni coppia di livelli energetici ha la sua "banda proibita" corrispondente a una frequenza. Se un atomo viene illuminato da radiazioni della frequenza esatta di uno di tali gap, l'atomo assorbirà la radiazione e gli elettroni salteranno al livello energetico superiore. Poco dopo, l'atomo riemetterà la radiazione mentre quegli elettroni torneranno ai livelli energetici naturali. Un rilevatore verifica quanta radiazione gli atomi assorbono (o quanta ne riemettono al rientro nei legami) e segnala se la frequenza in arrivo è troppo alta o troppo bassa. Tale feedback stabilizza la frequenza della sorgente alla frequenza scelta dagli atomi, che è molto precisa e viene inviata a un contatore che può contare le transizioni della radiazione elettromagnetica, costituenti i ticchettii dell'orologio atomico. Il conteggio è una base dei tempi ultra-accurata: in altre parole, un orologio.

Molti fenomeni possono influenzare la precisione dell'orologio: se la fonte si muove, la frequenza della radiazione dal punto di riferimento viene alterata dall'effetto Doppler, facendo sì che atomi diversi selezionino frequenze diverse in base alla loro velocità. Campi elettrici o magnetici esterni, o persino il calore irradiato da un essere umano, possono modificare la frequenza preferita degli atomi. Una vibrazione può far scendere la frequenza di un laser sorgente così tanto che gli atomi smetteranno di rispondere del tutto, interrompendo il ciclo di feedback.

Negli orologi fissi, per ridurre al minimo gli effetti di rumore del calore, si utilizzano più laser per raffreddare gli atomi fino quasi a zero Kelvin.

Per rilevare meglio il segnale, gli atomi si radunano in un reticolo periodico, una trappola a forma di cartone per le uova realizzata da un altro laser. Questa configurazione crea diversi gruppi separati di atomi che possono essere confrontati tra loro per ottenere una misurazione più precisa.

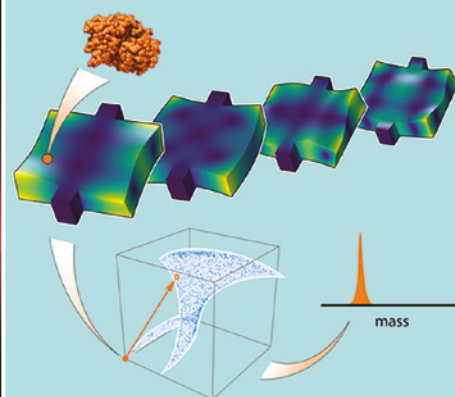
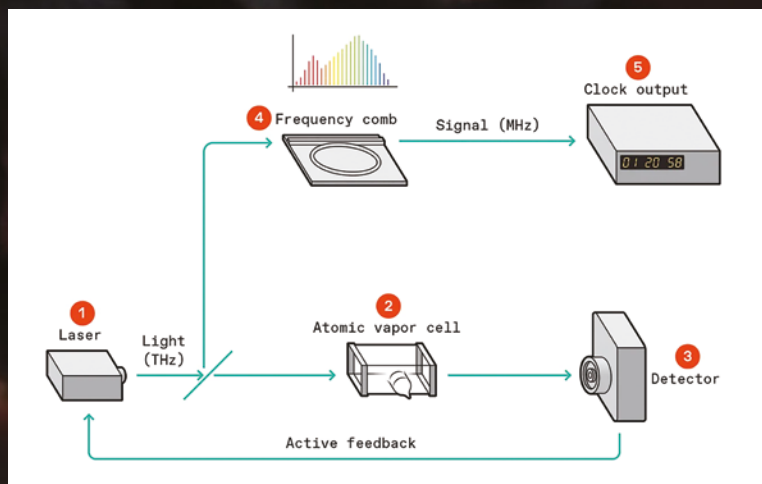
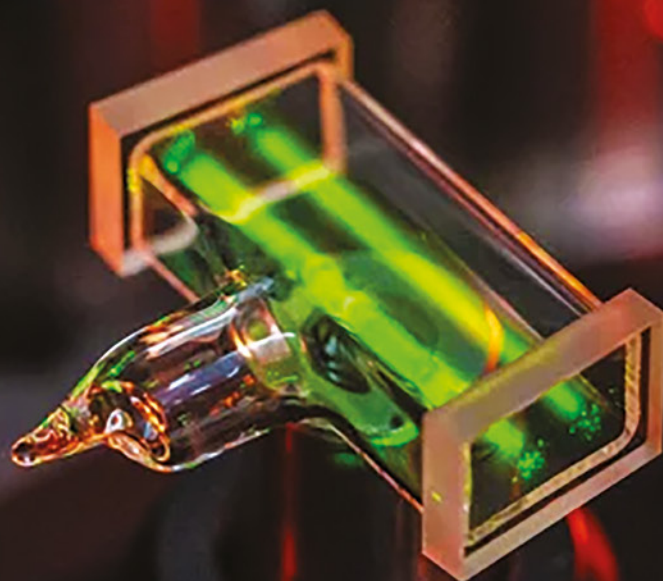
I laser consentono la sorprendente precisione dell'orologio, ma sono costosi e ingombranti, inoltre ciascuno

richiede elementi di controllo ottico. Il laser è un anello debole, perché quando si progetta un oscillatore a microonde, si mette una guida d'onda attorno ad esso e funziona per sempre. Quindi Vector Atomic ha dovuto ripensare l'intera costruzione del suo dispositivo: innanzitutto, ha ridotto il numero di laser, facendo funzionare l'orologio con atomi o molecole nel loro stato gassoso, confinati in una cella di vetro. E senza reticolo periodico per raggruppare gli atomi in gruppi separati e ottenere letture multiple. I laser adottati sono a infrarossi. La molecola usata, caratterizzata da una transizione che potesse essere stimolata da tale laser, è quella di iodio, i cui elettroni hanno una transizione a 532 nanometri. Un solo laser non baste-

rebbe perché nessun dispositivo elettronico è in grado di contare i tick. Per convertire il segnale ottico in microonde mantenendo la precisione del segnale originale, il team ha dovuto incorporare un pettine di frequenza, ossia un laser che emette luce in raf- fiche regolarmente distanziate nel tempo. Siccome i laser sono il fattore limitante, è stata utilizzata una cella di vetro riempita di vapore atomico anziché una camera a vuoto e raffreddamento e intrappolamento laser.

Vector Atomic sta lavorando a versioni ancora più compatte, del volume di appena 5 litri.

vectoratomic.com



La spettrometria di massa rivela il proteoma

Gli scienziati del Caltech hanno sviluppato un metodo basato sull'apprendimento automatico che consente di misurare con precisione la massa di singole particelle e molecole utilizzando complessi dispositivi nanometrici. La nuova tecnica apre la possibilità di utilizzare una varietà di dispositivi per la misurazione della massa e, quindi, l'identificazione delle proteine, e potrebbe aprire la strada alla determinazione della sequenza del proteoma completo (la raccolta di tutte le proteine in un organismo). La spettrometria di massa è uno strumento analitico per realizzare tutti i tipi di investigazione molecolare. Inizia con un campione ignoto ionizzato inviato lungo un percorso specifico; un campo magnetico o elettrico dà agli ioni una spinta laterale per vedere quanto si muovono. Più gli ioni sono leggeri e caricati positivamente, più verranno deviati; questo permette di misurare la massa e la carica di ciascuno ione presente, utile a indagare la composizione chimica del campione, a meno che i campioni non siano grandi come una proteina. Negli ultimi due decenni, con lo sviluppo di dispositivi su scala nanometrica chiamati sistemi nanoelettromeccanici (NEMS) è diventato possibile eseguire una spettrometria di massa che non richiede la ionizzazione del campione, il che consente di superare l'ostacolo. Il dispositivo NEMS è in genere realizzato da una corda di silicio vincolata alle estremità, che se colpito vibra come una corda di chitarra; posizionandovi su un campione, le singole frequenze di vibrazione saranno alterate. Cambiando la posizione e rilevando le frequenze, è possibile determinare un modello che descrive la massa della proteina.

www.caltech.edu

Il fotovoltaico? Lo installa il robot!

EDP ha avviato in Spagna la costruzione di un parco fotovoltaico tramite robot. È la prima volta che una soluzione tecnologica, basata sulla robotica e sull'informatica avanzata, sviluppata da Comau, viene testata in condizioni reali.

Il progetto, denominato AutoPV, prevede l'installazione di 3 MW, su un totale di 122 MW di potenza installata.

Il sistema proposto da Comau utilizza robot per svolgere le attività più pesanti, come la movimentazione di strutture e pannelli FV, mentre gli operatori si dedicano alle attività che richiedono maggiore specializzazione. Oltre a migliorare l'efficienza, l'automazione



eleva la sicurezza sul lavoro. Il progetto pilota consiste nella costruzione di una parte del parco solare FV utilizzando una fabbrica mobile, chiamata Hyperflex, che comprende una stazione di assemblaggio automatico dove viene preassemblata la struttura fotovoltaica e un rover che trasporta e posiziona la struttura nella posizione finale a terra. L'automazione prevede il trasporto e installazione della stazione, l'utilizzo per il montaggio degli inseguitori solari e lo smontaggio per

la missione successiva.

www.comau.com



Il fiore elettronico registra l'attività neuronale

I ricercatori dell'EPFL hanno sviluppato un dispositivo di registrazione neurale chiamato "e-Flower" che avvolge delicatamente gli organoidi in morbidi petali. Gli sferoidi neurali, cluster 3D di cellule cerebrali, stanno rivelandosi strumenti essenziali per comprendere le reti neurali e studiare le malattie neurologiche. L'e-Flower dell'EPFL, un array di microelettrodi 3D (MEA) a forma di fiore, consente ai ricercatori di monitorare l'attività elettrica di questi sferoidi in un modo che in precedenza era impossibile; ciò getta le basi per una ricerca più sofisticata sugli organoidi cerebrali, che sono modelli complessi e miniaturizzati di tessuti cerebrali. L'e-Flower consente di registrare l'attività neurale da una superficie molto più ampia di sferoidi neurali in tempo reale, senza danneggiare i modelli neurali 3D. Lo studio si è concentrato sugli sferoidi neurali, che sono cluster tridimensionali di neuroni che replicano alcune delle funzioni chiave del tessuto cerebrale. Sono più semplici degli organoidi, che contengono più tipi di cellule e imitano più da vicino il cervello. Gli sferoidi sono relativamente facili da produrre e manipolare in laboratorio, il che li rende ideali per i test in fase iniziale, tuttavia, l'obiettivo della ricerca è di applicare l'e-Flower agli organoidi cerebrali, che modellano in modo più accurato lo sviluppo e i disturbi del cervello. L'intento è capire meglio lesioni cerebrali e malattie neurologiche.

actu.epfl.ch

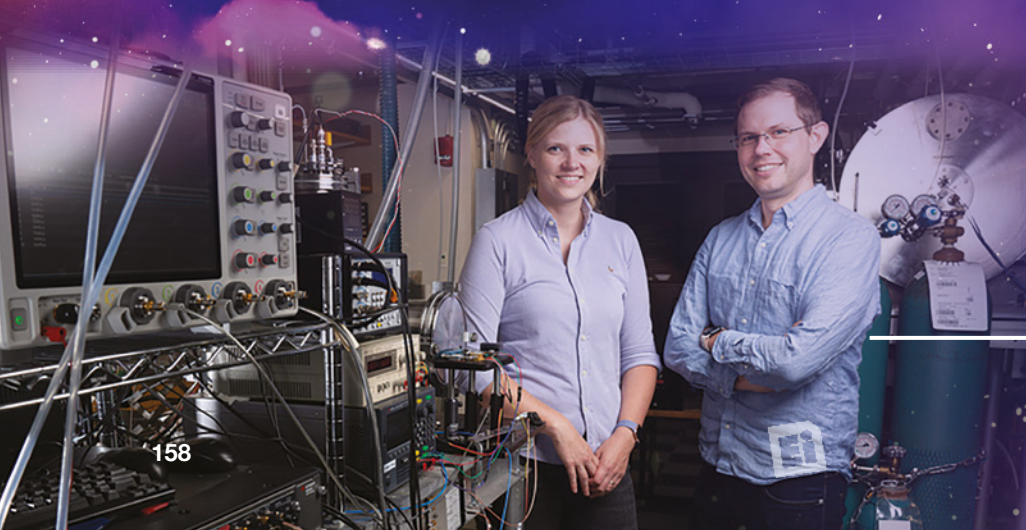
Una nube interstellare potrebbe rilevare l'origine del sistema solare

Ricercatori del MIT hanno scoperto che una nube interstellare contiene un'abbondanza di pirene, una grande molecola contenente carbonio nota come idrocarburo aromatico policiclico (IPA).

Si ritiene che gli IPA, che contengono anelli di atomi di carbonio fusi insieme, immagazzinino dal 10 al 25% del carbonio presente nello Spazio. Il pirene potrebbe essere stato la fonte di gran parte del carbonio nel nostro sistema solare, ipotesi supportata da una recente scoperta secondo cui i campioni restituiti dall'asteroide Ryugu contengono grandi quantità di pirene. Sulla Terra, gli

IPA si presentano come sottoprodotti della combustione di combustibili fossili. A causa della sua simmetria, il pirene è invisibile alle tecniche di radioastronomia finora utilizzate per rilevare quasi tutte le molecole nello Spazio, ma i ricercatori hanno rilevato un isomero del cianopirene (pirene che ha reagito con un cianuro) per romperne la simmetria e permetterne il rilevamento in una nube distante nota come TMC-1, utilizzando il radiotelescopio Green Bank Telescope (GBT) presso il Green Bank Observatory in West Virginia.

www.mit.edu



SERVIZIO DI PROGETTAZIONE

L'area R&D di Futura Group studia, progetta e realizza prototipi per differenti settori utilizzando tecnologie innovative. Il team Futura Group è composto, oltre che dal personale dei nostri laboratori tecnici, anche da un numero elevato di collaboratori esterni, ciascuno specializzato nel proprio settore.



SERVIZIO DI REALIZZAZIONE PCB

Grazie al Servizio di realizzazione PCB potrai ricevere i tuoi circuiti stampati semplicemente caricando sul nostro sito il tuo Gerber e scegliendo le dimensioni e le caratteristiche preferite. È possibile ordinare schede a una singola faccia oppure multistrato, scegliendo anche il colore del solder.



Trasformiamo le tue idee in realtà!

FUTURA ELETTRONICA È IL TUO PARTNER DI RIFERIMENTO PER MOLTEPLICI SERVIZI. I NOSTRI TECNICI SONO A TUA DISPOSIZIONE PER FORNIRE TUTTE LE INFORMAZIONI E UNA VALUTAZIONE PERSONALIZZATA IN BASE ALLE TUE NECESSITÀ.

SERVIZIO TAGLIO LASER

Futura Elettronica offre la possibilità di realizzare mediante taglio laser i tuoi progetti in plexiglass (o metacrilato) con spessore massimo di 10 mm e con dimensioni massime lavorabili di 130x90 cm. Carica sul nostro sito il tuo file .DXF, scegli le dimensioni e le caratteristiche preferite.



SERVIZIO STAMPA 3D

Futura Elettronica mette a disposizione le proprie stampanti 3D FDM (fused deposition modeling), per chi intende realizzare un oggetto tridimensionale con dimensioni fino a 40x40x40 cm partendo da un modello 3D digitale. Carica sul nostro sito il tuo file .STL, scegli le dimensioni e le caratteristiche preferite.



**FUTURA
ELETTRONICA®**

www.futurashop.it

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Affidabilità e precisione!

Generatore di segnali con frequenzimetro 2 CH - 60 MHz



cod. JT-JDS6600B

€ 178,00

VERSIONE

**JOY-IT & FUTURA
ELETTRONICA**

CON MANUALE D'USO IN ITALIANO

- Canali: 2
- Frequenza max. di uscita: 60 MHz
- Frequenza di campionamento: 100 MSa/s
- Forme d'onda standard: sinusoidale, quadra, rampa, triangolare, impulso, noise, arbitraria, impulso di Lorenz, aumento/diminuzione esponenziale, CMOS
- Interfaccia: USB
- Display: 2.4" TFT LCD colori

Accessori inclusi



Prezzo IVA inclusa.

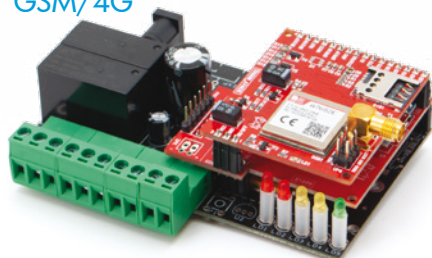
Gestisci e controlla la tua casa con i TELECONTROLLI GSM

Attiva e controlla da remoto qualsiasi dispositivo elettrico o elettronico: dalla centralina antifurto all'impianto di irrigazione, dalle luci di casa al sistema di condizionamento e raffrescamento. Configurazione via SMS/PC con predisposizione per gruppo di continuità.

4G
LTE



COMBINATORE TELEFONICO GSM/4G

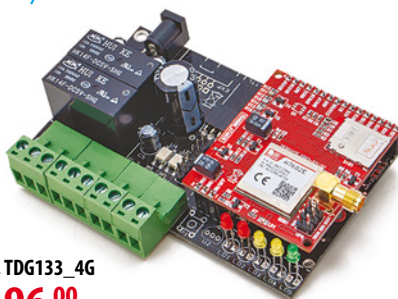


cod. TDG135MP3_4G
€ 125,00

- 2 Ingressi e 2 Uscite
- Menu vocale in Italiano
- Messaggi di allarme vocali
- Invio segnalazioni di allarme ingressi

Disponibile anche in versione 2G cod. TDG135MP3 • Euro 99,00

TELECONTROLLO BICANALE GSM/4G

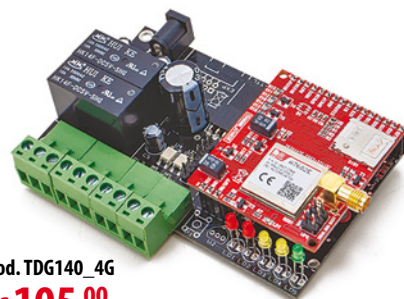


cod. TDG133_4G
€ 96,00

- 2 Ingressi e 2 Uscite
- Funzione Apricancello
- Invio segnalazioni di allarme ingressi

Disponibile anche in versione 2G cod. TDG133 • Euro 79,00

TELECONTROLLO GSM/4G CON COMANDI DTMF

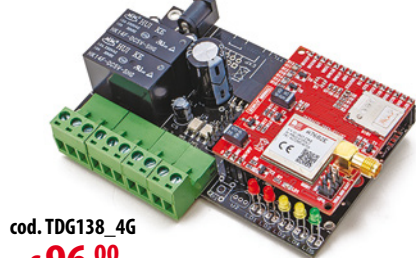


cod. TDG140_4G
€ 105,00

- 2 Ingressi e 2 Uscite
- Menu gestibile via DTMF
- Invio segnalazioni di allarme ingressi

Disponibile anche in versione 2G cod. TDG140 • Euro 88,00

TELEALLARME GSM/4G CON ANTI JAMMER

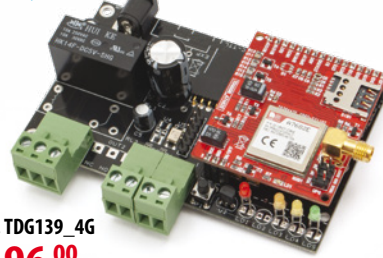


cod. TDG138_4G
€ 96,00

- 2 Ingressi e 2 Uscite
- Funzione anti Jammer o Telecontrollo
- Invio segnalazioni di allarme ingressi e Jammer

Disponibile anche in versione 2G cod. TDG138 • Euro 79,00

TERMOSTATO CON CONTROLLO GSM/4G



cod. TDG139_4G
€ 96,00

- Controllo riscaldamento
- Manuale/Automatich/Asservito Termostato
- Allarmi temperatura configurabili
- Invio segnalazioni di allarme temperatura e blocco caldaia

Disponibile anche in versione 2G cod. TDG139 • Euro 79,00

APRICANCELO GSM/4G

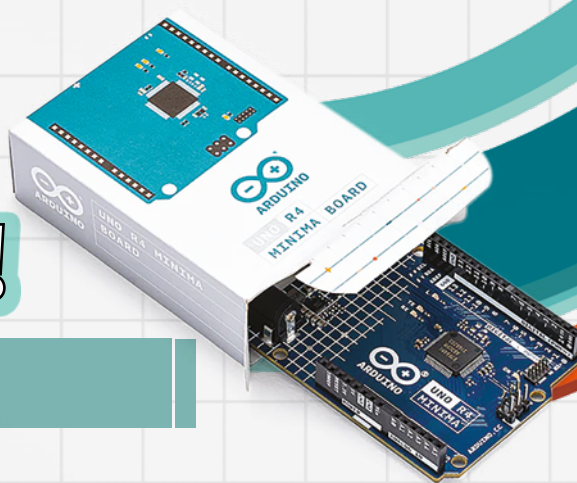


cod. TDG134_4G
€ 92,00

- 1 Ingresso
- Funzione Apricancello

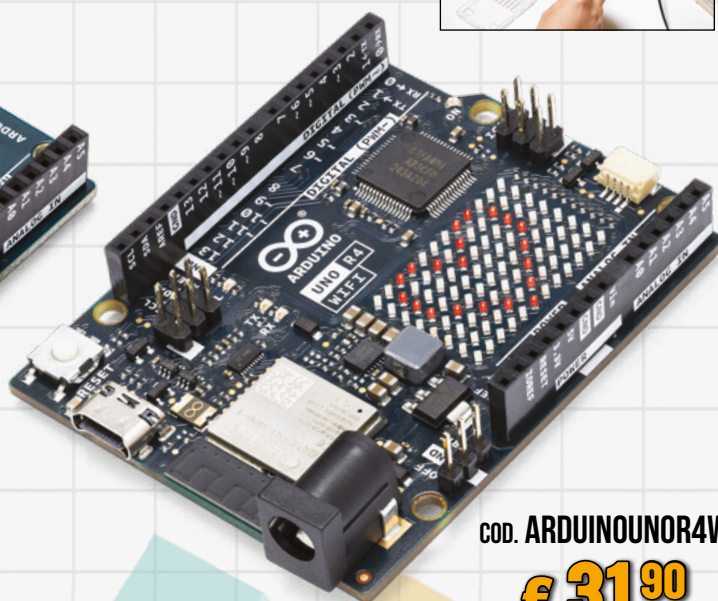
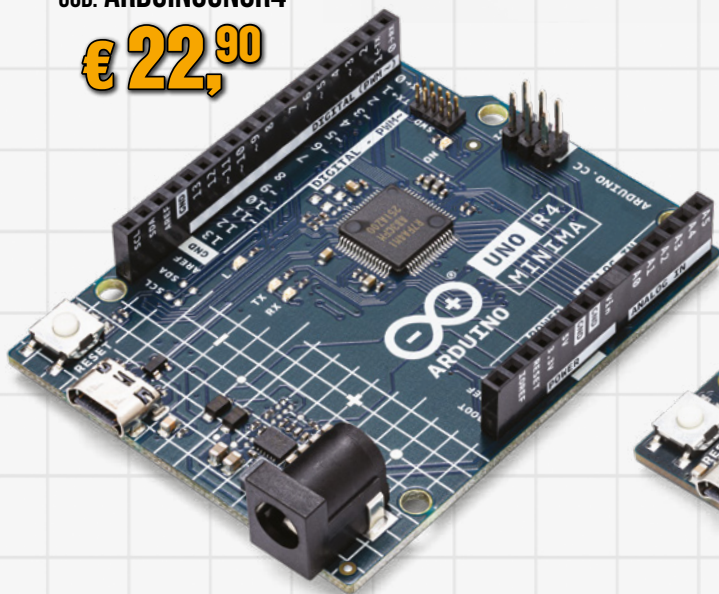
Disponibile anche in versione 2G cod. TDG134 • Euro 75,00

Scopri Arduino UNO R4!!!



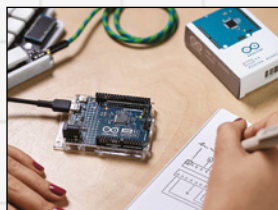
COD. ARDUINOUNOR4

€ 22,90



COD. ARDUINOUNOR4W

€ 31,90



Prezzi IVA inclusa.

CARATTERISTICHE TECNICHE		UNO R4 MINIMA	UNO R4 WiFi
MCU	R7FA4M1AB3CFM#AA0 (Arm cortex M4)	✓	✓
Memoria	256KB Flash / 32KB SRAM	✓	✓
Tensione operativa	5V	✓	✓
Tensione d'ingresso	6-24V	✓	✓
Velocità Clock	48 MHz	✓	✓
Porta di programmazione	USB-C	✓	✓
WiFi/BLE	ESP32-S3-MINI		✓
LED Matrix	12x8 (96 LED rossi)		✓
Connessioni supplementari	-connettore Qwiic -pin OFF -pin VRTC		✓
Pin digitali	14	✓	✓
PWM	6	✓	✓
ADC	6	✓	✓
DAC	1 (12 bit)	✓	✓
SPI	1	✓	✓
I2C	1	✓	✓
CAN	1	✓	✓