

Elettronica In

www.elettronica.in.it

€ 6,00 - Anno XXVII - n. 260 - DICEMBRE 2021 / GENNAIO 2022

- Parliamo con Raspberry Pi
- Plant Smart Monitor
- Smart Switch
- Animazioni natalizie su pannelli LED

Computer vision LED mapping



Juke-box MP3



Arduino current shield

CONOSCERE
E USARE



Node-RED

Costruisci un vero robot!



Mini robot da combattimento controllato da App

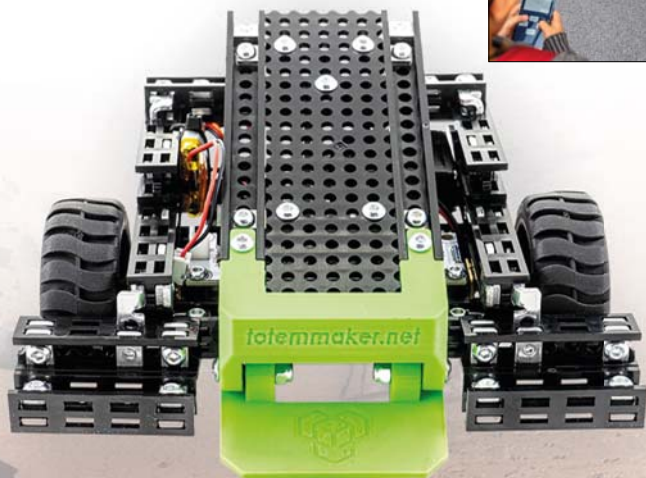
Minitrooper è un kit "Totem STEAM Toys" che permette di realizzare un piccolo robot da combattimento con catapulta e controllabile attraverso App (Totemmaker) per smartphone. Minitrooper può essere impiegato in vari giochi come: sumo, football, corse, battaglie tra robot, ecc. La catapulta può essere utilizzata per ribaltare gli avversari in piccoli combattimenti tra due o più robot, oppure per spingere una pallina nel gioco del football, ecc.

€ 79,⁵⁰

Cod. MINITROOPER



**CONTROLLABILI
TRAMITE
APP!**



Rover con braccio robotico e pinza controllato da App

Gripper Bot è un kit "Totem STEAM Toys" che permette di realizzare un piccolo rover con braccio robotico e pinza, controllabile attraverso App (Totemmaker) per smartphone. La pinza, montata sul braccio robotico, consente di afferrare e sollevare piccoli oggetti (max. 60 grammi).



€ 69,⁰⁰

Cod. GRIPPERBOT

Assemblando i robot imparerai:

- > Nozioni base di meccanica, elettricità e programmazione.
- > Soluzioni ingegneristiche semplici.

- > Utilizzo di diverse parti meccaniche e strumenti.
- > Utilizzo di parti meccaniche in diversi modelli/situazioni.

Prezzi IVA inclusa

**FUTURA
ELETTRONICA®**

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche
e vendita on-line su:

www.futurashop.it



Tra sogno e realtà

Siamo alla fine del 2021, l'anno in cui speravamo di lasciarci alle spalle il Covid 19. L'anno in cui avremmo dovuto prendere grandi decisioni per risolvere il problema del riscaldamento globale. Purtroppo nessuno di questi obiettivi è stato raggiunto, perché i problemi alla radice sono tutt'altro che facilmente risolvibili. Sul fronte Covid, c'è molto da fare per capire una malattia della quale si conosce ancora poco, ma si spera che tra campagne vaccinali (che si arricchiranno di nuovi e più efficienti preparati) e adozione di cure efficaci (non solo farmaci collaudati ma anche i promettenti antivirali Merck e Pfizer) le cose migliorino; anche se è inutile fingere di non capire che nessuna vaccinazione

Se le soluzioni per la pandemia sembrano esserci, non si può dire lo stesso per i cambiamenti climatici.

funzionerà fin quando non si riuscirà a raggiungere quella parte di mondo povero che costantemente bussa ai nostri confini.

Ma se le soluzioni per la pandemia sembrano esserci, non si può dire lo stesso per i cambiamenti climatici, di cui si è accennato al G20 di Roma e si è discusso (perché quello era il tema dominante) al COP26 di Glasgow, partito con pochi e semplici obiettivi, ritenuti tuttavia procrastinabili o inaccettabili dai Paesi economicamente e industrialmente emergenti, preoccupati che la crescita sostenibile possa bloccare il loro sviluppo e da quelli che politicamente si contrappongono al mondo occidentale, in un braccio di ferro che dura da decenni. Del resto i politici sanno di dover rispondere politicamente ora di qualcosa i cui benefici non vedranno, se non con gli occhi di quei giovani che chiedono loro un gesto che richiede un coraggio, che -come diceva il Manzoni- se uno non ce l'ha, mica se lo può dare. Ma è ovvio che nelle strade si fanno sogni, mentre nei Palazzi si fa politica. La tutela dell'ambiente passa comunque anche dai singoli e da comportamenti virtuosi, come la cura del verde in cui può aiutare il Plant Smart Monitor basato su micro:bit e l'adozione di soluzioni tecniche a basso consumo come il Computer Vision NeoPixel Mapping, che allietterà le festività di fine d'anno, in attesa che il 2022 ci porti buone notizie!



In copertina:
Accendiamo le nostre luminarie utilizzando degli effetti spettacolari grazie alla Computer Vision e i LED neopixel.


Boris Landoni
✉ boris.landoni@elettronica.in

ON LINE

✉ elettronica.in

SI RINGRAZIA

Futura Elettronica
Mouser
Texas Instruments

Elettronica In

Rivista mensile, anno XXVII n. 260
DICEMBRE 2021 / GENNAIO 2022

EDITORIALE

Direttore Responsabile

Boris Landoni

✉ boris.landoni@elettronica.in

Redazione

Stefano Garavaglia, Paolo Gaspari,
Davide Scullino, Gabriele Daghetta

✉ redazione@elettronica.in

Grafica

Alessia Sfulcini

✉ alessia.sfulcini@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 10 numeri Euro 45,00

Annuo 10 numeri digitale Euro 39,00

Le richieste di abbonamento vanno inviate a:

FUTURA GROUP srl

via Adige 11, 21013 Gallarate (VA)

✉ Tel. 0331-752668

oppure tramite il sito

<https://www.elettronica.in/abbonamenti/>

Distribuzione per l'Italia

SO.DI.P. Angelo Patuzzi S.p.A.

via Bettola 18 - 20092 Cinisello Balsamo (MI)

✉ Tel. 02-660301 ✉ Fax 02-66030320

Stampa

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

Tiratura
18.000 copie



Mensile
associato all'USPI,
Unione Stampa
Periodica Italiana

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

Prezzo di copertina Euro 6,00.

Gli arretrati nei formati cartaceo e digitale (pdf) sono acquistabili
sul sito della rivista al prezzo di Euro 6,00.

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, il firmware ed il 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 (www.elettronica.in/ase.pdf).
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.

© 2021 Futura Group srl

Futura
Group
Edizioni

Contenuti

01

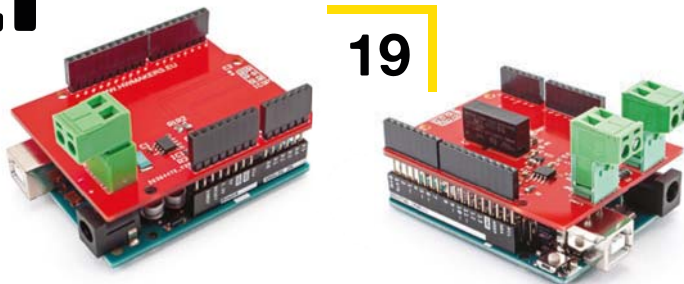
RUBRICHE

- 01 Editoriale
- 05 T&T Tips & Tricks
- 09 Embedded IoT
Digital Transformation of Things
- 13 Componenti & Sistemi
- 126 Scienza & Tecnologia

Arduino

Arduino current shield

19



19

ARTICOLI

ARDUINO

19 Arduino current shield

Il circuito integrato INA219 della Texas Instruments, permette la realizzazione di sistemi di misura della corrente continua in modalità shunt anche con Arduino; eccovi due shield, uno galvanicamente isolato e uno non, per sperimentare

RASPBERRY PI

37 Animazioni natalizie su pannelli LED

Creiamo un display luminoso con dei pannelli a matrice di LED, un Raspberry Pi e alcuni software per animare il nostro Natale.

Progetto di Copertina

COMPUTER VISION NEOPIXEL MAPPING

78



Raspberry Pi

Animazioni natalizie su pannelli LED



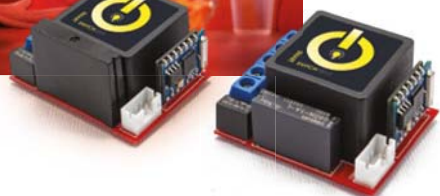
37

Networking

Smart Switch



47



Raspberry Pi

Parliamo con Raspberry Pi



107

Audio

Un Juke-box MP3



63

Programmazione

Plant Smart Monitor



93

Didattica

Conoscere e usare Node-RED



117

NETWORKING

47 Smart Switch

Interruttore IoT controllabile da remoto con App o comandi vocali utilizzando l'assistente vocale Amazon Alexa oppure Google Assistant e configurabile da pagina web. Ne proponiamo due versioni: una con relé e una con SSR.

AUDIO

63 Un Juke-box MP3

Realizziamo un Juke-box in chiave moderna, basato sul modulo riproduttore MP3 DFR0299 e gestito mediante identificazione RFID.

TOP PROJECT

78 Computer vision NeoPixel mapping

Rileviamo, tramite un algoritmo di computer vision, la posizione dei NeoPixel posti in ordine sparso e trasformiamoli in una matrice di LED virtuale cui applicheremo effetti luminosi.

PROGRAMMAZIONE

93 Plant Smart Monitor

Utilizziamo la flessibilità e potenza di micro:bit per realizzare un sistema di irrigazione intelligente delle piante, portatile ed interconnesso.

RASPBERRY PI

107 Parliamo con Raspberry Pi

Interagiamo verbalmente con la popolare scheda Raspberry Pi attraverso un microfono e l'Intelligenza Artificiale.

DIDATTICA

117 Conoscere e usare Node-RED

Proseguiamo lo studio del tool di flow-based programming orientato all'IoT ed alla connettività. In questa terza puntata analizziamo la gestione delle periferiche della Raspberry Pi tramite Node-RED, approfittandone per l'implementazione di nuovi esempi pratici. Terza puntata.

Gli strumenti di sviluppo in un unico posto

Migliaia di strumenti da centinaia
di produttori fidati



Scegli tra i prodotti della
nostra vasta selezione su
mouser.it/dev-tools



MOUSER
ELECTRONICS

Trailing edge e leading edge

Nel controllo variabile della potenza di motori e in generale di utilizzatori in corrente alternata, si opera la parzializzazione d'onda: con questo termine s'intende che per modulare la potenza si va a ridurre la durata di erogazione della corrente in ogni singola semionda, interrompendola prima della sua naturale conclusione ovvero si fa partire ogni semionda con un certo ritardo.

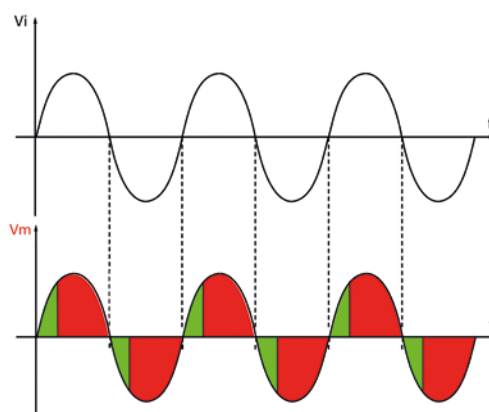
La parzializzazione d'onda si realizza quindi interdicendo il TRIAC utilizzato per alimentare il carico per una porzione del semiperiodo della corrente alternata.

Quindi nel caso di un motore elettrico, per avere il motore fermo non si fornisce tensione, mentre per aumentare la potenza si parte con il ritardo massimo e gradualmente si fa partire l'impulso sinusoidale sempre prima, fino ad avere sul motore la sinusoide completa, ovvero si taglia una porzione di semionde via-via decrescente.

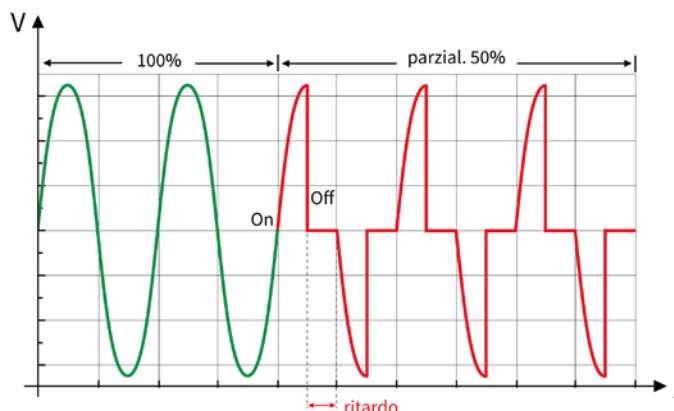
In ogni caso, conviene sincronizzare la conduzione con il passaggio per lo zero volt attraverso un circuito chiamato Zero Crossing Detector (ZCD), necessario perché altrimenti può capitare che l'accensione del TRIAC avvenga in corrispondenza del picco della sinusoide, con due conseguenze: l'applicazione brusca della massima corrente (e relativa sollecitazione eccessiva dei TRIAC o SCR) e la generazione di intensi impulsi di disturbo radioelettrico. Lo ZCD rileva il passaggio per lo zero e fa in modo che l'impulso di accensione ritardata dell'interruttore statico parta dopo e non in corrispondenza della cresta dell'onda sinusoidale.

Esistono due tecniche per implementare la parzializzazione d'onda in un circuito a corrente alternata: la tecnica leading-edge e la tecnica trailing-edge. Nella prima delle due modalità

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.



← **Controllo in alternata mediante parzializzazione d'onda leading-edge:** le parti rosse corrispondono alla conduzione (motore riceve corrente) e le verdi all'interdizione. Il tempo trascorso dal passaggio per lo zero (Zc) alla zona rossa determina la potenza al motore.



↑ **Parzializzazione al 50% con tecnica trailing-edge su una tensione alternata sinusoidale.**

la parzializzazione consiste nel ritardare l'entrata in conduzione dell'interruttore statico rispetto al momento in cui andrebbe naturalmente in conduzione: nel caso del circuito a TRIAC ciò avviene in entrambe le semionde e invece per i circuiti a SCR nella sola semionda positiva. Il ritardo è determinato rispetto al punto di passaggio per lo zero volt della sinusoide (Z_c = Zero Crossing) e si ottiene, nella forma più semplice, pilotando il gate del TRIAC o SCR attraverso la stessa tensione sinusoidale che alimenta l'utilizzatore, però attenuata in modo che la tensione corrispondente a quella di accensione dell'interruttore statico si raggiunga con il ritardo desiderato. Il relativo circuito è, quindi, piuttosto semplice, ma per contro presenta il limite di innescare disturbi da commutazione (spike) dovuti alla brusca accensione del tiristore e repentina applicazione della corrente al carico, che divengono tanto più marcati quanto più l'attivazione del TRIAC o SCR avviene vicino al picco della semionda.

La parzializzazione migliore sotto l'aspetto dei disturbi e della sollecitazione degli interruttori statici si ottiene adottando la modalità trailing-edge, ossia interrompendo la conduzione quando è trascorso un certo tempo dal picco, tempo che è dell'ordine di una frazione più o meno grande della durata della semionda della sinusoide.

La tecnica trailing-edge consente uno spegnimento morbido e siccome opera sull'interruzione della corrente e non sull'applicazione, consente di fornire al carico impulsi dalla partenza morbida, caratteristica della sinusoide tale, quindi, da limitare armoniche e pericolosi picchi di corrente.

Il principale difetto del parzializzatore trailing-edge è la maggiore complessità circuitale, perché il circuito deve attendere il picco e ritardare rispetto a questo, ma prima dello zero-crossing, l'accensione dell'interruttore controllato. Le immagini in queste pagine propongono i grafici che descrivono la parzializzazione d'onda leading-edge e la trailing-edge.

DVD: la tecnica Blue Ray

I CD-ROM e i DVD ROM, come anche i masterizzatori sempre di CD-ROM e DVD, funzionano sulla base della riflessione di un raggio laser da parte di "pozzetti" che si trovano sequenzialmente sulla superficie dei dischi. Tutti questi apparecchi sono costituiti

da una meccanica ed un'elettronica, più un'ottica: la meccanica fa girare il disco e lo mantiene in posizione e inoltre provvede a posizionare l'ottica in modo da leggere o scrivere; l'ottica comprende un diodo laser e un fotodiode: il primo proietta un raggio di luce all'infrarosso verso la superficie del disco e il secondo rileva il riflesso; l'inclinazione con cui questo riflesso investe il fotodiode dipende dalla profondità dei singoli "pozzetti" scavati in scrittura dal dispositivo scrittore. La profondità dello stato logico 0 è diversa da quella dell'1, il che permette al lettore di discernere i due livelli; il riconoscimento si deve al fatto che il raggio riflesso si presenta con un'intensità diversa in base all'angolazione con cui giunge sul fotodiode. Questo vale in lettura. Il masterizzatore scrive con un procedimento simile: il laser, più potente di quando non è in lettura, depolimerizza un composto situato sotto lo strato protettivo del disco, scavando ogni pozzetto corrispondente all'1 logico; il fotodiode rileva il riflesso per verificare la corretta scrittura. L'ultimo ritrovato in tema di tecnologia del disco ottico è quella del DVD Blue Ray, cosiddetto perché utilizza un laser a luce blu invece dell'infrarosso; il motivo di ciò deriva dall'esigenza di aumentare la quantità di dati scrivibili in un DVD, cosa fattibile solo - a parità di diametro del disco e di strati (2) disponibili -

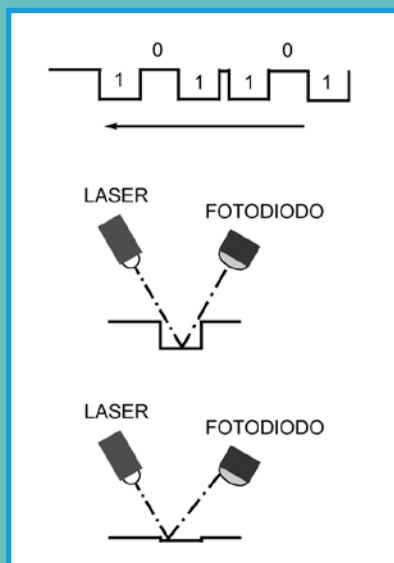
li- riducendo le dimensioni dei pozzetti costituenti i singoli bit. Siccome la dimensione dei DVD double-layer aveva raggiunto la lunghezza d'onda del laser e più di questa non poteva scegliere, si è pensato di ricorrere a un laser blu, la cui lunghezza d'onda è circa 1/3 di quella dell'infrarosso, perciò anche le dimensioni dei singoli pozzetti possono essere ridotti in proporzione, incrementando la densità per unità di superficie e quindi la quantità di dati immagazzinabili, che supera i 9,4 gigabyte. Nel lettore di dischi ottici, un raggio laser colpisce la superficie del disco dalla quale viene riflesso con un'angolazione dipendente dalla profondità del solco che incontra; l'inclinazione con cui il raggio colpisce il fotodiode sensore determina una diversa tensione e quindi lo logico zero o 1 del bit corrispondente.

DDR5: l'ultima frontiera delle memorie

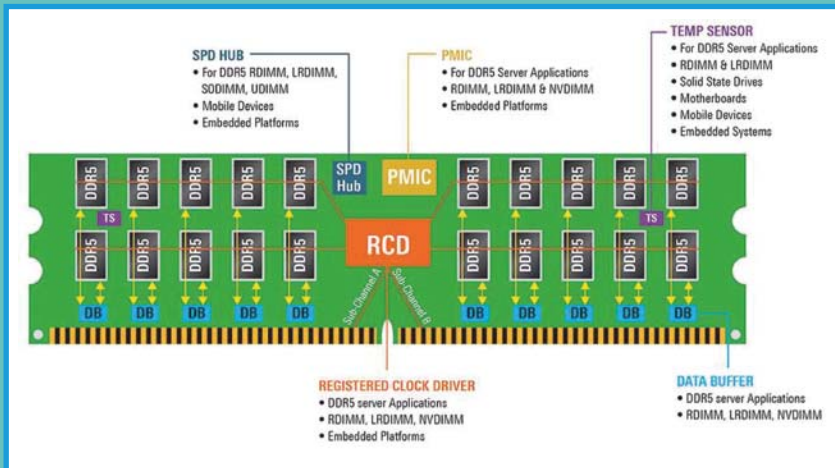
L'avanzare della tecnologia dei computer richiede sempre più capacità di memoria, velocità e prestazioni più elevate dalle CPU, perché le applicazioni continuano a necessitare di maggiore larghezza di banda, capacità e prestazioni, spinte da fattori come l'intelligenza artificiale, il Machine Learning e l'analisi dei Big Data. Per inviare i dati alla CPU nel minor tempo possibile servono memorie DDR (DRAM) in grado di fornire dati con accesso a latenza sempre più bassa. Anche le CPU diventano sempre più veloci, quindi è necessaria una larghezza di banda maggiore per spostare rapidamente i dati dai core che li utilizzano e la DRAM. Anche la capacità della DRAM è fondamentale nei sistemi informatici, per garantire che i dati necessari vi possano essere archiviati, giacché maggiore è l'ammontare della RAM, più è probabile che il sistema operativo sia in grado di mantenere in memoria i successivi dati necessari, invece di spostarli in uno spazio di archiviazione più lento.

Negli ultimi decenni, l'ottimizzazione dei processi di fabbricazione dei semiconduttori ha portato a una crescita costante della densità dei chip DRAM, tanto che la DDR4 è attualmente in grado di raggiungere fino a 16 Gb per die. Ma siccome anche questo non basta, è nata la DDR5, ultima frontiera in fatto di DRAM, che potrà spingersi fino a densità più elevate ed a larghezze di banda fino a 6.400 Mbps.

La DDR5 introduce nuove funzionalità



↑ Nel lettore di dischi ottici, un raggio laser colpisce la superficie del disco dalla quale viene riflesso con un'angolazione dipendente dalla profondità del solco che incontra; l'inclinazione con cui il raggio colpisce il fotodiode sensore determina una diversa tensione e quindi lo logico zero o 1 del bit corrispondente.



↑ Tipica DIMM DDR5 per server con dual sub-channel.

che consentiranno ai sistemi di prossima generazione di progredire e soddisfare le future esigenze di elaborazione.

Ma cosa c'è di nuovo nella DDR5?

Innanzitutto un raddoppio della velocità del canale, perché il transfer-rate della memoria DDR5 raggiungerà la velocità del canale di 6.400 MT/s con una velocità iniziale di 4.800 MT/s. La velocità massima della più performante DDR4 è oggi 3.200 MT/s. Poi la DDR5 permetterà densità di memoria dei moduli DIMM più elevata, giacché un singolo package DRAM può avere fino a 16 Gb, consentendo un massimo di 128 Gb su una stecca DIMM ad alta densità.

La DDR5 può attualmente consentire DIMM fino a 512 GB su un singolo modulo, utilizzando un componente DRAM 8-High Stack, basato su 16 Gb in una configurazione 8Rx4.

Poi, a favore della DDR5 gioca il minor consumo di energia dovuto alla riduzione della tensione operativa VDD/VDDQ/ VPP ad 1,1 V contro gli 1,2 V delle DDR4; un piccolo scarto, che però fa la

differenza quando i banchi di memoria diventano molti. Inoltre un regolatore di tensione su DIMM consente una gestione dell'alimentazione più efficiente a livello DIMM ed un sensore di temperatura su DIMM fornisce monitoraggio e controllo termico.

La larghezza del bus DQ è aumentata da 72 bit a 80 bit e c'è un buffer di dati aggiuntivo sulle DDR5 LRDIMM rispetto alle DDR4 LRDIMM e il supporto DFE è stato aggiunto sia all'RCD che alla DRAM. Infine, nella DDR5 è previsto il supporto del protocollo bus in banda laterale I²C. Tra le altre caratteristiche, ci sono due sotto-canali a 40 bit per singolo modulo DIMM.

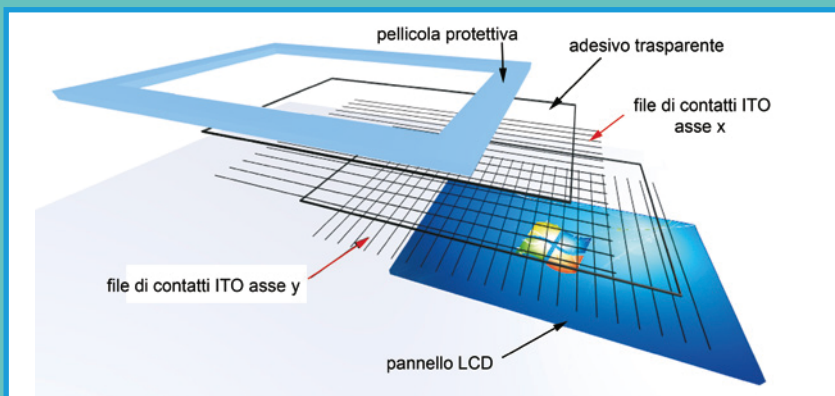
Il touch-screen capacitivo

Esistono sostanzialmente due tecnologie per realizzare i touch-screen

capacitivi: la Surface Capacitive (SCT) e la Projected Capacitive (PCT); nel primo caso vengono disposti ai quattro angoli del pannello tattile altrettanti elettrodi alimentati da una tensione (aventi in serie ognuno una resistenza per limitare la corrente di fuga che scorre nel corpo della persona che tocca lo schermo nel momento in cui viene sottratta la carica elettrica) e una sottile pellicola uniformemente distribuita su tutta la superficie. Quando si tocca con un dito lo schermo, avviene un trasferimento di carica che determina un assorbimento da ciascuno dei quattro elettrodi, che è uguale solo se il dito è all'incrocio delle due diagonali (al centro) altrimenti è differente tra i quattro; ai capi di ciascuna resistenza di limitazione viene misurata la caduta di tensione dovuta alla corrente assorbita dal dito e sulla base della proporzione tra le quattro cadute, un processore identifica la posizione del dito sullo schermo.

Quanto alla PCT, l'armatura interna non è unica ma composta da una griglia di righe e colonne con un chip separato per il rilevamento su riga e colonna; quando un dito tocca lo schermo, una piccola carica elettrica viene trasferita ad esso creando una caduta di tensione su tale punto dello schermo. Gli schermi tattili capacitivi, un tempo limitati al rilevamento di un dito soltanto, oggi possono rilevare 10 contatti simultanei, per agevolare, ad esempio, la scrittura veloce su tastiera virtuale a schermo; un tempo la difficoltà nel rilevare la posizione se toccati in più punti era dovuta al fatto che il decoder rilevava il contatto in senso assoluto. Oggi, con l'affinamento delle tecnologie (PCT) si possono rilevare più posizioni scansionando opportunamente la matrice.

↓ Schematizzazione del touch panel in tecnologia PCT.



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

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.

€ 13,90

Cod. UL13-12



12V-1,3AH

€ 24,90

Cod. UL7-12

Ministilo

6V-1,3AH



€ 7,90

Cod. UL13-6

6V-3,4AH

€ 11,90

Cod. UL34-6

Prezzi IVA inclusa.

Batterie ricaricabili al Ni-MH VARTA "Ready To Use"



€ 12,00

Cod. HR03AAA800



€ 13,00

Cod. HR6AA2100



€ 13,80

Cod. HR14C3000

SHV Energy e il consorzio Sigfox stanno implementando una soluzione globale per il monitoraggio del livello dei serbatoi di stoccaggio GPL; iniziato in Francia, Belgio e Germania, presto il progetto si espanderà con 50.000 unità in tutto il mondo.

La capacità di monitoraggio delle risorse IoT (Internet of Things) di Sigfox consente a SHV Energy di ridurre il costo del monitoraggio del livello dei serbatoi, aumentando al contempo l'efficienza e la produttività.

La raccolta dei dati è fondamentale per le attività che implicano lo stoccaggio e la gestione delle fonti energetiche e la rete Sigfox permette di raccogliere informazioni ad un costo estremamente basso, il che con-

› Monitorare a distanza i serbatoi di gas



sente l'estensione su ampia scala di un sistema di monitoraggio dei serbatoi, che sia omogeneo in tutto il mondo.

SHV Energy ha istituito un centro di eccellenza

per la telemetria e ha utilizzato le soluzioni di Sigfox e il partner Aton SB S.p.A per fornire una soluzione omogenea e globale per il monitoraggio del livello del serbatoio che fornisce analisi sofisticate dei dati e dispositivi intelligenti collegati per le letture giornaliere del livello del serbatoio. SHV Energy può ora ottimizzare i percorsi e gli orari di consegna e ha migliorato la sostenibilità riducendo l'impronta di carbonio dell'azienda.

Inoltre, la soluzione basata su Sigfox e la sua rete globale Sigfox OG è notevolmente più conveniente rispetto alle alternative cellulari.

I risparmi derivano da componenti a basso costo, assenza di schede SIM e costi di connettività modesti.

<https://www.sigfox.com>

› Astro: l'Alexa mobile di Amazon

Amazon ha realizzato un robot su tre ruote per la casa che ha chiamato Astro, dotato di telecamere, uno schermo da 10 pollici e due vani portabicchieri, integra e porta in giro l'assistente vocale Alexa; possiamo considerarlo un maggiordomo robotico in grado di muoversi autonomamente all'interno della casa, navigare per effettuare il check-in in aree specifiche, mostrare una vista live dei luoghi domestici tramite l'app Astro e persino fare da guardiano, inviando notifiche di allarme se rileva rumori sospetti, una concentrazione eccessiva di CO o una persona estranea attraverso un periscopio gestito dall'app, che emerge dal corpo e riprende l'ambiente circostante.

Tra le varie funzioni, Astro permette anche di stabilire da quali aree inviare riprese, per adeguarsi alle norme sulla privacy e se disabilitare audio, video e mobilità.

La killer application iniziale di Astro potrebbe chiamarsi Disney resort,



perché Amazon e Disney hanno firmato un accordo che punta a distribuire il robot negli hotel, sia per le sue funzioni di assistente Alexa mobile, sia per realizzare l'interazione a pagamento di Astro con i personaggi della Disney, azienda che punta ad arricchire la sua offerta di parchi tematici con forme di interattività.

Oltre alle funzioni interattive avanzate appena descritte, il robot di Amazon può portare con sé (e qui si parla di portare fisicamente) fino a 2 kg di peso, quindi più del peso dei bicchieri che stanno nel suo portabicchieri. Il robot Astro sarà disponibile al prezzo di lancio di 1.000 \$.

www.amazon.com

› Disponibile il servizio 5G gestito di Sierra Wireless

Sierra Wireless, fornitore di soluzioni IoT, annuncia la disponibilità del suo servizio di rete gestita 5G negli Stati Uniti, offrendo l'elevatissima velocità e la ridottissima latenza della connettività wireless cellulare 5G per le applicazioni business-critical. I servizi di rete completamente gestiti di Sierra Wireless forniscono soluzioni a banda larga end-to-end per connettività permanente, temporanea e di backup in applicazioni retail, food & beverage, sanità, segnaletica digitale, chioschi/POS ed e-learning. Queste soluzioni offrono una connettività Internet semplice, conveniente e affidabile, senza costi di acquisto delle apparecchiature, tempi di attività garantiti e gestione senza interruzioni. Grazie alla gestione e alla creazione di report basati

su cloud e al pooling tra operatori che garantiscono la migliore copertura di rete in ogni momento, il servizio di rete 5G di Sierra Wireless può essere utilizzato con i router 5G della serie XR di Sierra Wireless. Con il loro esclusivo accordo sul livello di servizio, i servizi di rete gestiti di Sierra Wireless assicurano funzioni business-critical con connettività garantita e monitoraggio costante delle prestazioni di rete gestiti 24/7/365 dal team di professionisti wireless di Sierra. I servizi includono anche l'uso dell'hardware e del software Sierra Wireless, con aggiornamenti per la sicurezza e il firmware, nonché un collegamento di gestione fuori banda sempre attivo al servizio di gestione AirLink (ALMS). Questa capacità unica garantisce il

› Intelligenza artificiale per preservare le foreste pluviali

Infineon Technologies, leader mondiale nei semiconduttori per la mobilità, l'efficienza energetica e l'IoT, ha annunciato oggi una collaborazione con Rainforest Connection (RFCx), un'organizzazione no-profit che utilizza tecnologia acustica, Big Data e Intelligenza Artificiale/Machine Learning per salvare le foreste pluviali e monitorare la biodiversità. In questa collaborazione congiunta, RFCx esplorerà l'uso delle tecnologie di rilevamento dei gas di Infineon per elevare ed espandere le capacità dei loro attuali dispositivi di ascolto acustico utilizzati per monitorare e proteggere gli ecosistemi

vulnerabili della foresta pluviale. Ciò include il monitoraggio di primati, uccelli, rane, insetti, pipistrelli e altri organismi e la protezione da minacce che vanno dal disboscamento illegale a quelli che ora possono essere anche incendi boschivi con questa collaborazione.

Infineon consentirà di monitorare meglio gli ecosistemi della foresta pluviale in varie località, indipendentemente dalle condizioni meteorologiche e dalle preoccupazioni ambientali", ha affermato Topher White, CEO e fondatore di Rainforest Connection. "La nostra missione è sviluppare e sfruttare la tecnologia di

conservazione per potenziare i nostri partner e proteggere gli ecosistemi minacciati. Con questa nuova partnership tecnologica con Infineon, siamo entusiasti di portare nuovi livelli di tecnologia per aiutare a proteggere le foreste pluviali. Sono entusiasta di partecipare al loro evento annuale OktoberTech Americas".

Il sistema di monitoraggio scalabile e in tempo reale di RFCx protegge e studia gli ecosistemi remoti attraverso un approccio che utilizza sensori acustici – che presto includeranno le tecnologie di rilevamento dei gas di Infineon in determinate località – e AI/Apprendimento automatico per consentire ai partner sul campo di agire. Un monitoraggio aggiuntivo del panorama sonoro dell'ecosistema in determinate località con i sensori di gas di Infineon consentirà a RFCx e ai suoi partner di rilevare gli incendi in tempo reale, insieme ad altre potenziali minacce.

La piattaforma trasmette i dati all'interno delle regioni naturali e remote del mondo ai partner locali sul campo che potrebbero presto includere le autorità antincendio locali.

Rainforest Connection (RFCx) utilizza la tecnologia e i big data per salvare le foreste pluviali e gli habitat più minacciati del mondo.

www.infineon.com



successo operativo continuo delle implementazioni 5G, semplifica le operazioni e riduce i tempi di fermo.

I servizi di rete gestiti di Sierra Wireless sono disponibili in tre opzioni:

- Basic, la soluzione cellulare di backup conveniente per sistemi non critici.
- Premium, che include tutte le funzionalità del servizio Basic insieme a apparecchiature installate professionalmente, monitoraggio proattivo e un contratto di livello di servizio aumentato.
- Pro, che include tutte le funzionalità del servizio pacchetto Premium, nonché un'opzione dual modem sulla serie di rou-



ter XR, piani dati 5G e ancora più opzioni per personalizzare i requisiti di gestione.

Una recente ricerca della IDC (International Data Corporation) prevede che le connessioni IoT mobili e gli abbonati a servizi 5G raggiungeranno 1,01 miliardi entro il 2023.

www.sierrawireless.com/iot-blog

› VMware definisce il percorso per l'edge computing

VMware ha introdotto VMware Edge, un portafoglio di soluzioni per eseguire, gestire e proteggere app edge-native su più Cloud. L'edge è un'infrastruttura digitale distribuita per l'esecuzione di carichi di lavoro tra diverse location, posizionati vicino a utenti e dispositivi che producono e consumano dati. La collocazione di un carico di lavoro all'edge è fondamentale per soddisfare i requisiti delle app edge-native. Un carico di lavoro edge-native posizionato in qualsiasi punto tra il Cloud e la sede remota del cliente è fornito in modalità ad a service è chiamato near edge. Un carico di lavoro edge-native collocato in una posizione remota del cliente nelle immediate vicinanze degli endpoint viene chiamato far edge. Le app cloud-native richiedono un edge multcloud, che unisca servizi underlay in esecuzione su una rete di service provider (connettività privata, edge computing multi-access) a servizi overlay forniti on top.

Il tutto orchestrato da un piano di gestione che fornisca una osservabilità, configurazione, installazione, operation e gestione coerenti in tutte le edge location.

VMware Edge riunisce soluzioni VMware che consentono alle organizzazioni di eseguire, gestire e protegge-

re le app edge native su più cloud sia nelle posizioni near edge che far edge. Le soluzioni VMware Edge includono:

- VMware Edge Compute Stack, è una macchina virtuale integrata e purpose-built e uno stack basato su container appositamente progettato per modernizzare e proteggere le app edge native nel far edge.
- VMware SASE, che combina le funzionalità SD-WAN con funzioni di sicurezza fornite dal cloud, tra cui sicurezza web cloud, accesso alla rete zero trust e firewall. Queste funzionalità saranno fornite in modalità as-a-service in entrambe le

edge location da un network globale di points of presence (PoPs).

- VMware Telco Cloud Platform, che ha fornito soluzioni near edge ai più grandi provider di servizi di comunicazione al mondo a partire dal loro core 4G/5G fino alla rete di accesso radio (RAN).

VMware Edge Compute Stack sarà in grado di funzionare sui ThinkSystem SE350 Edge Server di Lenovo e si integrerà con SmartHub.ai, sistema edge IoT software-defined, per integrare e gestire gli asset IoT.

www.vmware.com



INDISPENSABILE IN LABORATORIO!



L'alimentatore è lo strumento più importante di ogni laboratorio elettronico da cui si esige prima di tutto la possibilità di regolare con buona risoluzione la tensione e la corrente di uscita, requisiti soddisfatti da questi nuovi alimentatori. Mantengono i valori impostati in modo affidabile e indipendentemente dalle fluttuazioni della tensione di alimentazione o del carico collegato.



DISPONIBILE ANCHE NELLE VERSIONI:

DA 0-30 VDC / 0-10 A

COD. AL3010 € 106,00

DA 0-60 VDC / 0-5 A

COD. AL6005 € 106,00

€ 78,00

Cod. **AL3005N**

ALIMENTATORE LINEARE CON USCITA REGOLABILE 0-30 VDC / 0-5 A

Alimentazione: 110/220 VAC, 50/60 Hz

Uscita duale regolabile:

- Tensione: da 0 a 30 VDC

- Corrente: da 0 a 5 A

Risoluzione della tensione: 10/100 mV

Risoluzione della corrente: 1/10 mA

Precisione visualizzazione della tensione: $\pm 1\%$ +1 cifra

Precisione visualizzazione della corrente: $\pm 1\%$ +2 cifre

Dimensioni: 220x165x85 mm

Peso: 1,350 kg



Prezzi IVA inclusa.

Pacchetti di sicurezza funzionale ISO 26262 da Microchip

Microchip ha sviluppato pack di sicurezza funzionale certificati ISO 26262 per lo sviluppo di progetti a livello ASIL B e ASIL C. L'ecosistema di sicurezza funzionale completo per i DSC dsPIC33C include DSC dsPIC33C pronti per la sicurezza funzionale AEC Q100 di grado 0 con funzioni di sicurezza hardware dedicate, SGS TÜV Saar certificato ASIL B Ready Failure Modes, Effects and Diagnostic Analysis (FMEDA) report e Functional Safety Manual (FSM), Librerie diagnostiche di sicurezza funzionale certificate TÜV per progetti destinati ad ASIL C, oltre a un'applicazione di riferimento per la sicurezza funzionale.

L'ecosistema completo di sicurezza funzionale per MCU PIC18 e AVR include MCU PIC18-Q84 con certificazione di sicurezza funzionale AEC Q100 Grado 1 con CAN FD e AVR DA MCU con interfaccia LIN, entrambi con supporto hardware per sensori tattili capacitivi, SGS TÜV



Saar certificato ASIL B Ready Failure Modes, Effects and Diagnostic Analysis (FMEDA) report e Functional Safety Manual (FSM), librerie diagnostiche di sicurezza funzionale e certificati e rapporti di certificazione.

Inoltre Microchip offre un pacchetto di tool di progettazione che includono un compilatore di sicurezza funzio-

le MPLAB XC certificato TÜV SÜD con il certificato TÜV SÜD, un manuale di sicurezza funzionale per il compilatore insieme a piani di sicurezza e completa classificazione degli strumenti e rapporti di qualificazione per il compilatore, MPLAB X Integrated Development Environment (IDE), MPLAB Code Coverage.

www.microchip.com

Sensore di posizione Hall 3D low power - 20 kSPS da TI

TMAG5170 è il primo sensore di posizione 3D ad effetto Hall a fornire un errore totale a fondo scala del 2,6% a temperatura ambiente.

Per ottenere un controllo in tempo reale più rapido e accurato, il sensore supporta misurazioni fino a 20 kSPS per un throughput



a bassa latenza del movimento meccanico ad alta velocità. Il TMAG5170 non richiede calcoli off-chip e integra funzionalità come un motore di calcolo dell'angolo, la media delle misurazioni e la compensazione del guadagno e dell'offset. Queste funzionalità consentono loop di controllo più rapidi, latenza del

sistema ridotta e sviluppo software più semplice, indipendentemente dal posizionamento del sensore. Le funzioni di calcolo integrate del sensore riducono anche

il carico del processore del sistema fino al 25%, consentendo agli ingegneri di utilizzare microcontrollori

generici (MCU) come gli MCU MSP430 di TI.

Inoltre il TMAG5170 implementa controlli per la comunicazione, di continuità e del percorso del segnale interno, nonché la diagnostica configurabile per l'alimentazione esterna, il campo magnetico e la temperatura del sistema.

www.ti.com

Da ST MOSFET STPOWER MDmesh a 800V e basse perdite

La nuova serie di MOSFET a super-giunzione STPOWER MDmesh K6 di STMicroelectronics migliora diversi parametri chiave per ridurre al minimo le perdite di potenza in applicazioni di illuminazione basate su topologia flyback, come driver LED, lampade HID, nonché adattatori e alimentatori per display.

Il MOSFET MDmesh K6, con la migliore $R_{ds(on)}$ a 800 V ($R_{ds(on)max} = 0,22\Omega$) attualmente disponibile sul mercato, consente nuovi design compatti che combinano un'elevata densità di potenza con l'efficienza leader di mercato.

Inoltre, la serie K6 ha una tensione di soglia ridotta rispetto alla precedente generazione MDmesh K5, consentendo una tensione di pilotaggio inferiore e riducendo così le

perdite di potenza nel driver e guadagnando efficienza, principalmente per le applicazioni in standby a zero watt.

Anche la carica totale di gate ($Q_{gtyp} = 25,9nC$) è molto bassa, consentendo elevate velocità di commutazione e basse perdite di pilotaggio.

Un diodo di protezione ESD integrato aumenta la robustezza complessiva del MOSFET fino alla Classe 2 del modello HBM (Human Body Model). In package TO-220, il MOSFET è già in produzione, mentre da gennaio 2022 arriveranno le versioni DPAK e TO-220FP.

www.st.com



La ricarica smart secondo Phoenix Contact



Charx control è il sistema modulare Phoenix Contact di controller per la ricarica dei veicoli elettrici, basato su Linux. Adattabili al wall box, come al parco di ricarica interconnesso, i controller sono dotati delle più comuni interfacce di comunicazione e possono essere

configurati tramite Web Based Management. La comunicazione, le reti, gli edifici, i punti di ricarica ed i veicoli sono il fattore chiave per una mobilità elettrica basata sulle rinnovabili ed è quindi necessario integrare i veicoli elettrici nella Smart Grid o nella Smart Home e definire

processi efficienti per gestire dispositivi e patch dedicate ai CPO (Charging Point Operators). Tali applicazioni IoT e servizi "Smart" richiedono una comunicazione sicura basata su vari protocolli e ciò può essere garantito dai controllori Charx. www.phoenixcontact.com

Da Renesas MCU compatibile Bluetooth 5.3 LE

Renesas sta sviluppando nuovi microcontrollori compatibili Bluetooth 5.3 Low Energy, della famiglia Renesas Advanced (RA) Arm Cortex-M a 32 bit. La nuova specifica Bluetooth 5.3 è stata rilasciata il 13 luglio 2021 e include importanti nuove funzionalità, come consentire ai ricevitori di filtrare i messaggi senza coinvolgere lo stack host per migliorare il ciclo di lavoro del ricevitore. Consente inoltre ai dispositivi periferici di fornire canali preferiti a un dispositivo centrale al fine di migliorare il throughput e l'affidabilità. Inoltre, aggiunge connessioni subclassificate che migliorano il tempo di commutazione tra connessioni a ciclo di lavoro basso e alto per le applicazioni che occasionalmente devono passare al traffico burst. Oltre a queste caratteristiche, la funzionalità di ricerca della direzione introdotta in Bluetooth 5.1 e i canali isocroni aggiunti in Bluetooth 5.2 per la trasmissione audio stereo saranno supportati nei nuovi MCU di Renesas. L'inclusione di funzionalità SDR (Software Defined Radio)



consentirà ai clienti di migrare in seguito a nuove versioni delle specifiche. I nuovi prodotti saranno supportati dal pacchetto software flessibile (FSP) della famiglia RA e dal plug-in Renesas QE per Bluetooth LE, un profilo Bluetooth dedicato e uno strumento di supporto per lo sviluppo, oltre che il supporto TrustZone. Renesas si sta preparando a offrire più

combinazioni vincenti con i nuovi MCU Bluetooth 5.3 LE insieme a dispositivi analogici, di alimentazione e di temporizzazione complementari. Le combinazioni vincenti forniscono un'architettura facile da usare, semplificando il processo di progettazione e riducendo significativamente il rischio di progettazione per i clienti in un'ampia varietà di applicazioni. www.renesas.com

SK hynix lavora alla DRAM HBM3

SK Hynix, ha ultimato lo sviluppo di chip di memoria di ultima generazione basato sulla tecnologia HBM3 (High Bandwidth Memory 3), le cui prestazioni sono superiori a qualsiasi altra DRAM attualmente disponibile. La HBM3 consente di collegare verticalmente più chip di DRAM in modo da aumenta-

re in maniera significativa la velocità di trasferimento dei dati. L'HBM3 di SK hynix non è solo la DRAM più veloce al mondo, ma è anche dotata della maggiore capacità e di un livello di qualità superiore. Può elaborare fino a 819 GB/s, equivalenti a trasferire 163 filmati Full HD (5 GB ciascuno) al secondo. L'HBM3 di SK hynix

sarà fornito in due capacità di 16 GB e 24 GB. L'altezza dei chip DRAM del 24 MB a circa 30 micron, con 12 chip impilati utilizzando la tecnologia TSV (Through Silicon Via) con elettrodi di interconnessione che collegano i chip superiore e inferiore attraverso migliaia di fori sui chip DRAM. <https://www.skhynix.com/>



Extender HDMI a tecnologia Semtech



Semtech mette la propria tecnologia AVX a disposizione della nuova linea di extender HDMI VuWall, resistenti alle EMI, con lunghezze dei cavi garantite standard IEEE e disponibili in rame e fibra ottica.

Con l'emergere di nuove tecnologie di estensione AV, l'AVX200T di Semtech si spinge oltre la connettività multimediale tra apparecchiature AV e display remoti. Inoltre trasmette simultaneamente video 4K di qualità non compresso, audio multicanale, Gigabit Ethernet, USB e altri segnali di controllo su cavi in rame o fibra ottica. Gli extender HDMI 2.0 di VuWall estendono i segnali audio/video con cavo CAT o fibra ottica,

supportando una risoluzione fino a 4K@60Hz (full 4:4:4), compatibile con HDR e HDCP 2.2, permettendo di distribuire video e audio non compressi, sugli schermi senza perdita di segnale o ritardo. VuWall, leader tecnologico globale nei sistemi di controllo dei video wall e nelle soluzioni di visualizzazione unificata, ha aperto la strada alla nuova tecnologia ibrida VuTrex, che collega i sistemi AV, IT e IP per gestire e controllare senza problemi i contenuti visivi in tutte le organizzazioni. VuWall offre un ecosistema all'avanguardia di controller, encoder e decoder per video wall, tutti gestiti dalla loro esclusiva piattaforma di gestione centralizzata TRx, che elimina la complessità del controllo video wall tradizionale e della distribuzione del segnale su IP, senza richiedere programmazione.

www.semtech.com

Raspberry Pi Build HAT da Farnell

Si chiama Build HAT ed è il nuovo shield Farnell per l'integrazione col sistema educativo LEGO, offrendo a studenti, insegnanti, appassionati di elettronica alle prime armi e produttori una maggiore flessibilità durante la progettazione di nuovi prototipi. Il Build HAT e il sistema LEGO possono essere utilizzati per creare robot, automobili, giochi, una casa intelligente ecc. Il Raspberry Pi Build HAT è il secondo prodotto Raspberry Pi basato sul microcontrollore RP2040. La piastra LEGO Maker consente agli utenti di integrare facilmente i computer Raspberry Pi nel sistema di costruzione LEGO. Il Build HAT è compatibile con la generazione più recente di motori LEGO Technic e con i sensori inclusi



nel portfolio LEGO Education SPIKE Prime. Il Build HAT è facile da usare da Python su Raspberry Pi. L'alimentatore abbinato è da 48W, funziona con la rete a 110-240 Vca e fornisce 8 Vcc - 6 A. Farnell offre un'ampia gamma di dispositivi educativi e ha collaborato con diverse organizzazioni educative e governi per supportare l'implementazione strategica delle soluzioni di apprendimento STEM nelle scuole.

www.farnell.com

Da Toshiba nuovi MCU ARM Cortex-M4 a 40 nm

Toshiba Electronics ha annunciato la disponibilità di 20 nuovi dispositivi della famiglia M4N di microcontrollori ARM Cortex-M4 prodotti con un processo a 40 nm. Questi dispositivi estendono e migliorano ulteriormente la classe avanzata della famiglia TXZ+TM e includono un core ARM Cortex-M4 con unità a virgola mobile (FPU), funzionante a velocità fino a 200 MHz, codice Flash integrato da 2 MB e memoria flash dati da 32 kB con durata del ciclo di scrittura di 100 k. Questi nuovi microcontrollori offrono anche una gamma di opzioni di interfaccia e comunicazione come Ethernet, CAN e USB 2.0 FS OTG con PHY integrato.

Questi tipi di microcontrollori del gruppo M4N sono ideali per apparecchiature per ufficio, applicazioni di automazione di edifici e di fabbrica, nonché per reti industriali e dispositivi di gestione delle informazioni. All'interno dello spazio consumer, questi dispositivi sono adatti per elettrodomestici basati su IoT, sicurezza domestica, contatori intelligenti e altro ancora.

I nuovi prodotti del gruppo M4N hanno funzioni di comunicazione avanzate integrate tra cui un'interfaccia di memoria seriale che supporta anche Quad/Octal SPI, interfaccia audio (I²S) e interfaccia bus esterno oltre a UART, FUART, TSPI e I²C supportate da un 3- unità DMAC. Inoltre, i dispositivi possono allocare DMA e RAM indipendenti per ciascun circuito periferico e utilizzare una configurazione del circuito a matrice di

bus che consente al master del bus di trasferire in modo efficiente i dati. Di conseguenza, i dispositivi del gruppo M4N consentono a un controller Ethernet, CAN e controller USB di essere elaborati in modo indipendente in parallelo allo stesso tempo.

I dispositivi appena annunciati supportano una varietà di applicazioni di rilevamento con un convertitore analogico/digitale (ADC) a 12 bit ad alta velocità e alta precisione che consente di impostare i tempi di campionamento e mantenimento individuali per ciascuno dei 24 canali di ingresso ADC. I nuovi dispositivi includono anche un convertitore digitale-analogico (DAC) a 8 bit a doppio canale e una gamma di funzioni di controllo del motore.

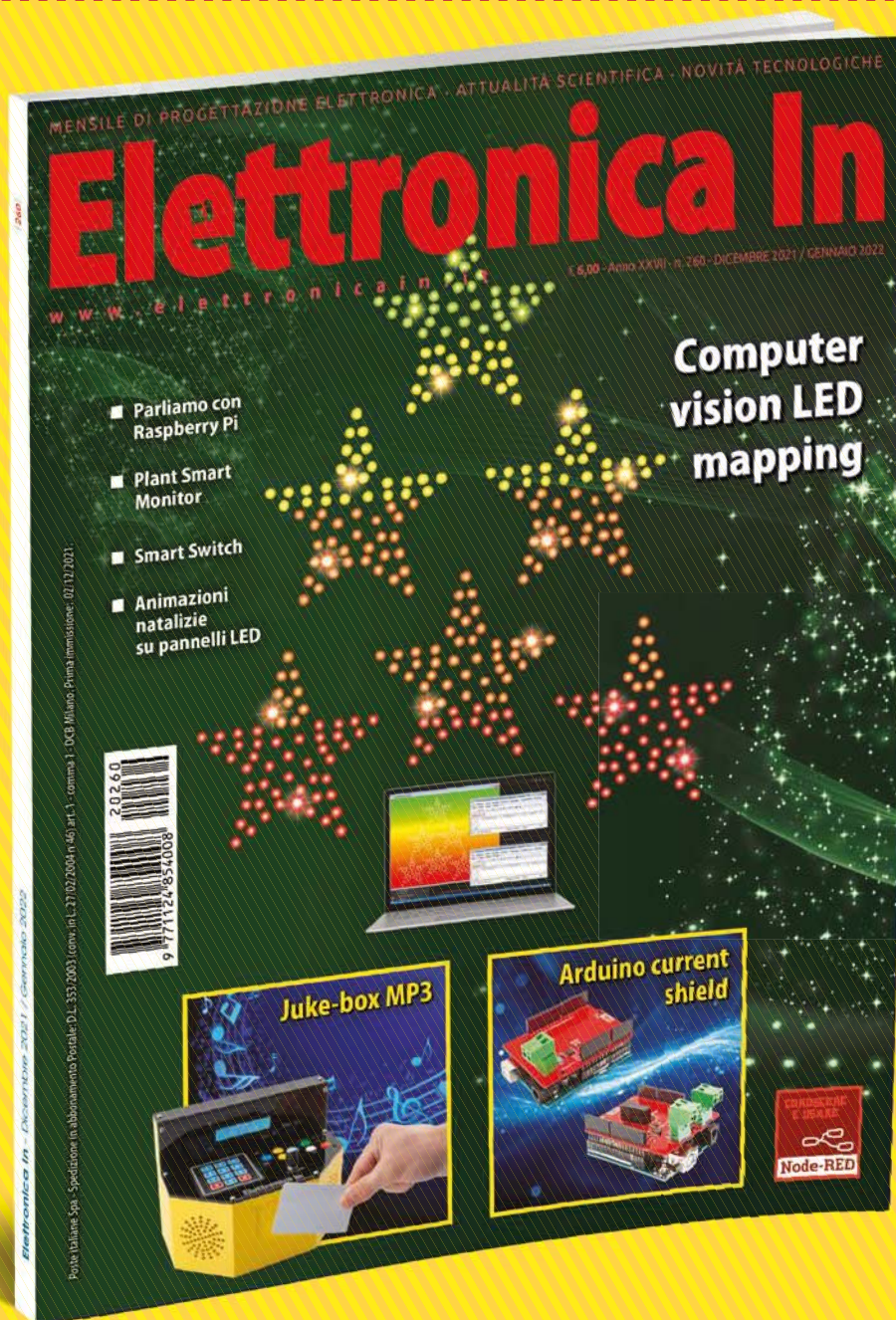
Le funzioni di autodiagnosi incorporate nei dispositivi per ROM, RAM, ADC e clock aiutano i clienti a ottenere la certificazione di sicurezza funzionale IEC60730 Classe B. La documentazione completa, il software di esempio (con esempi di utilizzo effettivo) e il software del driver che controlla l'interfaccia per ciascuna periferica sono disponibili per il download sul sito Web Toshiba. I nuovi dispositivi sono disponibili in package a passo fine LQFP e VFBGA

www.toshiba.com





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/abbonamenti



ABBONAMENTO ANNUALE PER RIVISTA DIGITALE

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

COME ABBONARSI

→ **on-line**

compilando il modulo riportato
nella pagina "**Abbonamenti**"
del nostro sito www.elettronica.in.

→ **e-mail**

scrivendo a
abbonati@elettronica.in

→ **telefono**

telefonando a
+39-0331-752668

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

50% DI SCONTO per l'acquisto di un volume
della libreria "Elettronica In"

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

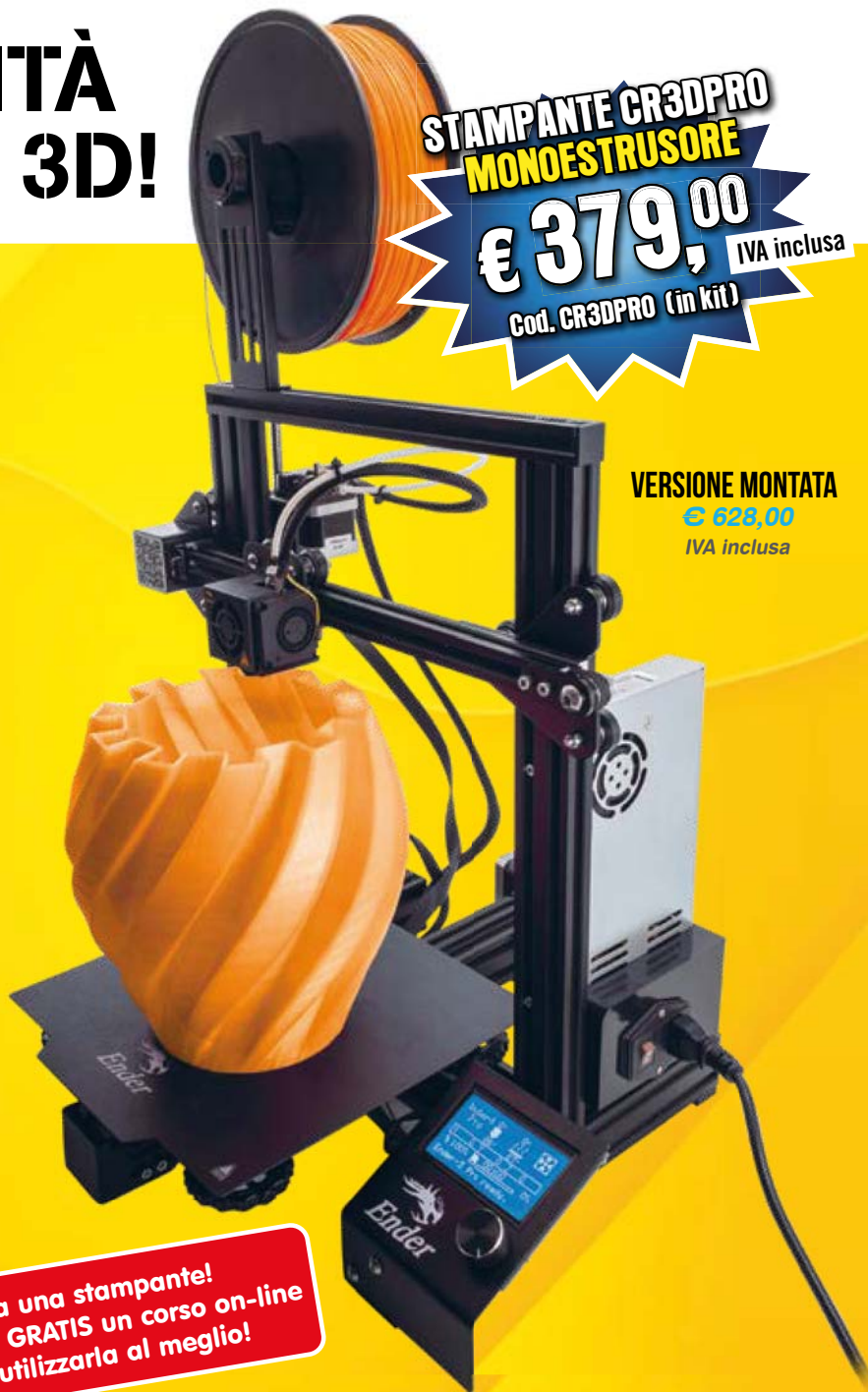


ULTIME NOVITÀ DAL MONDO 3D!

CR3DPRO

**STAMPANTE 3D PER STAMPE
FINO A 22 x 22 x 25 cm**

- ✓ Tecnologia FFF FDM
- ✓ Diametro filamento 1,75 mm
- ✓ Volume di stampa (xyz) 220 x 220 x 250 mm
- ✓ Dimensione Nozzle 0,4 mm
- ✓ Estrusore MK10
- ✓ 4 motori passo-passo Nema17 (200 step)
- ✓ Spostamento asse Z con vite trapezia Ø8 mm a 3 principi in acciaio INOX
- ✓ Struttura con profilati in alluminio
- ✓ Scocca in alluminio a struttura aperta
- ✓ Livellamento del piatto manuale di precisione
- ✓ Schermo LCD
- ✓ Firmware Marlin open source
- ✓ Connettività SD - USB
- ✓ Sistema di recupero della stampa
- ✓ Slicing consigliato Cura
- ✓ Sistemi operativi Windows, Mac OSX, Linux
- ✓ Dimensioni telaio (xyz) 440 x 410 x 650 mm
- ✓ Peso 5,6 Kg
- ✓ Alimentazione 200-240 V / 3.4 A
- ✓ Output 24VDC - 270W



VERSIONE MONTATA
€ 628,00
IVA inclusa

Acquista una stampante!
Potrai seguire GRATIS un corso on-line
su come utilizzarla al meglio!



VERSIONE MONTATA
€ 1148,00
IVA inclusa

CRCP3IN1

**MACCHINA 3 IN 1:
STAMPANTE 3D,
INCISORE LASER
E FRESATRICE CNC**

**CRCP3IN1
MACCHINA 3 IN 1**

€ 899,00 IVA inclusa
Cod. CRCP3IN1 (in kit)

- ✓ Telaio in profilati di alluminio V-slot da 40 x 40 mm
- ✓ Scocca in alluminio a struttura aperta
- ✓ 5 motori passo-passo Nema17 (200 step)
- ✓ Spostamento asse Z: doppia vite trapezia Ø8 mm a 3 principi in acciaio INOX
- ✓ Struttura con profilati in alluminio 40 x 40 mm
- ✓ Schermo grafico touch da 4,3"
- ✓ Firmware Marlin open source
- ✓ Connettività SD - USB
- ✓ Alimentazione 200-240 V / 3.4 A
- ✓ Output 24 VDC - 270 W
- ✓ Dimensioni telaio (xyz): 400 x 450 x 650 mm
- ✓ Peso 10 Kg



ARDUINO CURRENT SHIELD

dell'Ing. **VINCENZO MENDOLA**

Il circuito integrato INA219 della Texas Instruments, permette la realizzazione di sistemi di misura della corrente continua in modalità shunt anche con Arduino; eccovi due shield, uno galvanicamente isolato e uno non, per sperimentare.

A

rduno ci ha abituati a fare di tutto: controllare carichi "locali" o remoti, acquisire dati ambientali attraverso un gran numero e una vasta tipologia di sensori diversi, rendendo disponibili i valori misurati localmente o a distanza tramite una connessione di rete, raggiungendo potenzialmente qualsiasi luogo attraverso internet, tutto questo tramite delle specifiche schede, gli "shield", appositamente progettati per essere meccanicamente ed elettricamente connessi ad arduino. Una funzionalità "trasversale" che potrebbe essere utile in varie applicazioni è la misura della corrente circolante in un circuito che alimenta o è alimentato da arduino. Potremmo, ad esempio, misurare la corrente assorbita da un carico DC connesso ad Arduino, verificare l'assorbimento di arduino stesso e degli shield a questo connessi o magari misurare la corrente e la tensione erogata da un piccolo pannello fotovoltaico utilizzato per alimentare, magari insieme ad una batteria al litio, il nostro sistema IoT. L'idea di realizzare un current sense shield non è nuova, così come non lo è l'utilizzo dell'integrato scelto per farlo, ma a differenza di quanto comunemente disponibile e già visto anche su queste pagine, in questo articolo vorremmo presen-

tare un dispositivo che sia in grado di misurare la corrente circolante in un sistema alimentato in corrente continua, con un carico che sia isolato o che debba o che si voglia isolare dalla circuiteria che alimenta arduino. Normalmente i sistemi di misura che sono stati finora presentati sulle riviste oppure in rete, condividono lo stesso riferimento di massa che fa capo al microcontrollore.

Ci sono particolari applicazioni in cui questa connessione deve essere volutamente separata, solitamente in ambiti medicali o industriali (per soddisfare gli elevati valori di isolamento di tensione richiesti dalla normativa di riferimento) e in alcuni casi più specifici semplicemente per disaccoppiare le masse; difficilmente sono situazioni che si presentano al comune hobbista ma vogliamo comunque mostrarvi un modo semplice per affrontare anche questa necessità, qualora vi si dovesse presentare o vogliate provare a sperimentare. Vediamo innanzitutto come è possibile realizzare un semplice current shield e come poi questo possa essere reso "isolato".

Partiamo dalla scelta dell'integrato che costituisce il cuore dello shield, dato che effettua fisicamente la misura della corrente nella sezione di circuito di cui siamo interessati, tramite l'utilizzo di una resistenza di shunt di piccolo valore, posta in serie al nostro carico, nella quale scorre la corrente da misurare.

L'integrato misura la caduta di tensione ai capi del resistore di valore noto e sufficientemente accurato (almeno 1%) da cui determina tramite la Legge di Ohm, il valore della corrente I in esso circolante, essendo

$$I = \frac{V}{R}$$

dove V rappresenta la caduta di tensione ai capi del resistore ed R la sua resistenza.

SCELTA DEL CURRENT SENSE: INA219

Ci sono moltissimi circuiti integrati progettati appositamente per realizzare dei sistemi di misura in corrente continua, semplici, compatti, affidabili ed economici; ce ne sono sia di tipo analogico, sia di tipo digitale, alcuni integrano al loro interno la resistenza di shunt, altri integrano tutta la parte di condizionamento del segnale e possono fornire in uscita, oltre al valore di corrente, la tensione a cui è alimentato il carico ed il valore di potenza assorbita dallo stesso; generalmente questa tipologia di dispositivi, lavora su bus I²C, SPI o SMBus.

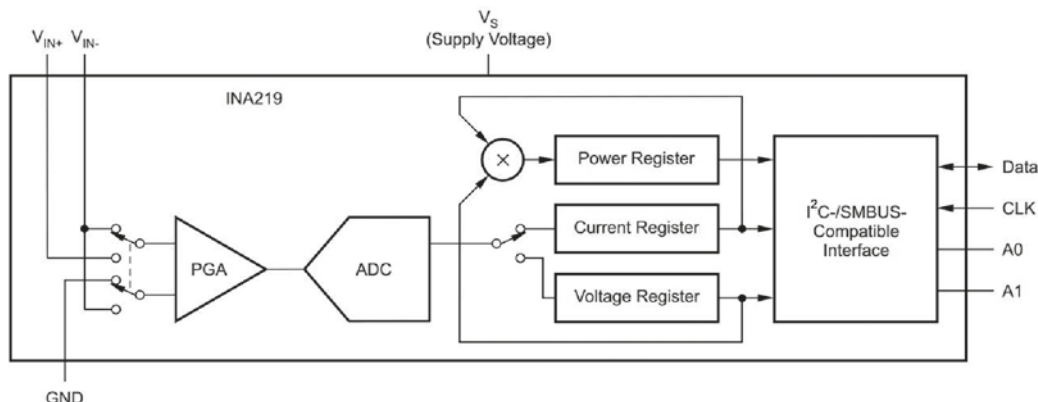
Un circuito integrato facilmente reperibile adatto per i nostri scopi è ad esempio l'INA219, prodotto dalla Texas Instruments: è un current sense bidirezionale con power monitor funzionante con un'interfaccia I²C o SMBus compatibile, che garantisce un'accuratezza dello 0.5% su tutto il range di temperatura per cui è qualificato a funzionare (-40°C - +125°C). È disponibile in due versioni, "A" e "B" in cui la seconda garantisce maggior accuratezza di misura (+/-0.5%) ed è realizzato in package per montaggio SMT (Surface Mount Technology) sia di tipo SOIC-8 che SOT-23, entrambi ad 8 pin. Il dispositivo fornisce oltre al valore della corrente circolante I , il valore della tensione V presente sulla linea che alimenta il carico (compresa tra 0 e 26V) ma anche la caduta di tensione ai capi della resistenza di shunt; avendo a disposizione il valore puntuale di queste grandezze l'integrato può facilmente calcolare e fornire l'entità della potenza fornita al carico, essendo questa pari a $P=V \cdot I$.

INA219: COM'È FATTO E COME FUNZIONA

In Fig. 1 è visibile una schematizzazione del funzionamento dell'integrato, come riportata nel datasheet ufficiale della TI.

Come si può vedere dallo schema a blocchi, al suo interno è presente un PGA, un ADC, i 3 registri

➔ **Fig. 1**
schema a blocchi
interno del current
sense INA219.



relativi alle misure di corrente, tensione e potenza (ottenuta come prodotto delle altre due grandezze) ed un'interfaccia I²C/SMBUS compatibile.

La tensione di alimentazione dev'essere compresa tra 3 e 5,5V mentre, come abbiamo visto, la tensione applicata al carico di cui si vuole misurare la corrente circolante, può essere compresa tra 0 e 26V; non ci sono particolari sequenze da rispettare nella loro applicazione: può essere indifferentemente presente prima la tensione di alimentazione o quella di misura o viceversa senza alcun ordine o sequenza da rispettare.

Tramite il registro "MODE" è possibile configurare l'INA219 per funzionare in modalità "continua", in cui acquisisce e campiona i valori corrispondenti alle grandezze di misura e li converte in uscita senza interruzione, oppure in modalità "triggerata", in cui misura i valori di tensione e di corrente solamente in corrispondenza del verificarsi di un particolare evento di trigger. Il PGA (Programmable Gain Amplifier) all'ingresso della catena di acquisizione permette di impostare un guadagno di 2, 4 oppure 8 (320mV) per il valore della tensione di shunt, e due valori di fondo scala per la tensione del BUS, pari a 16 o 32V (pur restando la massima tensione applicabile pari a 26V). L'integrato è in grado di misurare sia la tensione differenziale tra IN+ e IN- sia la tensione di modo comune su IN-. L'ADC è di tipo delta-sigma con frequenza di campionamento di 500 KHz(+/-30%), con una buona reiezione del rumore; la Texas Instruments suggerisce di applicare un semplice filtraggio all'ingresso dello stadio facente capo al PGA, consistente in due resistenze in serie a IN+ e IN- di valore pari a 10Ω ed un piccolo condensatore ceramico a questi in parallelo, se ci sono transienti da misurare che hanno frequenze multiple (armoniche) della frequenza di campionamento di 500 kHz, quindi a partire da 1 MHz e superiori. Sperimentalmente abbiamo verificato che effettivamente questo filtraggio non fornisce nessun vantaggio significativo nel caso di misure di corrente in condizioni normali e comuni; in ogni caso, come anche indicato dal costruttore, la presenza delle due resistenze inficia in maniera trascurabile il valore di accuratezza della misura finale ed anzi queste forniscono una piccola protezione degli ingressi dell'integrato nel caso di dV/dt causati principalmente da picchi di tensione derivanti da effetti induttivi legati a particolari condizioni di funzionamento. Essendo in ogni caso un progetto dimostrativo e di carattere divulgativo, si è scelto di mantenere la realizzazione il più semplice possibile e quindi di non utilizzare tale filtraggio

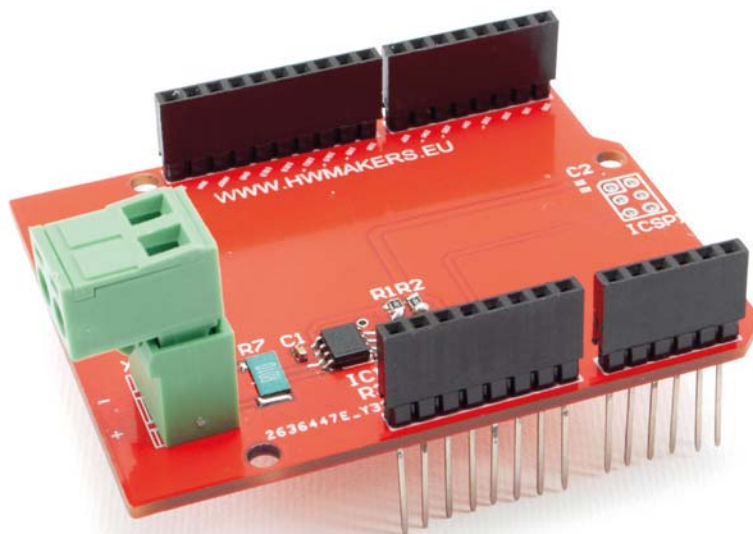
all'ingresso dei pin afferenti al PGA dell'INA219. Affinché l'INA219 sia in grado di effettuare correttamente la conversione delle grandezze che si vogliono misurare, bisogna impostare adeguatamente il valore di risoluzione nel "current register" così come il valore del resistore di shunt "R_{SHUNT}". Come abbiamo visto, l'integrato è in grado di misurare sia la tensione differenziale ai capi di IN+ e IN- sia la tensione ai capi di IN-. Si consiglia un'attenta e approfondita lettura del datasheet del componente per una completa e dettagliata conoscenza del funzionamento dello stesso.

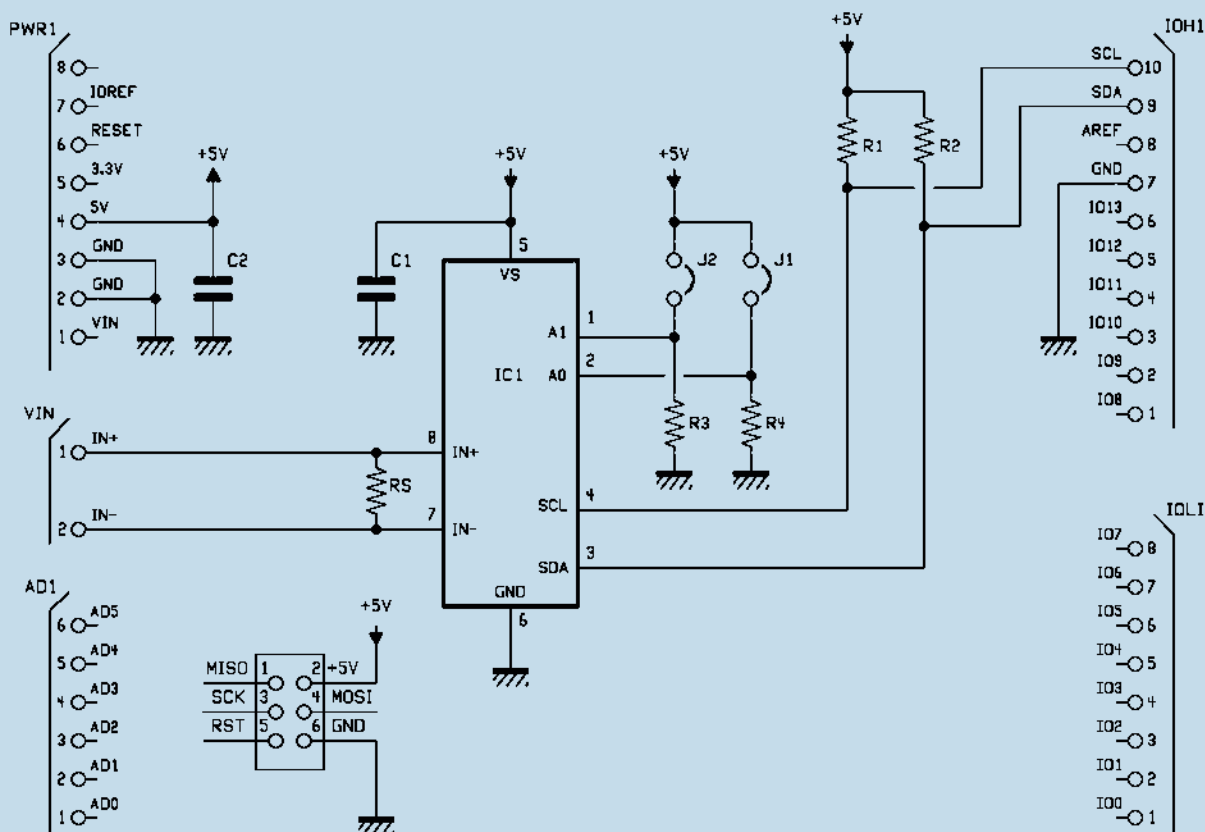
SHIELD INA219 NON ISOLATO

Come abbiamo visto, il circuito integrato INA219 è veramente un ottimo componente, che ci aiuta grandemente nella realizzazione di un completo circuito di misura della corrente continua, questo perché integra al suo interno tutto quanto è necessario per realizzare un completo sistema di misura di corrente, tensione e potenza, senza la necessità di far effettuare al microcontrollore a cui è connesso alcun calcolo, come ad esempio il prodotto della tensione per la corrente circolante, per determinare la potenza assorbita dal carico e senza dover ricorrere ad alcun condizionamento del segnale dal punto di vista elettronico: in uscita all'integrato, su BUS I²C, vengono forniti i valori della corrente circolante nel carico, della tensione ai capi dello stesso e della potenza da questo assorbita. Per questo motivo il circuito integrato INA219 della Texas Instruments permette la realizzazione di un current sense shield per Arduino o mbed o per altri sistemi di sviluppo a microcontrollore, in maniera veramente semplice e quindi alla portata di tutti.



Lo shield non
isolato.





Come abbiamo visto, l'INA219 ha al suo interno tutto il necessario per effettuare le misure richieste, per cui il circuito complessivo risulta piuttosto semplice: oltre all'INA219 (IC1), sono presenti le resistenze di pull-up per il corretto funzionamento della comunicazione I2C o SMBus (R1, R2) connesse ai pin 9 (SDA) e 10 (SCL) del connettore a 10 pin di arduino; R3 ed R4 sono le due resistenze da 0 ohm, utilizzate in combinazione a J1 e J2 per definire l'indirizzo di IC1 utilizzato dal bus I2C nel caso in cui sullo stesso ci dovessero essere più dispositivi e quindi fosse necessario identificarli univocamente; RS è la resistenza di shunt, per la quale generalmente si utilizza un resistore dal valore di di 0.1 o 0.01 ohm con un'accuratezza di almeno 1% e possibilmente superiore.

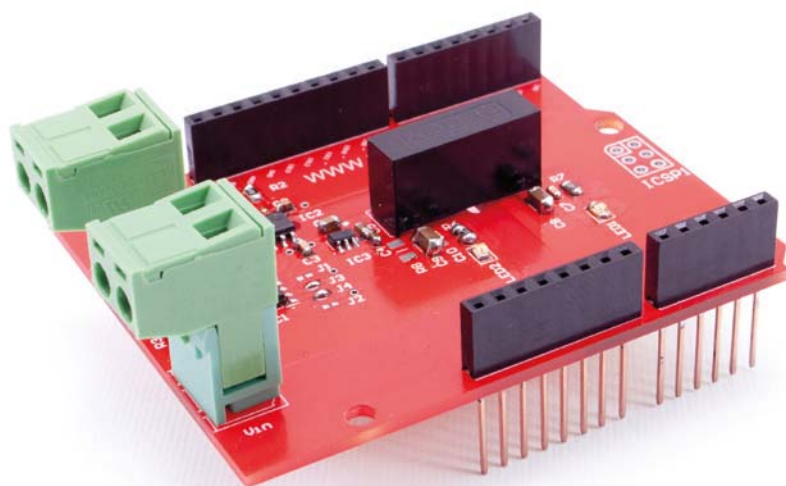
RS può anche essere di valori differenti a quelli appena visti, purché si mantenga la caduta di tensione ai suoi capi, in corrispondenza del massimo valore di corrente, il più basso possibile, tale da non superare la potenza massima dissipabile dal

resistore e comunque tale per cui non si superi il valore massimo ammesso dall'integrato, pari a 320mV. C1 e C2 sono i due condensatori di disaccoppiamento dell'alimentazione; C2 può essere tranquillamente non montato, è stato inserito in caso si voglia disaccoppiare ulteriormente l'alimentazione a 5V in prossimità del connettore "ICSP1"; infine VIN (X1) è il connettore cui verrà connesso il carico: "IN+" costituisce il morsetto cui si collega la tensione positiva della linea di alimentazione, mentre "IN-" deve essere connesso al morsetto di "entrata" del carico come visibile in **Fig. 2** (l'altro morsetto del carico va collegato a massa).

Come visibile in **figura 2** e dallo schema elettrico, il circuito stampato e le connessioni rispecchiano lo standard, la forma ed il layout degli shield per arduino UNO e derivati; successivamente vedremo come lo shield può essere facilmente adattato alla nuova famiglia "MKR" per chi di voi abbia necessità o voglia semplicemente testare l'invio nel cloud delle proprie misure.

SHIELD INA219 ISOLATO

La realizzazione di uno shield isolato, a partire da quello appena visto, è relativamente semplice e consiste essenzialmente nell'introduzione di un convertitore DC/DC isolato (IC4) per garantire l'isolamento della tensione di alimentazione e di un isolatore bidirezionale compatibile con I²C (IC2). IC4 e IC2 devono essere scelti in base al valore massimo di tensione d'isolamento che si vuole complessivamente avere; nel nostro caso abbiamo scelto per IC4 un convertitore della Recom, il modello REM1-0505S, che garantisce un isolamento di 5,2kVcc ma è possibile montare componenti differenti, perché sul circuito stampato è stato previsto un doppio passo, con entrambi i pin 5 e 6 connessi a massa, in modo da poter coprire una grande varietà di modelli di produttori differenti, purché questi abbiano comunque package SIP7. Fate riferimento ai datasheet specifici del convertitore che intendete utilizzare per essere sicuri della compatibilità dello stesso, a meno che non abbiate intenzione di utilizzare un layout differente da quello da noi proposto. Per quanto riguarda IC2, l'ISO1640 assicura un isolamento certificato di 5kVRMS, anche in questo caso più che sufficienti per i nostri scopi. Lo schema elettrico completo dello shield isolato è visibile nella pagina seguente: Lo schema presenta molte somiglianze con quello precedente, ma con l'aggiunta dei componenti necessari ad assicurare l'isolamento desiderato,



Lo shield isolato.

tra cui abbiamo IC2 con i condensatori di filtraggio sia in ingresso che in uscita, i due LED utilizzati per avere un'indicazione visiva della presenza della tensione di alimentazione su entrambe le parti di circuito le cui alimentazioni si vogliono elettricamente separare; nel nostro caso abbiamo 5V sia in ingresso che in uscita ma esistono anche convertitori che permettono di avere tensioni differenti tra le due parti, effettuando la conversione sia a valori inferiori che superiori a quelli applicati all'ingresso, a seconda delle proprie esigenze. A valle di IC4 troviamo IC3, un LDO a basso dropout e basso rumore della Texas Instruments, che riduce la tensione isolata di 5V a 3.3V, che utilizziamo per alimentare l'INA219. A differenza del caso precedente dello shield non isolato, in cui abbiamo alimentato l'integrato current sense a 5V, cioè lo stesso valore con cui viene alimentato l'arduino, questa volta con lo

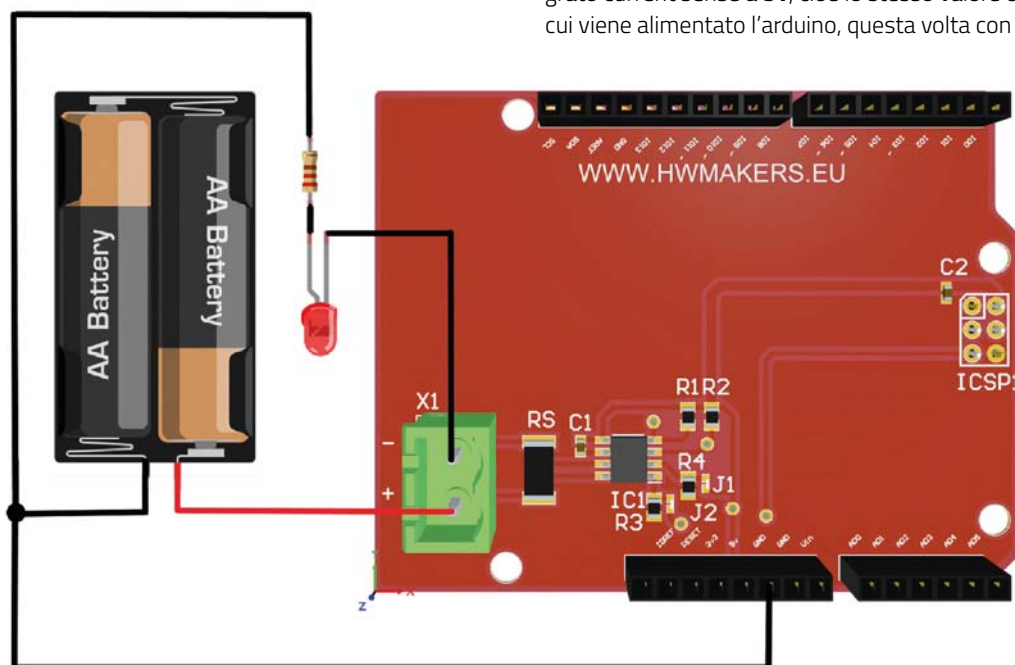
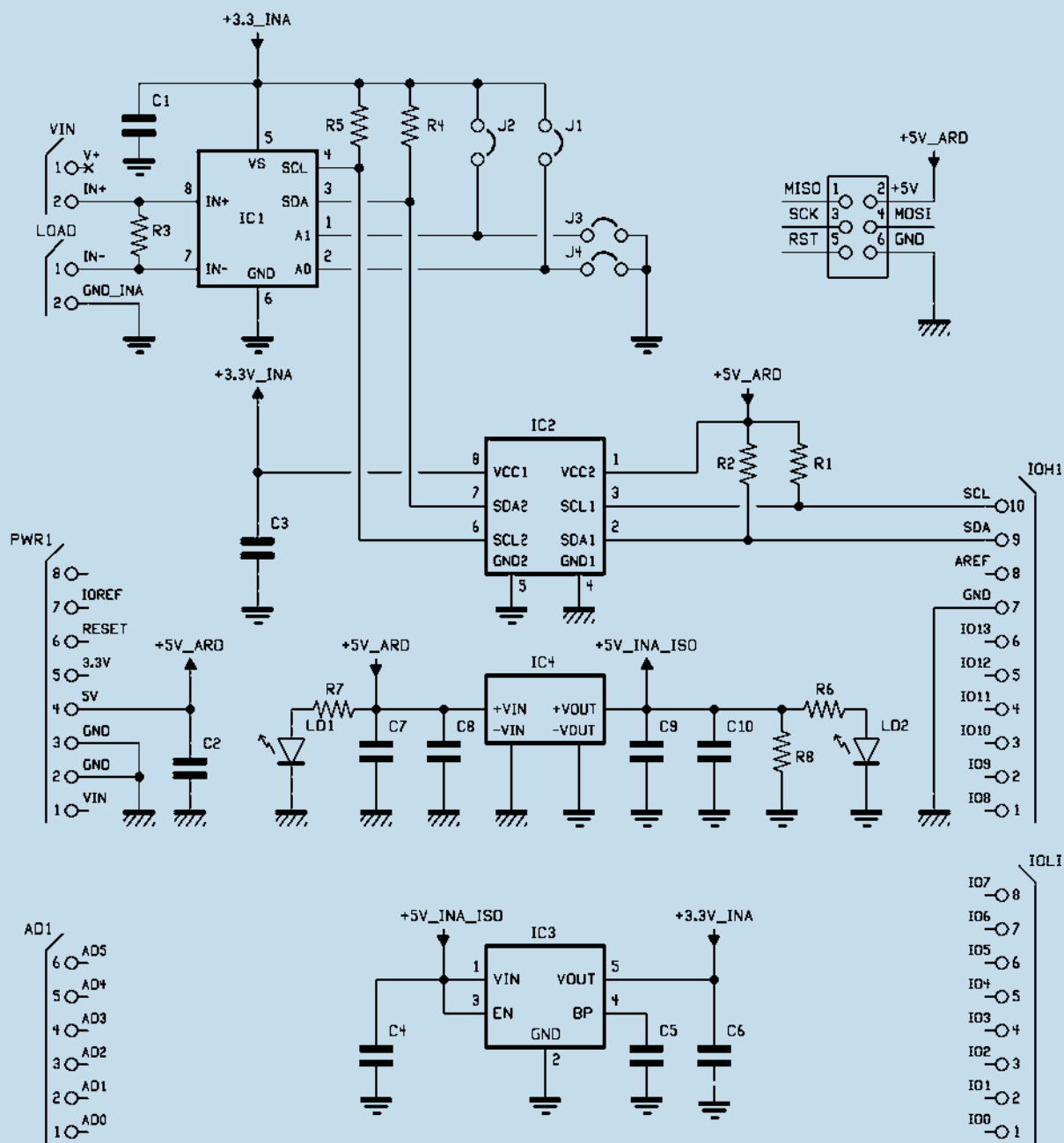


Fig. 2
Schema di collegamento del carico allo shield INA219 non isolato.



shield isolato abbiamo voluto mostrare cosa succede isolando oltre all'alimentazione, anche il bus I2C: la tensione ai due capi dell'isolatore, nel nostro caso IC2 ISO1640BDR, può essere differente, questo perché potrebbe capitare di dover utilizzare una comunicazione I2C tra due componenti facenti capo a parti del circuito che necessitano di tensione di alimentazione differente, ad esempio nel caso in cui alimentassimo, come avviene nel nostro caso, il microcontrollore a 5V ed il sensore con cui questo deve comunicare in I2C richiedesse

invece di una tensione di alimentazione differente, ad esempio 3,3V, come abbiamo deciso di fare in questo caso con l'INA219. Infatti osservando più nel dettaglio lo schema elettrico, e dal relativo datasheet vediamo che IC2, utilizzato per isolare il bus I2C, prevede due tensioni di alimentazione, non necessariamente identiche, ognuna per alimentare una delle due sezioni da isolare: al pin 1 abbiamo una tensione di 5V, fornita dalla Arduino, mentre al pin 8 abbiamo una tensione di 3,3V, isolata dalla prima, con cui alimentiamo l'INA219.

Ovviamente sarebbe stato possibile utilizzare anche un DC-DC isolato con ingresso a 5V e uscita a 3,3V e fare a meno del LDO IC3 ma questi componenti sono meno diffusi e comunque ci sembra interessante mostrare come sia possibile, partendo da una tensione isolata ottenuta da un convertitore switching, ottenere una tensione a basso rumore, in uscita da un piccolo LDO collegato a valle del convertitore, utilizzata per alimentare un current sense. Si ricorda inoltre che questi convertitori non sono stabilizzati, per cui la tensione di uscita dipende dal carico, ed è pertanto consigliato l'uso di un LDO.

R1 e R2 sono le due resistenze di pull-up dell'I²C, lato arduino ma, essendo il bus isolato, in questo caso è necessario utilizzare anche due resistenze di pull-up dal lato dell'INA219, R4 e R5, collegate, come abbiamo appena visto, a 3,3V invece che a 5V. I ponticelli J1, J2, J3 e J4 hanno la stessa funzione di definizione dell'indirizzo dell'INA219 nella comunicazione del bus I²C vista in precedenza, utile se si dovessero avere più elementi che devono comunicare con la MCU sullo stesso bus.

La resistenza R8 è stata prevista perché alcuni convertitori isolati per funzionare correttamente hanno bisogno di essere collegati ad un carico a vuoto minimo diverso da zero, per cui, come visto in precedenza, fate riferimento al datasheet del vostro integrato. Nel caso del REM1-0505S R8 non è necessaria, perché si può lasciare l'uscita anche senza carico.

Al connettore VIN (X1) si collega la tensione con cui dev'essere alimentato il carico, facente capo all'ingresso "IN+" del current sense, mentre il connettore LOAD (X2) serve per collegare il carico e fa capo a "IN-" e alla massa isolata del circuito.

Dalla rappresentazione in 3D in **Fig. 4** e dal piano

di montaggio si possono vedere le due fresature sotto IC4 e IC2, create per garantire l'elevato isolamento di tensione dei componenti anche una volta montati sul PCB.

Anche in questo caso si è realizzato il circuito stampato in modo da rispettare, oltre al pinout dei connettori, anche la forma e la posizione dei fori di fissaggio, per mantenere la compatibilità con i prodotti ufficiali, assicurando una semplice l'interoperabilità con tutto l'ecosistema Arduino. Anche se lo shield è stato sviluppato per il fattore di forma originale, cioè per le board che vanno dalla Arduino UNO in avanti, anche lo shield isolato può essere facilmente adattato alla nuova famiglia "MKR", come verrà spiegato successivamente in questo articolo.

LIBRERIE E SKETCH

La complessità della configurazione di tutti i parametri di funzionamento dell'integrato è semplificata dall'utilizzo di una delle librerie dedicate realizzate per l'INA219. Tra le tante disponibili, abbiamo scelto di utilizzare "ArduinolNA219" raggiungibile su github al link <https://github.com/flav1972/ArduinolNA219> perché ci è sembrata quella in grado di permettere una configurazione e quindi una personalizzazione dei parametri di funzionamento più semplice delle altre; ovviamente nulla vieta di usarne un'altra o di scriverne una propria. L'utilizzo della libreria è possibile indifferentemente con entrambi gli shield, sia quello isolato sia quello non isolato. Vediamo più nel dettaglio il funzionamento della libreria scelta:

- nella funzione di "setup()" bisogna richiamare le funzioni "configure" e "calibrate" (le funzioni sono ben commentate all'interno di "INA219.h" e "INA219.cpp");

Table 5. ADC Settings⁽¹⁾

ADC4	ADC3	ADC2	ADC1	Mode/Samples	Conversion Time
0	X ⁽²⁾	0	0	9 bit	84 μs
0	X ⁽²⁾	0	1	10 bit	148 μs
0	X ⁽²⁾	1	0	11 bit	276 μs
0	X ⁽²⁾	1	1	12 bit	532 μs
1	0	0	0	12 bit	532 μs
1	0	0	1	2	1.06 ms
1	0	1	0	4	2.13 ms
1	0	1	1	8	4.26 ms
1	1	0	0	16	8.51 ms
1	1	0	1	32	17.02 ms
1	1	1	0	64	34.05 ms
1	1	1	1	128	68.10 ms

(1) Shaded values are default.

(2) X = Don't care

Tabella 1
Configurazione
dell'ADC
dell'INA219.



▼ Listato 1

```
/* *****  
 * INA219 library example  
 * 9 January 2016 by Flavius Bindea  
 *  
 * this code is public domain.  
 * ***** */  
  
#include <Wire.h>  
#include <INA219.h>  
  
#define SHUNT_MAX_V 0.04 /* Rated max for our shunt is 75mv for 50 A current:  
                          we will mesure only up to 20A so max is about 75mV*20/50 lets put some more*/  
#define BUS_MAX_V 16.0 /* with 12v lead acid battery this should be enough*/  
#define MAX_CURRENT 20 /* In our case this is enough even shunt is capable to 50 A*/  
#define SHUNT_R 0.015 /* Shunt resistor in ohm */  
  
INA219 monitor;  
  
void setup()  
{  
  Serial.begin(57600);  
  monitor.begin();  
  // setting up our configuration  
  // default values are RANGE_32V, GAIN_8_320MV, ADC_12BIT, ADC_12BIT, CONT_SH_BUS  
  monitor.configure(INA219::RANGE_16V, INA219::GAIN_2_80MV, INA219::ADC_64SAMP, INA219::ADC_64SAMP, INA219  
  ::CONT_SH_BUS);  
  
  // calibrate with our values  
  monitor.calibrate(SHUNT_R, SHUNT_MAX_V, BUS_MAX_V, MAX_CURRENT);  
}  
  
void loop()  
{  
  Serial.println("*****");  
  
  Serial.print("raw shunt voltage: ");  
  Serial.println(monitor.shuntVoltageRaw());  
  
  Serial.print("raw bus voltage: ");  
  Serial.println(monitor.busVoltageRaw());  
  
  Serial.println("--");  
  
  Serial.print("shunt voltage: ");  
  Serial.print(monitor.shuntVoltage() * 1000, 4);  
  Serial.println(" mV");  
  
  Serial.print("shunt current: ");  
  Serial.print(monitor.shuntCurrent() * 1000, 4);  
  Serial.println(" mA");  
  
  Serial.print("bus voltage: ");  
  Serial.print(monitor.busVoltage(), 4);  
  Serial.println(" V");  
  
  Serial.print("bus power: ");  
  Serial.print(monitor.busPower() * 1000, 4);  
  Serial.println(" mW");  
  
  Serial.println(" ");  
  Serial.println(" ");  
  
  delay(1000);  
}
```

- "begin()" inizializza la comunicazione con l'INA219;
- "configure" configura il modo di funzionamento dell'INA.

Gli argomenti della funzione sono range, gain, bus_adc, shunt_adc, mode:

- range: impostazione del range di tensione del bus



Table 6. Mode Settings⁽¹⁾

MODE3	MODE2	MODE1	MODE
0	0	0	Power-down
0	0	1	Shunt voltage, triggered
0	1	0	Bus voltage, triggered
0	1	1	Shunt and bus, triggered
1	0	0	ADC off (disabled)
1	0	1	Shunt voltage, continuous
1	1	0	Bus voltage, continuous
1	1	1	Shunt and bus, continuous

(1) Shaded values are default.

- RANGE_16V : Range 0-16V
- RANGE_32V (default): Range 0-32V

- **gain:** impostazione del guadagno e del relativo valore massimo della tensione di shunt
- GAIN_1_40MV: 40mV
- GAIN_2_80MV: 80mV
- GAIN_4_160MV: 160mV
- GAIN_8_320MV (default): 320mV

Per maggiori dettagli relativi alla configurazione dell'ADC che segue, fate riferimento alla **Tabella 1:**

- **bus_adc:** configurazione del campionamento dell'ADC relativamente al bus e numero di campioni utilizzati per calcolare il valor medio
- ADC_9BIT: 9bit, tempo di conversione 84us.
- ADC_10BIT: 10bit, tempo di conversione 148us.
- ADC_11BIT: 11bit, tempo di conversione 276us.
- ADC_12BIT (default): 12bit tempo di conversione 532us.
- ADC_25SAMP: 2 campioni, tempo di conversione 1.06ms.
- ADC_4SAMP : 4 campioni, tempo di conversione 2.13ms.
- ADC_8SAMP : 8 campioni, tempo di conversione 4.26ms.
- ADC_16SAMP: 16 campioni, tempo di conversione 8.51ms
- ADC_32SAMP: 32 campioni, tempo di conversione 17.02ms.
- ADC_64SAMP: 64 campioni, tempo di conversione 34.05ms.
- ADC_128SAMP: 128 campioni, tempo di conversione 68.10ms.
- **shunt_adc:** configurazione del campionamento dell'ADC relativamente allo shunt e numero di campioni utilizzati per calcolare il valor medio (stessi valori del bus_adc)
- **mode:** impostazione del modo di funzionamento
- PWR_DOWN : spegnimento
- ADC_OFF
- CONT_SH : shunt continuo

- CONT_BUS : bus continuo
- CONT_SH_BUS (default): shunt e bus continuo.

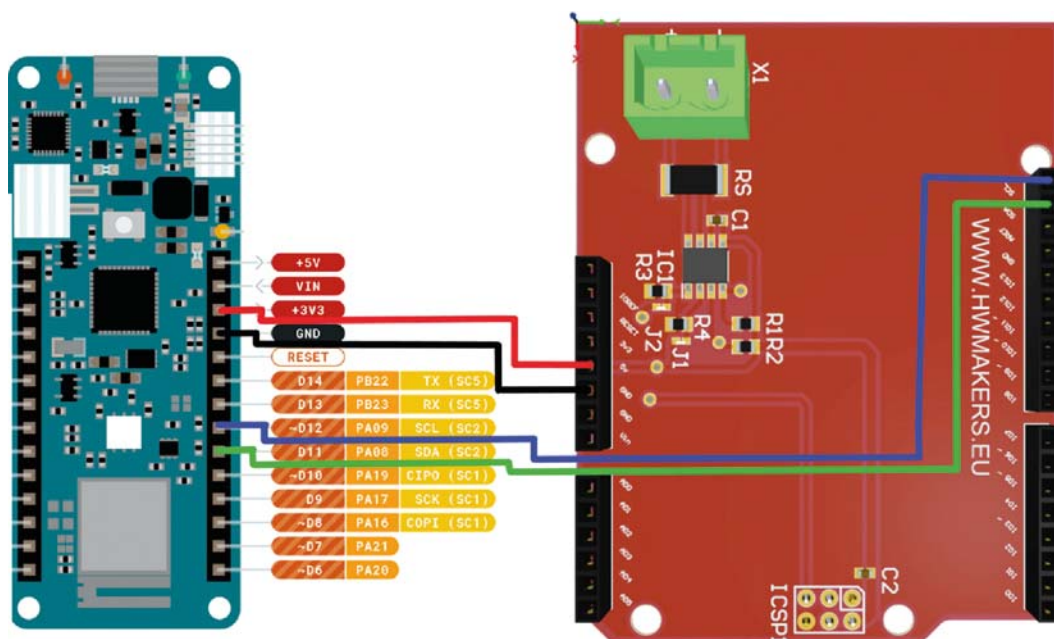
Per maggiori dettagli inerenti le modalità di configurazione, si può fare riferimento alla **Tabella 2:**

- **calibrate(r_shunt, v_shunt_max, v_bus_max, i_max_expected)** funzione che effettua i calcoli come indicato nel datasheet e inizializza i registri di calibrazione; gli argomenti sono:
- **r_shunt:** valore del resistore di shunt in ohm
- **v_shunt_max:** massimo valore della tensione ai capi dello shunt
- **v_bus_max:** massima tensione del bus
- **i_max_expected:** massima corrente assorbita dal bus e dallo shunt
- **reconfig()** riscrive l'ultima configurazione utilizzata dell'INA219
- **recalibrate()** riscrive i registri di calibrazione come sono stati calcolati dalla funzione **calibrate()**
- **ready()** restituisce il valore del bit "ready"; il valore è aggiornato in accordo all'ultima chiamata a **busVoltageRaw()** o **busVoltage()**
- **overflow()** restituisce il valore del bit di overflow. Il suo valore è aggiornato in accordo all'ultima chiamata a **busVoltageRaw()** o **busVoltage()**
- **reset():** resetta l'INA219.
- **shuntVoltageRaw():** restituisce il valore raw binario della tensione ai capi dello shunt
- **busVoltageRaw():** restituisce il valore raw binario della tensione del bus
- **shuntCurrentRaw() :** restituisce il valore raw binario della corrente circolante nel bus
- **shuntVoltage():** restituisce la tensione dello shunt in volt.
- **busVoltage():** restituisce la tensione del bus in volt.
- **shuntCurrent():** restituisce la corrente circolante nello shunt in ampere
- **busPower():** restituisce la potenza assorbita dal bus in watt

Tabella 2
Modalità di
configurazione
dell'INA219.



Fig. 3
Esempio di
connessione
dell'arduino
MKR1010 WIFI allo
shield non isolato.



Questa libreria non implementa la modalità di funzionamento "triggerata" ma soltanto quella continua.

Una volta noti i dettagli di come è stata implementata la libreria, è facile realizzare un proprio sketch di test oppure comprendere gli sketch di esempio forniti insieme alla libreria e raggiungibili o dal link <https://github.com/flav1972/ArduinoINA219/tree/master/examples>, oppure direttamente cercando tra gli esempi presenti nell'IDE di arduino, visibili dopo aver installato correttamente la libreria "ArduinoINA219". Iniziamo dal più facile, "ina219_test_nondefault.ino" (**Listato 1**). Come vedete, l'utilizzo dello shield tramite la libreria è veramente molto semplice: una volta inclusa la libreria "Wire" per la comunicazione I²C e "INA219" per la gestione dell'integrato, si definisce il valore della tensione ai capi della resistenza di shunt in volt (`#define SHUNT_MAX_V`), la massima tensione prevista sul bus sempre in volt (`#define BUS_MAX_V`), il massimo valore della corrente circolante nel resistore di shunt in ampere (`#define MAX_CURRENT`) ed infine il valore della resistenza di shunt in ohm (`#define SHUNT_R`). Fatto questo, si inizializza l'INA219 nella funzione Setup() tramite "monitor.begin()" e se ne configura la modalità di funzionamento, impostando i parametri visti precedentemente tramite la riga di codice "monitor.configure (INA219::RANGE_16V, INA219::GAIN_2_80MV, INA219::ADC_64SAMP, INA219::ADC_64SAMP, INA219::CONT_SH_BUS);" come segue:

- range di tensione del bus pari a 16V (`"INA219::RANGE_16V"`);

- impostazione del guadagno e del relativo valore massimo della tensione di shunt a 2, 80mV (`"INA219::GAIN_2_80MV"`);
- configurazione dell'ADC per la misura della tensione sul bus, utilizzando 64 campioni per il calcolo del valor medio (`INA219::ADC_64SAMP`);
- configurazione dell'ADC per il calcolo della tensione di shunt, utilizzando 64 campioni per il calcolo del valor medio (`INA219::ADC_64SAMP`);
- modalità di funzionamento di tipo "continuo" dell'integrato (`INA219::CONT_SH_BUS`) (che è comunque l'unica possibile per questa libreria, perlomeno nel momento in cui scriviamo).

Tramite l'istruzione `monitor.calibrate(SHUNT_R, SHUNT_MAX_V, BUS_MAX_V, MAX_CURRENT)` vengono richiamati i valori visti in precedenza, che definiscono operativamente le grandezze coinvolte nella misura, quali appunto il valore di resistenza dello shunt, la tensione massima prevista ai suoi capi, la massima tensione sul bus e la massima corrente in esso circolante. All'interno del loop sono presenti i vari comandi per l'invio sulla seriale (`"Serial.print"`) dei valori di tensione e di corrente da misurare, compresi i valori "RAW" (quantizzati) misurati dall'ADC.

UTILIZZO CON ARDUINO MKR

Data la peculiarità dell'INA219 di contenere tutto il front-end analogico, il convertitore analogico digitale, l'interfaccia I²C/SMBus compatibile sulla quale fornisce direttamente i valori di tensione e di corrente misurati, questo tipo di current sense

è particolarmente adatto per l'uso con microcontrollori dalle risorse limitate, quindi è ottimo per essere utilizzato con Arduino UNO, Leonardo e tutte le versioni con MCU ad 8 bit; nulla toglie che possa essere utilizzato anche con microcontrollori a 32 bit, come ad esempio i prodotti della serie arduino MKR, oppure con schede MBED, che presentano, tra le tante, due differenze importanti: lavorano con una tensione di alimentazione del microcontrollore a 3,3V invece di 5V e hanno una forma differente da quella degli shield standard per arduino UNO e schede simili o da questa derivate. Questo potrebbe indurre a pensare che non sia possibile utilizzare i nostri shield con queste schede di sviluppo, ma non è così e adesso vi spieghiamo quali accorgimenti servono.

Chiaramente per ovviare al problema della geometria sarebbe necessario disegnare un nuovo shield meccanicamente compatibile con arduino MKR o con le board MBED o comunque con un'altra scheda di sviluppo a microcontrollore che si volesse utilizzare, che avesse anch'essa una forma diversa da quella di arduino UNO; per chi si accontentasse di utilizzare il PCB con INA219 senza montarlo direttamente sopra la scheda con MCU a 32 bit, può farlo tenendo conto di un importante accorgimento che deve assolutamente essere preso: essendo il circuito nato per essere montato su arduino uno, presenta un'alimentazione del bus I2C a 5V, anche la versione isolata, perlomeno dal lato dell'arduino, mentre i microcontrollori a 32 bit generalmente utilizzati per le schede di sviluppo devono essere

alimentati a 3,3V. Alcuni microcontrollori a 32 bit con alimentazione a 3,3V hanno gli ingressi "5V tolerant" ma comunque sconsigliamo anche in questi casi di lavorare con il bus I²C a 5V. Si può ovviare al problema semplicemente collegando il terminale "5V" dello shield alla tensione 3,3V della scheda di sviluppo. In **Fig. 3** è mostrato un semplice esempio che mostra come collegare un'Arduino MKR1010 WIFI allo shield non isolato; le connessioni all'Arduino rimangono le stesse anche per quanto riguarda lo shield isolato, per il quale cambia solo la connessione al carico, come abbiamo visto in precedenza. Se si volesse procedere, si raccomanda la massima attenzione perché un errore nelle connessioni dell'alimentazione potrebbe causare il danneggiamento dell'Arduino MKR.

Nel caso si utilizzino microcontrollori della famiglia SAMD come nel caso dell'arduino MKR 1010 WIFI, è necessario modificare la libreria, in quanto questa non ne prevede l'uso ed utilizza ancora la vecchia istruzione "_delay_ms()" invece di delay(), cosa che porta ad errori durante la compilazione della stessa (la funzione "delay()" è compatibile con gli Atmel sia ad 8 che a 32 bit).

INVIAMO LE MISURE NEL CLOUD

Adesso che abbiamo visto come utilizzare l'arduino MKR 1010 WIFI con i nostri shield di tipo isolato e non isolato che montano l'integrato della Texas Instruments INA219, vediamo come si può trasmettere la grandezza misurata nel cloud, utilizzando ad esempio il servizio offerto, anche in versione gratuita, da Arduino, chiamato arduino IOT cloud, raggiungibile tramite il link <https://create.arduino.cc/iot/>. Vogliamo ad esempio realizzare un semplice sistema IOT, in cui alimentiamo il nostro

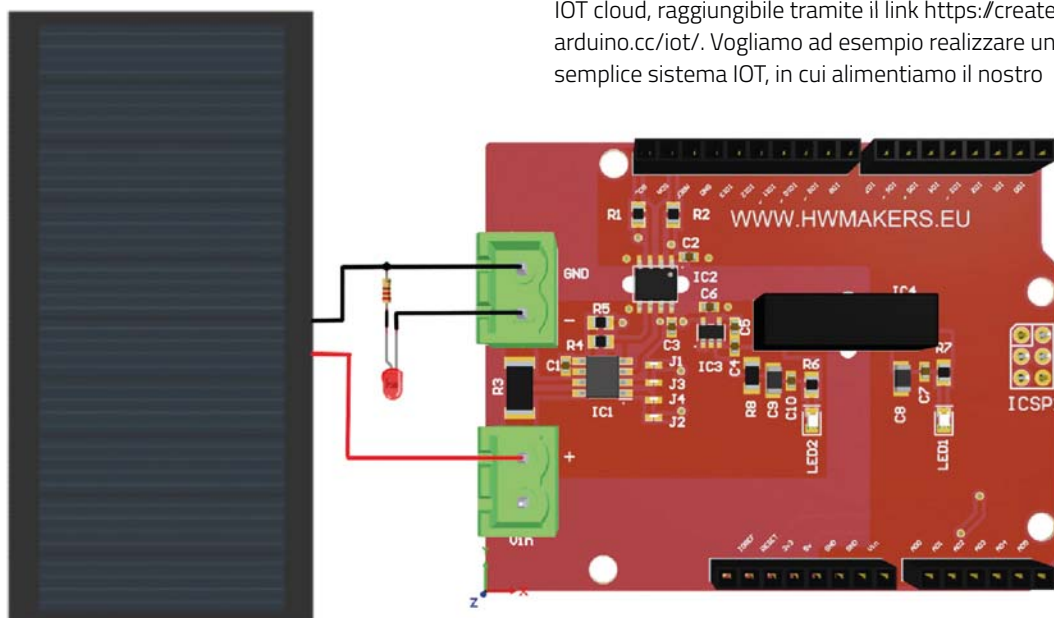


Fig. 4
esempio di
connessione
dello shield
isolato ad
un pannello
fotovoltaico.

Fig. 5
Registrazione di
un nuovo account
Arduino.

Sign up to Arduino

Enter your email *

Choose a username *

Choose a password *
Minimum 8 characters long

☐ I have read the [Privacy Policy](#) and accept the [Terms of Service](#) *

☐ I would like to receive the Arduino newsletter.

☐ I agree to see personalised commercial offers from Arduino based on my browsing and purchasing behaviour.

☐ I would like to get email updates about special deals and commercial offers from Arduino.

SIGN UP

arduino tramite un piccolo pannello solare di cui ci interessa monitorare continuamente la corrente erogata (**Fig. 4**). Volendo, è possibile monitorare anche altre grandezze, come la tensione in uscita dal pannello e la potenza erogata, per le cui configurazioni si potrà procedere in modo analogo a quanto vedremo tra breve. Se non avete già creato un account sul sito di arduino in precedenza, la prima cosa da fare è ovviamente crearne uno, inserendo la propria email e scegliendo username e password (**Fig. 5**). Una guida ufficiale completa

Fig. 6
Creazione e
definizione di una
variabile all'interno
dell'Arduino Cloud
IoT.

Add variable

Name
current

[Sync with other Things](#)

Floating Point Number eg. 1.55

Declaration
float current;

Variable Permission

☒ Read & Write ☐ Read Only

Variable Update Policy

☐ On change ☒ Periodically

all'uso di questo cloud è disponibile in inglese alla pagina web

<https://docs.arduino.cc/cloud/iot-cloud/tutorials/iot-cloud-getting-started>.

Una volta entrati nel proprio account si può selezionare IOT cloud nel menù a tendina in alto a destra, di fianco all'icona dell'omino, rappresentante il nostro account, oppure cliccare sul riquadro ovale con la scritta "ARDUINO IOT CLOUD". Fatto ciò si selezionerà "Devices" e si creerà un nuovo oggetto ("Things", "Setup" e "ADD VARIABLE") per poi definire il nome della variabile, nel nostro caso "current", che stabiliremo sarà di tipo "float" ("floating point number"), che avrà i permessi di lettura e scrittura e che verrà aggiornata periodicamente, ogni secondo (**Fig. 6**). Fatto questo, è necessario aggiungere un nuovo dispositivo oppure associarne uno già esistente all'oggetto creato, selezionando, sempre in "Setup", "Device", "Select Device". In **Fig. 7** è visibile un esempio di un dispositivo già esistente, un arduino MKR 1010 WIFI e come si presenta lo stesso una volta associato alla grandezza da monitorare, nel nostro caso la corrente "current". Se invece non si è ancora associata nessuna scheda, perché magari è la prima volta che utilizzate questo cloud, bisogna selezionare "Devices" nel menù in alto sullo schermo, "ADD DEVICE" e una volta aperta la finestra visibile in **Fig. 8**, basterà selezionare "Set up an Arduino device" dovendo collegare un arduino MKR 1010 ufficiale. NOTA: nel caso in cui si utilizzi l'Arduino IOT cloud per la prima volta, nel momento in cui si voglia configurare un nuovo dispositivo, verrà richiesto di installare l'Arduino Agent, necessario per portare a termine con successo la procedura.

Si seguono poi le indicazioni riportate sullo schermo fino al termine dell'installazione, durante la quale verrà anche configurato l'integrato per la crittografia presente sul nostro MKR, utilizzato per rendere più sicura la comunicazione con il cloud. Possiamo ora visualizzare in tempo reale la corrente generata dal nostro piccolo pannello fotovoltaico e circolante nel carico, un LED, creando nella dashboard i widget che vogliamo, scegliendo "add" e poi "widget" se vogliamo conferire ai nostri dati l'aspetto grafico che più ci aggrada, oppure semplicemente "add" "things" se siamo interessati solamente alla visualizzazione numerica della grandezza inviata nel cloud (**Fig. 9**):

Il team di arduino ha realizzato un'applicazione per dispositivi mobili sia Android sia iOS disponibile nei rispettivi store, "IOT remote", con la quale è possibile monitorare le grandezze inviate nel cloud

dai nostri dispositivi IOT. Chiaramente quanto visto, per funzionare, richiede anche di caricare un nuovo sketch nel nostro arduino, che permetta di comunicare correttamente con il servizio arduino IOT cloud: sempre nei nostri oggetti "Things" è presente il link all'editor web ("Sketch") che ci consente di scrivere il codice per la nostra scheda di sviluppo. Una volta aperto, vediamo che, oltre a "CURRENT_sep14.ino" ci sono anche "ReadMe.doc", "thingProperties.h" e "Secret" (Fig. 10)

Nel **Listato2** proponiamo il codice "current.ino" per inviare nel cloud il valore di corrente misurato. Come si vede le modifiche rispetto alla versione vista precedentemente sono veramente minime e consistono essenzialmente nella presenza di "#include "arduino_secrets.h"" che permette di impostare il nome della rete wifi a cui ci si vuole connettere tramite "SECRET_SSID" e "SECRET_PASS", entrambi definiti all'interno di "SECRET" e "#include "thingProperties.h" in cui sono definiti i parametri di accesso alla rete e al cloud e le caratteristiche della grandezza (THING) che si vuole monitorare da remoto tramite il cloud, nel nostro caso definita tramite la variabile "float current;"

Il file "thingProperties.h" viene generato automaticamente dall'arduino IOT cloud una volta che si sono impostati e configurati correttamente tutti i parametri del proprio hardware (il proprio modello specifico di arduino a cui viene dato uno specifico numero identificativo) e la grandezza da trasmettere nel cloud.

Nel **listato 3**, è presente il codice di "thingProperties.h".

REALIZZAZIONE PRATICA

La realizzazione pratica dei due shield non è molto complessa, ed è alla portata di tutti gli hobbisti che abbiano dimestichezza con l'uso del saldatore e con la saldatura di componenti a montaggio superficiale di dimensione non inferiore a 0603. Per quanto riguarda l'INA 219, disponibile sia in package SOIC8 sia in versione SOT23-6, nel nostro progetto è stato scelto con il package più

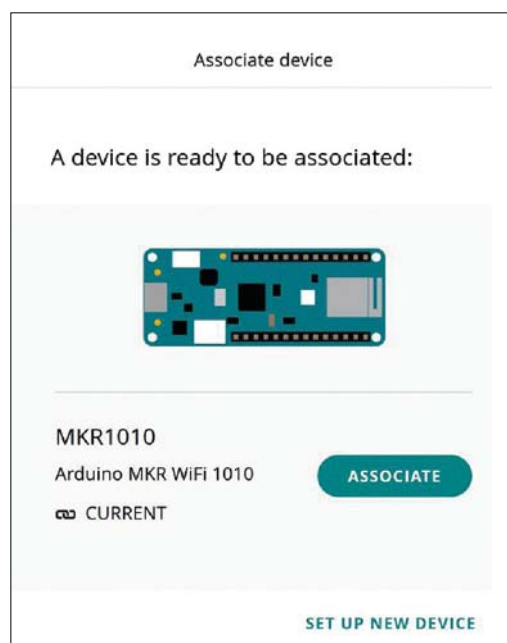


Fig. 7
Esempio di associazione di un dispositivo esistente alla grandezza da monitorare.

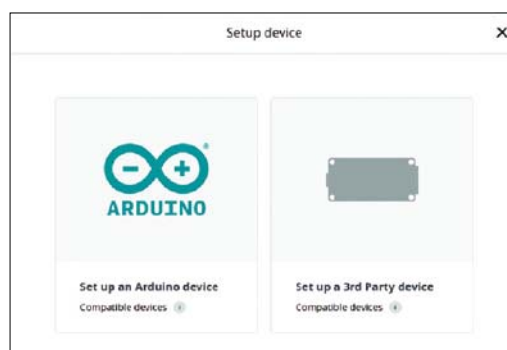


Fig. 8
Configurazione di un nuovo dispositivo.

grande dei due, quindi SOIC8, la cui saldatura è alla portata di chiunque abbia un po' di esperienza. Per la realizzazione degli shield si parte dai due circuiti stampati, ottenibili per fotoincisione; sul nostro sito sono disponibili i file gerber insieme agli altri file del progetto. Una volta in possesso del circuito stampato, si inizia saldando i componenti dal profilo più basso, cioè resistenze, condensatori ed i due led, rispettando per questi ultimi il corretto

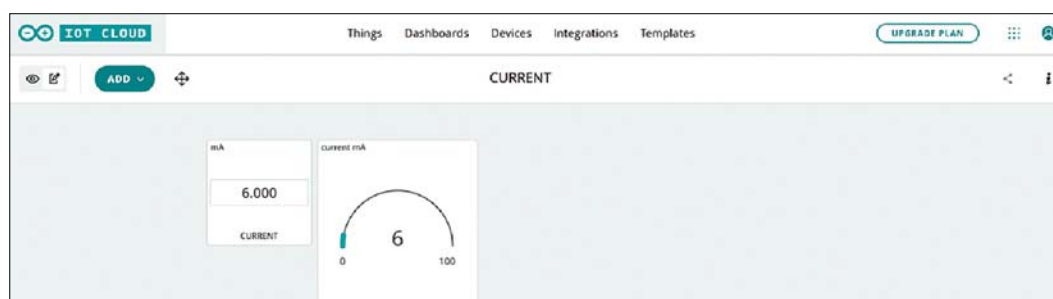


Fig. 9
Misura della corrente generata dal nostro pannello, in tempo reale, visualizzata nella nostra dashboard.

↓ Listato 2

```
// ArduinoINA219 - Version: Latest
#include "arduino_secrets.h"
#include "thingProperties.h"
#include <Wire.h>
#include <INA219.h>
#define SHUNT_MAX_V 0.05 /* shunt maximum voltage*/
#define BUS_MAX_V 16.0 /* set maximum bus voltage to 16v*/
#define MAX_CURRENT 5 /* set maximum current to 5A*/
#define SHUNT_R 0.01 /* Shunt resistor in ohm */
INA219 monitor;
void setup()
{
  Serial.begin(57600);
  monitor.begin();
  // setting up our configuration
  // default values are RANGE_32V, GAIN_8_320MV, ADC_12BIT, ADC_12BIT, CONT_SH_BUS
  monitor.configure(INA219::RANGE_16V, INA219::GAIN_2_80MV, INA219::ADC_64SAMP, INA219::ADC_64SAMP,
  INA219::CONT_SH_BUS);
  // calibrate with our values
  monitor.calibrate(SHUNT_R, SHUNT_MAX_V, BUS_MAX_V, MAX_CURRENT);
  // Defined in thingProperties.h
  initProperties();
  // Connect to Arduino IoT Cloud
  ArduinoCloud.begin(ArduinoIoTPreferredConnection);
  setDebugMessageLevel(2);
  ArduinoCloud.printDebugInfo();
}
void loop()
{
  ArduinoCloud.update();
  current=monitor.shuntCurrent()* 1000;
  Serial.println("*****");
  Serial.print("raw shunt voltage: ");
  Serial.println(monitor.shuntVoltageRaw());
  Serial.print("raw bus voltage: ");
  Serial.println(monitor.busVoltageRaw());
  Serial.println("--");
  Serial.print("shunt voltage: ");
  Serial.print(monitor.shuntVoltage() * 1000, 4);
  Serial.println(" mV");
  Serial.print("shunt current: ");
  Serial.print(monitor.shuntCurrent() * 1000, 4);
  Serial.println(" mA");
  Serial.print("bus voltage: ");
  Serial.print(monitor.busVoltage(), 4);
  Serial.println(" V");
  Serial.print("bus power: ");
  Serial.print(monitor.busPower() * 1000, 4);
  Serial.println(" mW");

  Serial.println(" ");
  Serial.println(" ");
  delay(1000);
}
```

↓ Listato 3

```
// Code generated by Arduino IoT Cloud, DO NOT EDIT.
#include <ArduinoIoTCloud.h>
#include <Arduino_ConnectionHandler.h>
const char THING_ID[] = ""; //HW ID CODE HIDDEN HERE FOR SECURITY REASONS
const char SSID[] = SECRET_SSID; // Network SSID (name)
const char PASS[] = SECRET_PASS; // Network password (use for WPA, or use as key for WEP)
void onCurrentChange();
float current;
void initProperties(){
  ArduinoCloud.setThingId(THING_ID);
  ArduinoCloud.addProperty(current, READWRITE, 1 * SECONDS, onCurrentChange);
}
WiFiConnectionHandler ArduinoIoTPreferredConnection(SSID, PASS);
```



✓

→

Arduino MKR WIFI 1010

...

GO TO IOT CLOUD

CURRENT_sep14a.ino

ReadMe.adoc

thingProperties.h

Secret

```

1 // ArduinoINA219 - Version: Latest
2
3
4 #include "thingProperties.h"
5 #include <Wire.h>
6 #include <INA219.h>
7
8
9 #define SHUNT_MAX_V 0.05 /* Rated max for our shunt is 5mv for 5 A current:
10                          we will mesure only up to 20A so max is about 75mV*20/50 lets put
11 #define BUS_MAX_V 16.0 /* with 12v lead acid battery this should be enough*/
12 #define MAX_CURRENT 5 /* In our case this is enough even shunt is capable to 5A*/
13 #define SHUNT_R 0.01 /* Shunt resistor in ohm */
14
15 INA219 monitor;
16
17 void setup()
18 {
19   Serial.begin(57600);
20   monitor.begin();
21   // setting up our configuration
22   // default values are RANGE_32V, GAIN_8_320MV, ADC_12BIT, ADC_12BIT, CONT_SH_BUS
23   monitor.configure(INA219::RANGE_16V, INA219::GAIN_2_80MV, INA219::ADC_64SAMP, INA219::ADC_64
24
25   // calibrate with our values

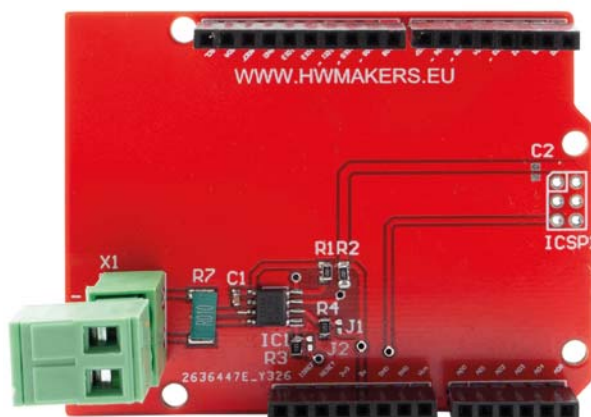
```

Fig. 10
Ecco come si presenta la pagina del web editor di arduino online con il nostro sketch pronto per essere caricato.

orientamento, come indicato nel piano di montaggio, facendo attenzione a non indugiare troppo con il saldatore per non danneggiarne la delicata

struttura che lo ricopre. Si passa poi alla saldatura dei circuiti integrati a montaggio superficiale, per poi proseguire con i due connettori a passo

shield NON ISOLATO



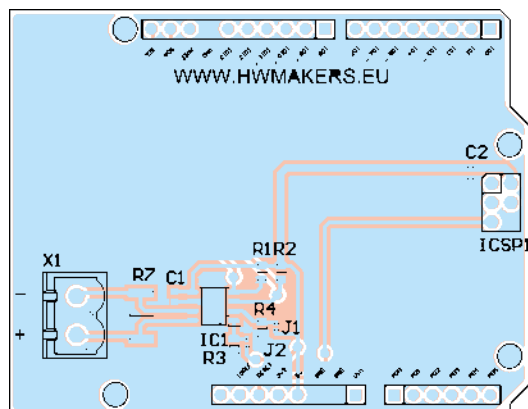
Elenco Componenti:

R1, R2: 4,7 Kohm (0603)
R3, R4: 0 ohm (0603)
RS: vedi testo
C1, C2: 100nF (0603)

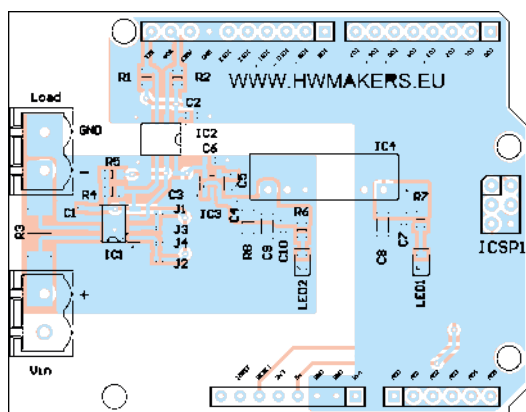
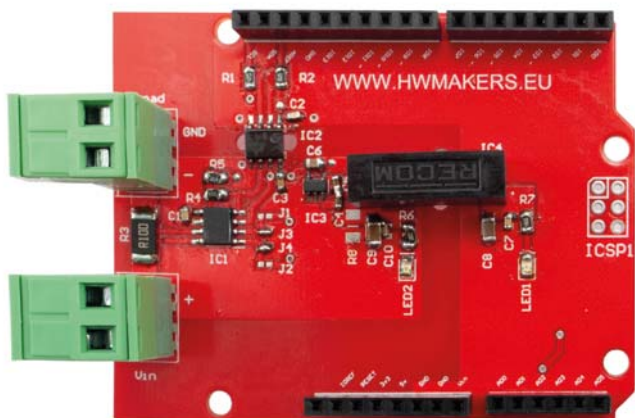
IC1: INA219(A)(B)IDR

Varie:

- morsetto componibile passo 5,08mm 2 posizioni



- header arduino 10 poli
- header arduino 8 poli
- header arduino 6 poli
- Stampato (70x59)



Elenco Componenti:

R1, R2, R4, R5: 4K7 0603
R3: vedi testo
R6, R7: 680 0805
R8: vedi testo
C1, C2, C3, C7, C10: 100nF 0603
C4, C6: 1uF 50V 0603
C5: 10nF 0603
C8: 10uF 16V 1206
C9: 22uF 16V 1206
IC1: INA219(A)(B)IDR
IC2: ISO1640BDR
IC3: LP3985IM5-3.3/TPS79933
IC4: REM1-0505S o equivalente
LED1, LED2: LED verde 0805

Varie:

- morsetto componibile passo 5.08mm 2 posizioni
- header arduino 10 poli
- header arduino 8 poli
- header arduino 6 poli
- Stampato (70x59)

5.08mm, seguiti dagli strip femmina a pin lunghi che costituiscono gli header di Arduino; a questo punto possiamo montare il convertitore isolato non regolato da 5V in package SIP7. Per la saldatura si consiglia l'utilizzo di un saldatore isolato dalla tensione di rete tramite trasformatore, con una punta piccola, mentre per il posizionamento dei componenti è necessario, se non indispensabile, usare un'apposita pinzetta antistatica in metallo; non è invece indispensabile, anche se consigliato, l'utilizzo di flussante di buona qualità. Per aiutarsi nel posizionamento dei componenti e nella saldatura degli stessi, consigliamo l'utilizzo di una lente con luce integrata, oppure un piccolo microscopio con ingrandimento adatto alla saldatura dei componenti elettronici. Infine si consiglia la pulizia del flussante in eccesso, presente comunque anche nella lega saldante, tramite gli appositi solventi o l'acetone.

CONCLUSIONI

In questo articolo vi abbiamo presentato un integrato della Texas Instruments, l'INA219, che ci ha permesso di realizzare facilmente un completo sistema di monitoraggio di tensione corrente e potenza di un carico, associato ad Arduino ed eventualmente connesso al servizio arduino IOT cloud per poter monitorare il carico da remoto. Nonostante gli shield siano stati pensati per soddisfare le caratteristiche elettriche e geometriche standard di Arduino UNO e compatibili, vi abbiamo spiegato come sia possibile collegare entrambi gli shield anche ad un arduino MKR 1010 WIFI, che presenta una forma diversa ed un microcontrollore atmel SAMD a 32 bit, funzionante con un alimentazione a 3,3V. Ora che avete tutti gli elementi per lavorarci, iniziate pure a compiere le vostre misure con lo shield che più si adatta alle vostre necessità. □



Cosa occorre?

I componenti utilizzati in questo progetto sono di facile reperibilità.

I gerber delle shield per realizzare i PCB sono scaricabili dal sito della rivista alla pagina del numero 260.

I circuiti stampati possono essere realizzati con il servizio di stampa PCB dal sito www.futurashop.it/produzionePCB

Il materiale va richiesto a:

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

Taglia e incidi con Cricut[™] MAKER



Prezzo IVA inclusa

Macchina da taglio intelligente



€ 439,00

cod. CRICUTMAKER

Dai vita alla fantasia e alla creatività con Cricut Maker l'innovativa macchina da taglio. Grazie a Design Space, il software di progettazione gratuito, potrai realizzare praticamente qualsiasi progetto fai-da-te, dall'arte 3D all'arredamento per la casa, gioielli, progetti di carta e molto altro ancora e inviarti direttamente a Cricut Maker dal tuo tablet, dal PC o dallo smartphone. Cricut Maker taglia in modo rapido e accurato oltre 300 materiali: tessuti, pelle, carta, cartamodelli, cuoio, fogli di balsa e tanto altro. Espandi le potenzialità di Cricut Maker utilizzando utensili dedicati, alcuni inclusi nella confezione, altri acquistabili separatamente.



Scopri il mondo di ARDUINO e le sue infinite potenzialità!

Progetta, costruisci e divertiti con i moduli per Arduino predisposti per essere inseriti sui mattoncini della LEGO®. Puoi utilizzarli per robotica, con Arduino o Raspberry Pi, ecc. Nella confezione sono comprese, oltre ai sensori e agli accessori, anche una board Arduino Uno compatibile e una scheda Shield sperimentale dove puoi collegare in tutta semplicità i moduli sensori e attuatori. Una dispensa a corredo vi accompagna con gli sketch per testare i vari moduli con Arduino e la Sensor Shield.

*29 MODULI PER ARDUINO
COMPATIBILI CON LEGO®*

€ 79,00

Cod. ARDULEGOKIT

I MODULI PRESENTI NELLA CONFEZIONE:

- Modulo con display 7 segmenti a 4 cifre
- Modulo con sensore di fiamma
- Modulo con LED RGB 10 mm
- Modulo sensore di temperatura e umidità
- Modulo touch sensor
- Scheda sensore umidità terreno
- Scheda sensore livello liquidi
- Modulo con LED rosso
- Modulo con LED verde
- Modulo con LED giallo
- Modulo con sensore effetto di Hall
- Modulo con LED RGB SMD
- Modulo con sensore luce ambiente
- Modulo con pulsante rosso
- Modulo con pulsante giallo
- Modulo con interruttore a pulsante
- Modulo con buzzer senza elettronica
- Modulo sensore di vibrazione
- Modulo con sensore di temperatura
- Modulo con potenziometro
- Modulo con fotoresistenza
- Modulo con interruttore reed
- Modulo con ricevitore IR a 3 pin
- Modulo con buzzer con elettronica
- Modulo con encoder rotativo
- Modulo sensore scala di grigi
- Modulo con microswitch

Prezzi IVA inclusa



Starter kit V6 con Arduino Uno REV3



**IL PIÙ
COMPLETO!**

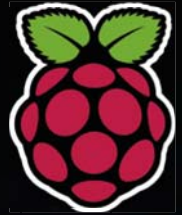
€ 65,00

Cod. ARDUKITV6

Acquista anche il kit
completo con tutti i
componenti necessari per
realizzare gli esperimenti
descritti nel libro "l'ABC
di ARDUINO".

ANIMAZIONI NATALIZIE SU PANNELLI LED

di EMANUELE SIGNORETTA



Creiamo un display luminoso con dei pannelli a matrice di LED, un Raspberry Pi e alcuni software per animare il nostro Natale.

Il Natale è sinonimo di festa ed è consuetudine ornare le abitazioni con ghirlande e luci per celebrare l'evento religioso più importante dell'anno. Le luminarie natalizie hanno subito tantissime evoluzioni nel corso dei decenni, dalle catene luminose con lampadine ad incandescenza, a quelle a led multicolor con suoneria integrata e giochi di luce, passando per un'infinità di altre varianti. Nel corso degli ultimi anni hanno iniziato a prendere piede le catene luminose composte da led indirizzabili che permettono di realizzare effetti di luce sempre più complessi. Come Elettronica In vogliamo proporvi un progetto ancora più evoluto che, usando

un Raspberry Pi e dei pannelli a matrice di LED, vi permetterà di animare la vostra festa stupendo grandi e piccini.

IL PROGETTO

Non è la prima volta che presentiamo un progetto con le matrici di LED; questi pannelli garantiscono un'ottima visibilità anche a distanza e la semplicità di utilizzo li rendono gestibili anche con board tipo Arduino o Fishino. Qualche anno fa (numero 208 e 209 di Elettronica In) abbiamo presentato LED Matrix: una scheda molto più performante di Arduino. LED Matrix è infatti basata su FPGA e garantisce una velocità di gestione ottimale per questo tipo di pannelli. Le animazioni venivano inviate alla board di controllo tramite PC e un apposito software (JINX), oppure potevano essere caricate su scheda SD e quindi eseguite in autonomia.

In questo progetto utilizziamo ancora la scheda LED Matrix, ma sostituiamo il PC con un Raspberry Pi, in modo tale da poter riconfigurare le animazioni tutte le volte che vorremo senza dover dedicare un computer appositamente al sistema.

L'HARDWARE

Per questo progetto utilizzeremo un Raspberry Pi4 da 8GB di RAM (disponibile sul sito www.futurashop.it con il codice RPI4-8GB), due

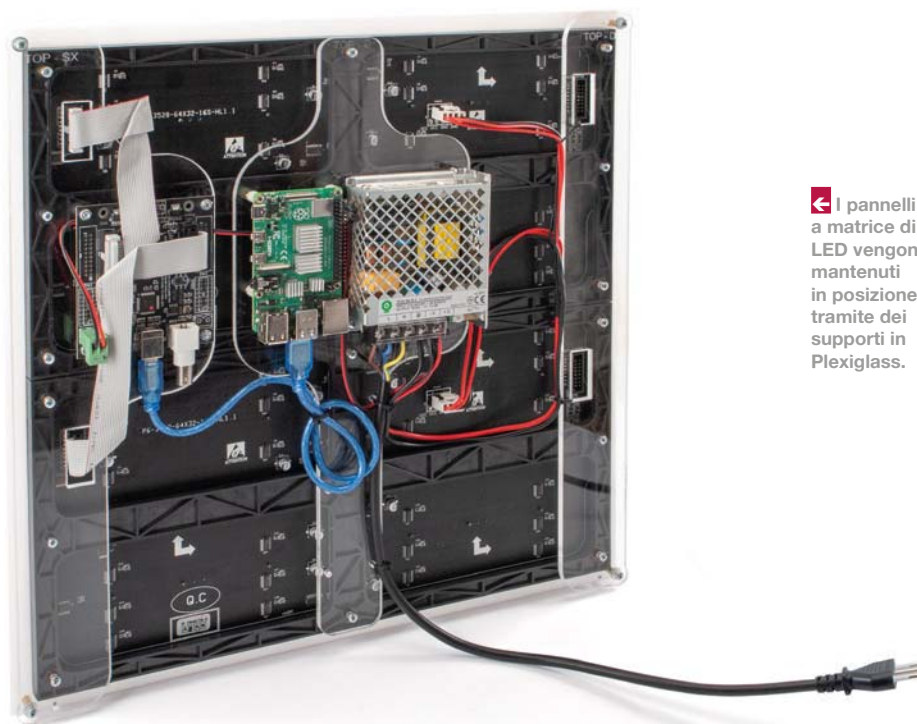
pannelli a matrici di led da 64x32 led (cod. RG-BLEDPANEL6432), una scheda di controllo LED Matrix (cod. FT1286M) e un alimentatore switching (cod. MW05005).

Nello schema di collegamento è riportato il cablaggio necessario per il corretto funzionamento.

IL SOFTWARE

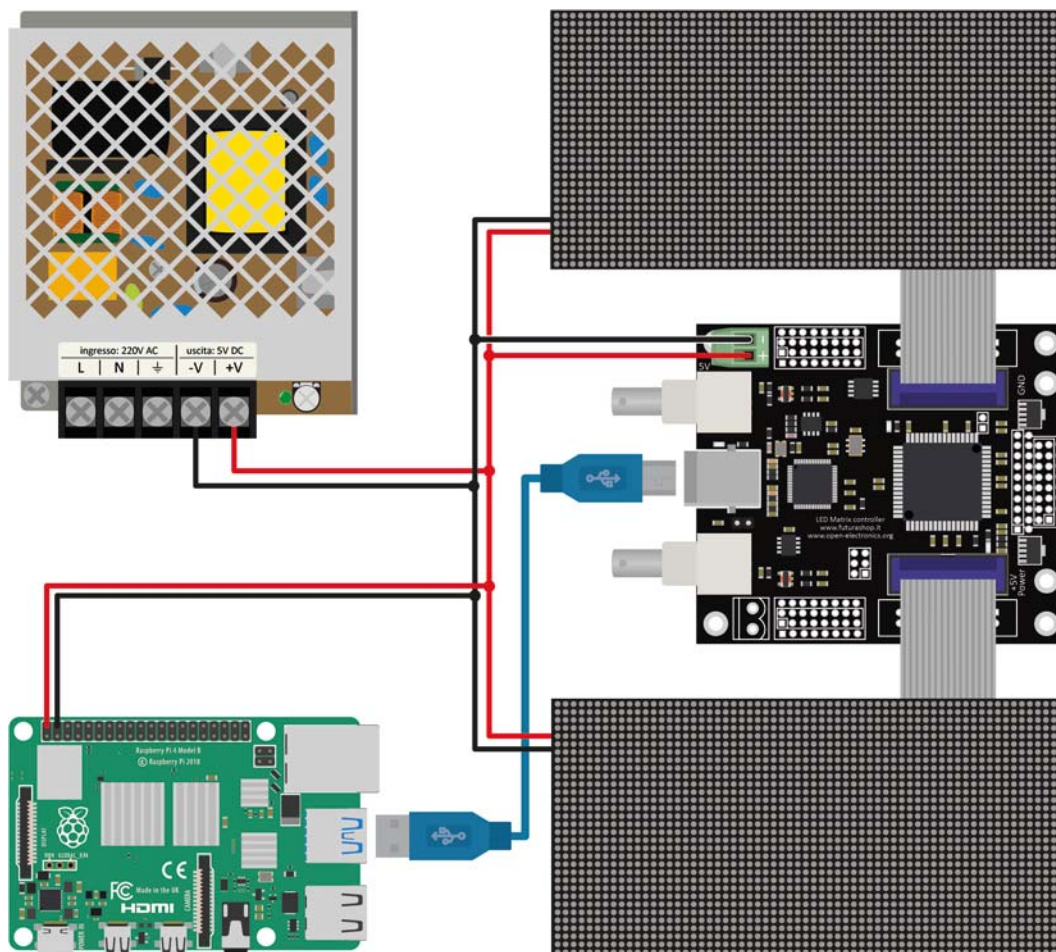
Il software principale su cui è basato il nostro progetto è **TwisterOS**.

TwisterOS è una distribuzione Linux, derivata del progetto Debian, che può essere installata su Raspberry Pi 4 e incorpora due interfacce grafiche molto simili a Windows 10 e quella di OSX Catalina. Ovviamente il motore base è sempre Debian, ma vengono forniti numerosi software presenti su Raspbian. La parte grafica di interfaccia è veramente ben curata tanto da confondersi con quelle dei concorrenti. Questa distribuzione offre numerosi software preinstallati, tra cui WineHQ, software open-source ampiamente utilizzato per eseguire software Windows su sistemi Linux. Per gestire e eseguire le animazioni useremo Glediator, un programma scritto in JAVA specifico per le matrici led e un tool scritto in Python per avviare automaticamente Glediator e il "demone" *systemd* per far partire lo script Python all'accensione del Raspberry Pi.



← I pannelli a matrice di LED vengono mantenuti in posizione tramite dei supporti in Plexiglass.

schema di CABLAGGIO



PREPARAZIONE SD E MODIFICA DEL BOOTLOADER

Questo progetto ben si adatta a miglioramenti e personalizzazioni, per questo motivo abbiamo deciso di predisporre già da adesso il Raspberry Pi 4 più potente, ovvero la versione da 8Gb. Per ottenere prestazioni ancora migliori, è possibile avviare il sistema operativo da un **SSD** invece che dall'**SD card**. Va da sé che questo è un consiglio e non un'imposizione, ma cogliamo l'occasione per spiegare come flashare il SO su un'unità esterna. Per iniziare scarichiamo il tool Raspberry Pi Imager sul nostro PC dal link www.raspberrypi.com/software/. Ricordiamo che il tool è cross-platform e che quindi può essere installato ed eseguito su macchine Windows, Linux e MacOS. Una volta installato avviamolo e inseriamo la scheda microSD nel PC (**Fig. 1**). Selezioniamo

allora la voce "Scegli SO" -> "Misc utility images" -> "Bootloader" -> "USB Boot".

Successivamente scegliamo la voce "Scegli scheda SD" e selezioniamo la scheda microSD. Effettuata quest'ultima scelta clicchiamo sulla voce "Scrivi" e diamo i vari consensi necessari per avviare la scrittura del sistema operativo su SD. Dopo poco tempo ci apparirà un pop-up per informarci che la scrittura è andata a buon fine e che è possibile rimuovere la scheda SD dal PC. Inseriamo quindi nell'apposito slot del Raspberry Pi e colleghiamo tutti i dispositivi di archiviazione da quest'ultimo. Colleghiamo il mini PC all'alimentazione e il gioco è fatto! Se lo colleghiamo al monitor, infatti, possiamo notare lo schermo verde lampeggiare qualche volta fino a stabilizzarsi. Attendiamo qualche secondo, dopodiché possiamo spegnere il Raspberry Pi e rimuovere la scheda SD.

Fig. 1
Scelta dell'utility
per il bootloader.



Questa operazione è necessaria per inizializzare correttamente il minicomputer: dal prossimo avvio il SO verrà cercato su una periferica USB e non su SD card.

Come periferica USB abbiamo scelto un SSD per avere il massimo delle prestazioni. Installiamo quindi TwisterOS. Per farlo è sufficiente scaricare il file immagine dalla pagina twisteros.com/twisteros.html (**Fig. 2**) e utilizzare di nuovo sull'Imager. Questa volta l'opzione da selezionare sarà "Scegli SO" -> "Usa immagine personalizzata" -> "/PERCORSO_DEL_FILE_IMMAGINE".

Ora colleghiamo l'SSD al PC utilizzando un convertitore USB-SATA e selezioniamolo dal menù "Scegli scheda SD". Avviamo il flashing dell'immagine e una volta terminata la procedura possiamo collegare l'SSD ad una delle porte USB 3.0 del Raspberry Pi per poter eseguire il primo avvio.

PRIMO AVVIO

Una volta collegato l'SSD possiamo collegare il LED Matrix Controller, mouse, tastiera e monitor e infine l'alimentazione. Le periferiche sono indispensabili esclusivamente per la prima configurazione e successivamente possono anche essere rimosse.

Una volta alimentato il Raspberry Pi, dopo qualche minuto ci apparirà una schermata di benvenuto (**Fig. 3**) e un pop-up che ci propone di fare un breve tutorial e di completare la configurazione. Premiamo quindi su "Next" -> "Let's Go...". Si aprirà una nuova schermata (**Fig. 4**) dell'utility di configurazione di Raspberry Pi OS dove potremo inserire un nuovo hostname (ad esempio LEDMatrixController) e clicchiamo su "Change Password...". Nel nuovo pop-up inseriamo la password e confermiamola, una volta fatto clicchiamo su "OK" per procedere. Spostiamoci nel menù "Interfaces" (**Fig. 5**) e abilitiamo: SSH, VNC, Serial Port e Serial Console.

Successivamente dal menù "Localisation", premiamo su "Set Locale..." e impostiamo la lingua come mostrato in **Fig.6**.

Eseguiamo la stessa procedura per "Set Timezone...", "Set Keyboard..." (**Fig. 7**) e "Select WiFi Country..." così da impostare Raspberry Pi per la nostra nazione e lingua.

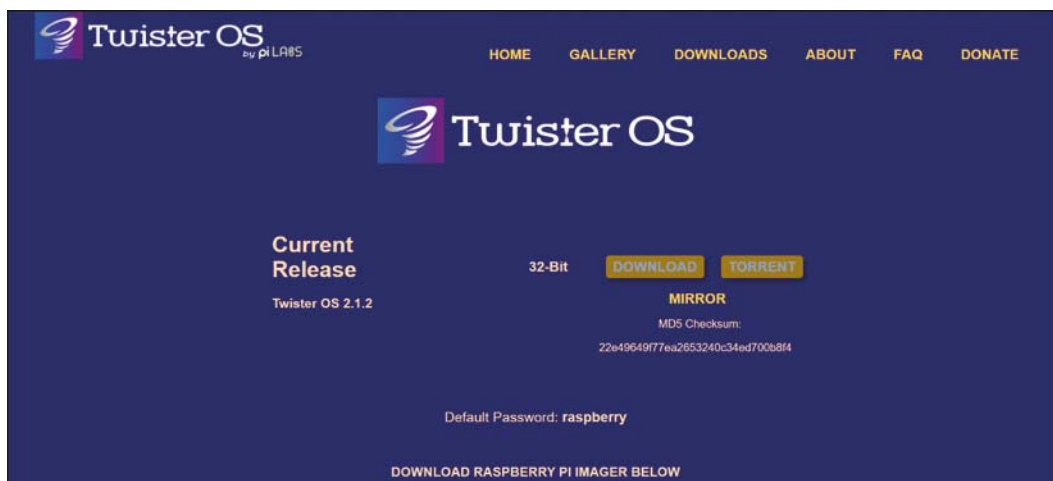
Per poterci collegare al Raspberry Pi senza monitor utilizzando VNC è indispensabile configurarlo correttamente. Il Raspberry Pi, infatti, adatta automaticamente la risoluzione a seconda dello schermo rilevato.

Se il monitor non è presente il sistema non avvia il desktop environment.

Apriamo dunque il terminale e digitiamo il comando `sudo raspi-config` e rechiamoci nel menù "Display Options" -> "Resolution" e impostiamo quella più adatta al nostro monitor (**Fig. 8**).

A questo punto la configurazione di base è terminata. Dal desktop clicchiamo l'icona del WiFi in alto a destra per eseguire la connessione alla rete. Selezioniamo l'SSID e inseriamo (se presente)

Fig. 2
Pagina di download
di TwisterOS.



la chiave di autenticazione. Una volta eseguita la connessione clicchiamo sull'icona in basso al centro del menù e selezioniamo *"Advanced Network Settings"*, scegliamo la rete precedentemente configurata e successivamente nel sottomenù *"Impostazioni IPv4"*. Assegniamo, dunque, un IP statico al Raspberry Pi tenendo conto della subnet mask di appartenenza e settiamo anche i parametri per i server DNS.

Una volta terminata la configurazione di rete riavviamo il Raspberry Pi affinché possa ottenere un nuovo indirizzo IP e vengano applicate le impostazioni salvate precedentemente.

A questo punto possiamo anche scollegare il monitor dal Raspberry Pi e connetterci utilizzando **VNC** e/o il protocollo **SSH**. Tra i programmi utili per collegarci al Raspberry Pi in SSH (porta 22) consigliamo Putty, MobaXTerm o TeraTerm su Windows oppure il terminale nativo su Linux e MacOS. Per utilizzare VNC possiamo installare invece VNC Viewer oppure Remmina.

CONFIGURAZIONE DEL LED MATRIX CONTROLLER

Per configurare il Led Matrix Controller è necessario instaurare una connessione seriale su una delle due porte della scheda. Ricordiamo infatti che il convertitore USB/TTL della scheda crea due **VPC** (Virtual Com Port) associate ad una singola porta USB: una viene utilizzata per ricevere dati nel protocollo *"Glediator"* e l'altra per impostare i parametri di configurazione. Sui sistemi Linux sono già presenti tutti i driver necessari e quindi per poter configurare il tutto è sufficiente installare **Putty** sul PC utilizzando i seguenti comandi sul terminale di Raspberry Pi:

```
sudo -i
apt-get update
apt-get upgrade -y
apt-get install putty -y
```

Dopo l'installazione, lanciando il programma digitando *putty*, da terminale apparirà una schermata nella quale possiamo inserire le configurazioni come in **Fig. 9**. Nel dettaglio selezioniamo la voce *"Serial"* e poi rimpiazziamo *"/dev/ttyS0"* con *"/dev/ttyUSB1"* e impostiamo il baud rate (*"Speed"*) a 115200 bps. Apriamo dunque la connessione premendo su *"Open"*. Si aprirà una schermata nera, e, una volta premuto il tasto invio, verrà mostrata la schermata di configurazione blu con la scritta *"LED Matrix"* (**Fig. 10**).

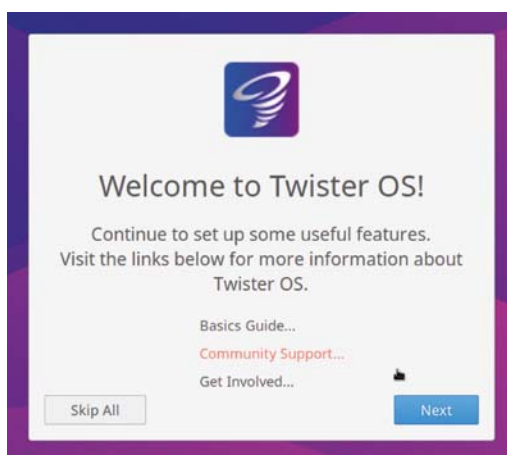


Fig. 3
Primo avvio di TwisterOS.

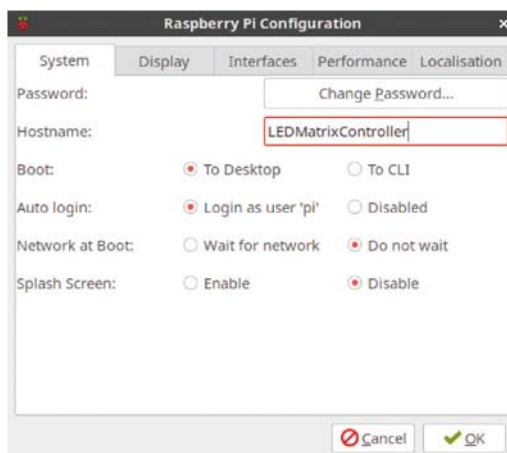


Fig. 4
Prima schermata di configurazione.

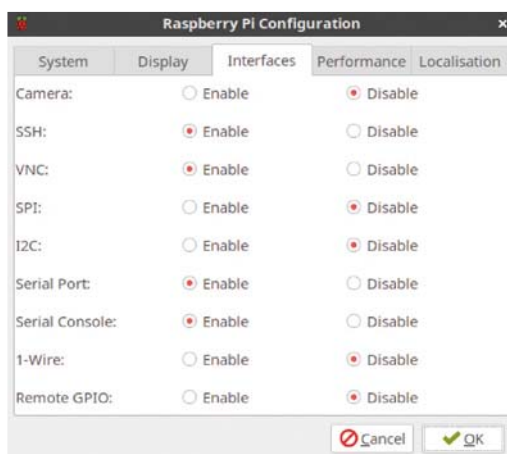


Fig. 5
Abilitazione delle interfacce.



Fig. 6
Impostazione della lingua.

Fig. 7
Impostazione
della tastiera.

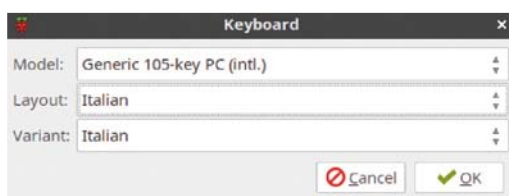


Fig. 8
Impostazione della
risoluzione.

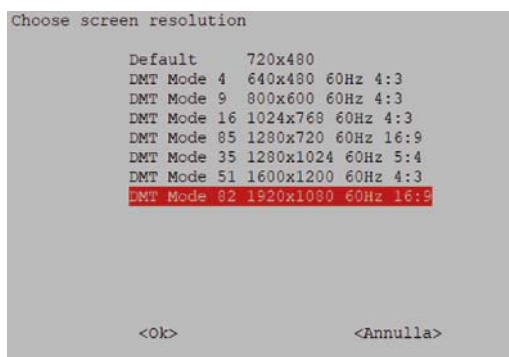
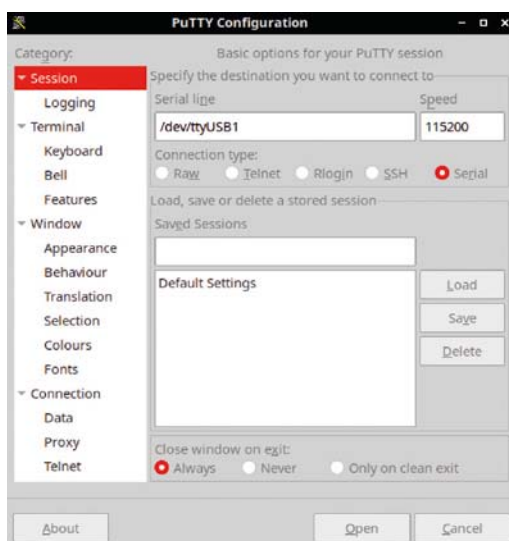


Fig. 9
Schermata
iniziale di Putty.



Se non dovete ottenere la schermata blu sullo schermo, chiudete Putty e ripetete le stesse istruzioni per l'altra porta (/dev/ttyUSB0). A questo punto digitiamo il comando "Preset" seguito dal tasto TAB. Verranno elencati una serie di preset disponibili: scegliamo il 6 e digitiamo invio. Digitiamo il comando "Reboot" e premiamo ancora invio, per riavviare la scheda.

INSTALLAZIONE JAVA E LIBRERIE

Per poter utilizzare il software Glediator è indispensabile che sul sistema operativo siano installati Java e una libreria di sistema per la comunicazione seriale.

Quindi dal terminale diamo nell'ordine i seguenti

comandi per l'installazione dell'ambiente Java:

```
sudo -i
apt-get install default-jre -y
apt-get install default-jdk -y
```

Per l'installazione di Glediator dovremo utilizzare la WayBack Machine ovvero l'archivio digitale del World Wide Web (https://web.archive.org/web/*/Solderlab.de) oppure scaricare l'archivio dal sito della rivista in quanto non più disponibile da quello ufficiale.

Una volta scaricato il pacchetto potremo installarlo lanciando il comando `java -jar 'Glediator_V2.jar'`.

Per comunicare con il software Glediator dovremo installare la libreria che si occupa della comunicazione seriale. Aprite quindi la pagina packages.debian.org/buster/librtx-java e selezionare la versione **armhf**. Verremo reindirizzati in una pagina per la scelta del mirror dal quale effettuare il download.

Una volta scaricata la libreria rechiamoci nella cartella in cui si trova il file `deb`, selezioniamolo e facciamo click destro su "Apri con 'Installa pacchetti'". Quando richiesto confermiamo la nostra volontà di voler installare il pacchetto cliccando su "Installa" e successivamente inserendo la password utente, quando richiesta.

CONFIGURAZIONE GLEDIATOR

Siamo adesso pronti, dunque, per il "primo avvio" di Glediator. Rechiamoci nuovamente nella cartella in cui si trova il file `.jar` (supponiamo sia la cartella "Glediator" sul Desktop), riapriamo un terminale e lanciamo il comando `java -jar 'Glediator_V203.jar'`. Sul terminale compariranno delle stampe di debug e se tutto sarà andato a buon fine ci comparirà una schermata come in **Fig. 11**. Ora dovremo configurare il software: dal menù "Options" -> "Matrix size" inseriamo i dati come in **Fig. 12** e successivamente dal menù "Options" -> "Output" impostate i vari campi come illustrato nella **Fig. 13**.

Come ultimo step è sufficiente aprire la porta seriale su cui avviare la trasmissione (in questo caso "dev/ttyUSB0") con baud rate 115200 bps. Se clicchiamo sul pulsante "Serial Open" i led si illumineranno e inizieranno ad eseguire tutti i comandi impartiti loro da Glediator. Una volta chiusa questa finestra di dialogo possiamo iniziare a prendere confidenza con l'interfaccia utente di Glediator per generare una lunga serie di anima-

zioni e persino riprodurre file precedentemente generati da Jinx o Glediator stesso. Per velocizzare e semplificare questa operazione dal sito della rivista potrete scaricare un'animazione natalizia e caricarla su Glediator premendo su "Load" e poi su "Play" (Fig 14). Ricordiamo che le scene vengono automaticamente salvate alla chiusura e ripristinate all'avvio di Glediator.

AVVIO AUTOMATICO

Eseguire in maniera automatica il software Glediator non è un grosso problema, purtroppo non è possibile fargli eseguire in automatico un'animazione prememorizzata. A differenza di Jinx, infatti, Glediator non apre in automatico la comunicazione con i vari dispositivi di output, di qualunque tipo essi siano.

Per ovviare al problema avevamo due possibilità: lasciare all'utente l'avvio del sistema operativo e del software attraverso un display touch-screen, oppure creare un'automazione che lo facesse per noi.

Ovviamente la seconda opzione è risultata quella più congeniale per diversi motivi, tra tutti l'automatizzato ripristino dello stato precedente in caso di blackout e/o in assenza dell'utente.

Per fare ciò abbiamo utilizzato un software open source scritto in Python che registra l'input di mouse e tastiera e permette di riprodurli in qualunque momento. Il nome di questo software è ATBSWP (Automate The Boring Stuff With Python) ed è reperibile al link github.com/RMPR/atbswp.



Fig. 10
Configurazione
del Led Matrix
Controller.

Per installare ATBSWP è sufficiente eseguire quanto riportato nella guida ufficiale e cioè utilizzare il terminale e digitare i seguenti comandi:

```
sudo apt install git python3-dev python3-tk python3-
setuptools python3-wheel python3-pip python3-
wxgtk4.0
git clone https://github.com/RMPR/atbswp.git && cd
atbswp
python3 -m pip install pyautogui pynput --user
```

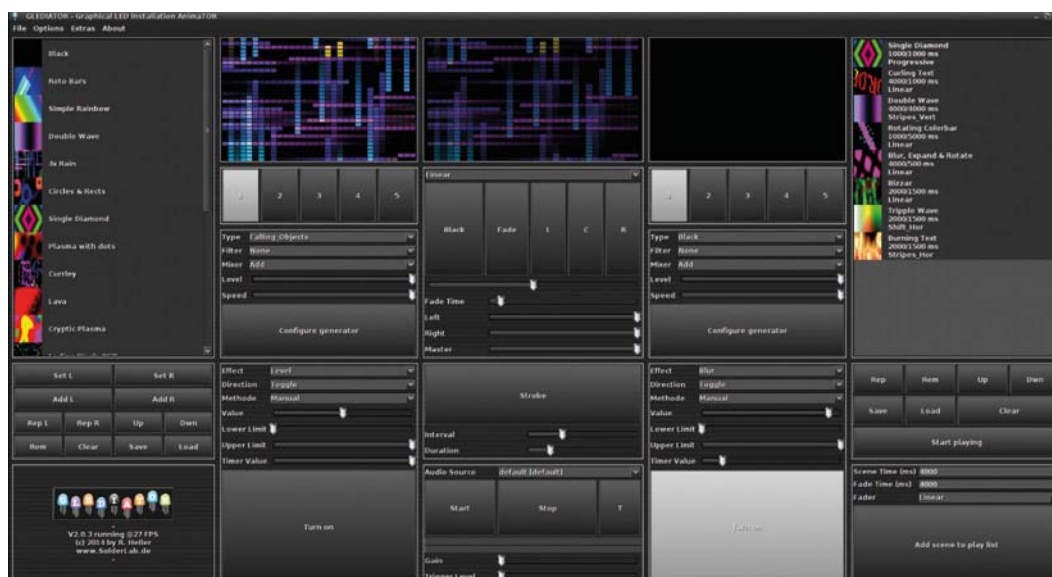
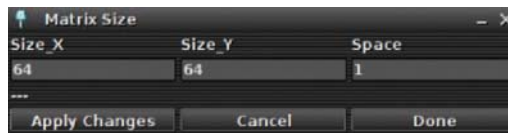


Fig. 11
Schermata
iniziale di
Glediator.

Fig. 12
Impostazione
delle dimensioni
della matrice su
Glediator.



A questo punto ATBSWP è stato correttamente installato. Prima di iniziare la registrazione è necessario eseguire ancora qualche semplice operazione per eseguire Glediator tramite uno script bash invece che dal classico terminale. Nello specifico abbiamo creato un file nella cartella di Glediator utilizzando il comando

```
sudo nano Glediator_avvio.sh
```

e abbiamo inserito le seguenti righe di codice

```
#!/bin/bash
nohup java -jar '/home/pi/Desktop/Glediator/Glediator_V203.jar'
```

Una volta terminata la modifica premiamo CTRL+X e quando richiesto premiamo y. Verrà dunque chiuso l'editor di testo e per rendere il file eseguibile utilizziamo il comando `sudo chmod +x Glediator_avvio.sh`. Giunti a questo punto selezioniamo il file `Glediator_avvio.sh` con il testo destro e seguiamo il menù "Apri con altra applicazione" -> "Emulatore di terminale".

Ricordiamoci di rendere questa opzione definitiva spuntando la casella in basso e a questo punto possiamo aprire Glediator eseguendo un doppio click sull'icona del file bash.

Lanciando lo script `Glediator_avvio.sh` verrà aperto il programma di animazione ed eseguendo con il comando:

```
python3 atbswp/atbswp.py
```

lanceremo il software ATBSWP.

Una volta avviato si aprirà una finestra come quella in **Fig. 15**, dalla quale dovremo premere sull'icona della fotocamera. In questo modo il programma inizierà a registrare tutti i nostri movimenti del mouse per poi riproporli quando vorremo.

Una volta catturati tutti i passaggi possiamo interrompere la registrazione premendo nuovamente sull'icona della fotocamera e salvare il file nel formato .pyc, ad esempio `capture.pyc`, cliccando sull'icona a forma di download. A questo punto è possibile riprodurre la registrazione lanciando il file:

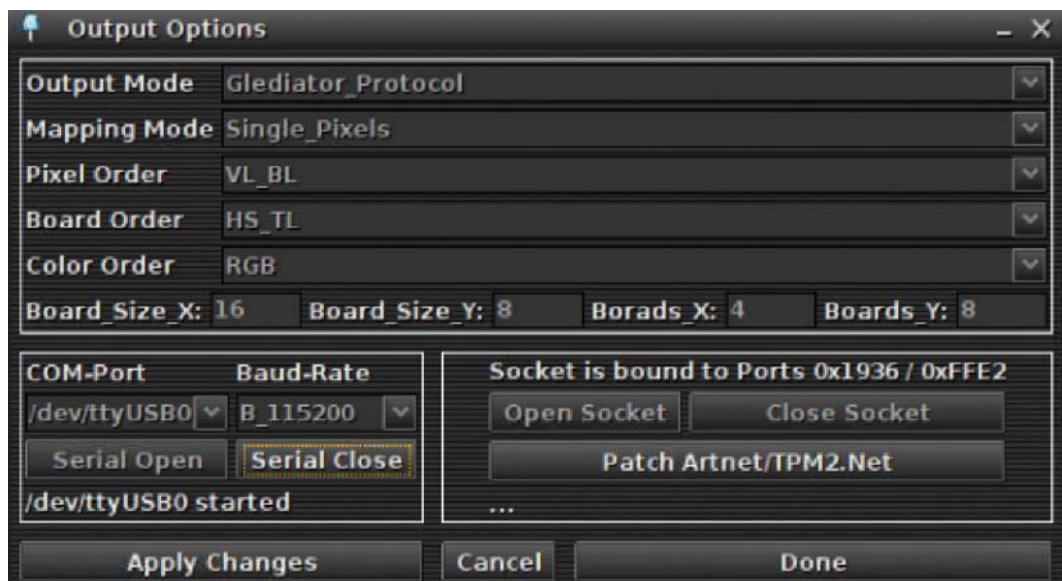
```
python3 capture.pyc
```

Non ci rimane ora che far partire lo script Python all'avvio di TwisterOS e per questo abbiamo deciso di utilizzare **systemd**. Per impostare l'avvio automatico dello script Python utilizzando **systemd** usiamo il comando:

```
sudo nano /lib/systemd/system/avvio-animazione.service
```

per aprire l'editor di testo da terminale e copiamo il contenuto del **Listato 1**. Una piccola digressione si rende obbligatoria in merito alla riga `"ExecStart= env -i DISPLAY=:0.0 /usr/bin/python3 /`

Fig. 13
Configurazione
dell'output di
Glediator.



Listato 1

```
[Unit]
Description= Avvia la riproduzione del file .pyc per avviare Glediator e il suo output
After=multi-user.target

[Service]
Type=idle
ExecStart= env -i DISPLAY=:0.0 /usr/bin/python3 /home/pi/Desktop/Glediator/capture.pyc
WorkingDirectory=/home/pi
User=pi

[Install]
WantedBy=multi-user.target
```

`home/pi/Desktop/Glediator/capture.pyc`. Quest'ultima, infatti, serve ad indicare il comando che va eseguito all'avvio del Raspberry Pi. La parte `"env -i DISPLAY=:0.0"` serve ad identificare lo "schermo" sul quale hanno effetto i comandi, infatti se non fosse presente non saremmo in grado di definire l'ambiente grafico e quindi non sarebbe possibile eseguire lo script correttamente. Il comando `"/usr/bin/python3"` è stato utilizzato per indicare quale versione di Python eseguire. Come argomento di Python (cioè lo script che deve essere eseguito) abbiamo riportato il percorso dello script `"capture.pyc"`, quindi cambiamolo a seconda della posizione del file. Una volta salvato il file lanciamo i seguenti comandi:

```
sudo chmod 644 /lib/systemd/system/avvio-animazione.service
sudo systemctl daemon-reload
sudo systemctl enable avvio-animazione.service
sudo reboot
```

Ad ogni avvio, Raspberry Pi lancerà lo script Python che replicherà i movimenti sullo schermo, avvierà Glediator e caricherà l'animazione.

MONTAGGIO FINALE

Lo schema di cablaggio mostra nel dettaglio come collegare i vari componenti. Prestate attenzione alla tensione di alimentazione per i pannelli, Raspberry Pi e LED Matrix che deve essere di 5V. In particolare su quest'ultima dovreste aprire il Jumper vicino alla presa USB poichè utilizzeremo l'alimentazione dai connettori anziché dalla porta USB.

CONCLUSIONE

Si conclude la presentazione di questo progetto le cui applicazioni sono infinite e ricoprono diversi ambiti. Sta al lettore decidere il campo d'utilizzo e quale limite porre alla propria fantasia. □

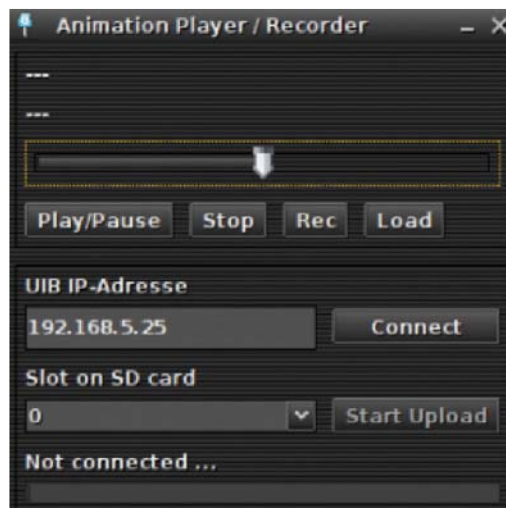


Fig. 14
Caricamento di un'animazione precedentemente creata.



Fig. 15
Schermata iniziale di ATBSWP.



Cosa occorre?

I componenti principali utilizzati in questo progetto sono disponibili presso Futura Elettronica. In particolare Raspberry Pi 4 da 8Gb (cod. RPI4-8GB) è disponibile a Euro 99,00, la scheda LED Matrix (cod. FT1286M) è in vendita a Euro 94,00, il display a matrice 64x32 (cod. RGBLEDPANEL6432) costa Euro 76,00, l'alimentatore 5V 50W (cod. MW05005) è disponibile a Euro 15,00

Il materiale va richiesto a:

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

HDMI
↓
VGA

€ 11,50

Cod. **BOXHDMITOVGA**

CONVERTITORE DA HDMI A VGA 1080P PLUG AND PLAY

Permette di convertire un segnale digitale HDMI (compatibile con HDMI1.3) in un segnale VGA (SUB-D) ad alta definizione con audio in uscita. Con questo convertitore, il computer, il lettore video portatile, le console di gioco e alcuni altri dispositivi video con uscita HDMI possono connettersi a schermi HD di grandi dimensioni oppure a proiettori HD con ingresso VGA.

CONVERTITORE DA AV A HDMI PLUG AND PLAY

AV
↓
HDMI

€ 14,00

Cod. **BOXAVTOHDMI**

Permette di convertire un segnale audio/video CVBS (PAL e NTSC) in ingresso in un segnale HDMI in uscita. Con questo convertitore, il computer, il lettore video portatile, le console di gioco e alcuni altri dispositivi video con uscita CVBS possono connettersi a schermi con ingresso HDMI. L'uscita HDMI supporta una risoluzione fino a 1080P 60Hz.

Convertitori di segnale

Permettono di convertire un segnale digitale HDMI in VGA, AV in HDMI, VGA in HDMI, HDMI in AV.

CONVERTITORE DA HDMI AD AV PLUG AND PLAY

Permette di convertire un segnale HDMI in ingresso in un segnale video composito CVBS (PAL e NTSC) in uscita. Con questo convertitore, il computer, il lettore video portatile, le console di gioco e alcuni altri dispositivi video con uscita HDMI possono connettersi a schermi con ingresso CVBS. L'ingresso HDMI supporta una risoluzione fino a 1920x1080 a 60 Hz.

VGA
↓
HDMI

€ 11,50

Cod. **BOXVGATOHDMI**

CONVERTITORE DA VGA A HDMI PLUG AND PLAY

Permette di convertire un segnale audio e video VGA (SUB-D) in un segnale HDMI (tipo A) ad alta definizione con audio. Con questo convertitore, il computer, il lettore video portatile, le console di gioco e alcuni altri dispositivi video con uscita VGA possono connettersi a schermi HD di grandi dimensioni oppure a proiettori HD con ingresso HDMI. L'ingresso VGA supporta video fino a 1920x1080 a 60 Hz, l'uscita HDMI può supportare video fino a 1080P.

HDMI
↓
AV

€ 11,00

Cod. **BOXHDMITOAV**

SMART SWITCH

di MAURIZIO ŠKERLIČ



Interruttore IoT controllabile da remoto con App o comandi vocali utilizzando l'assistente vocale Amazon Alexa oppure Google Assistant e configurabile da pagina web. Ne proponiamo due versioni: una con relé e una con SSR.

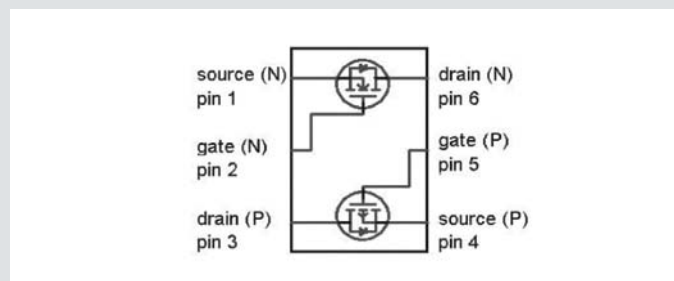
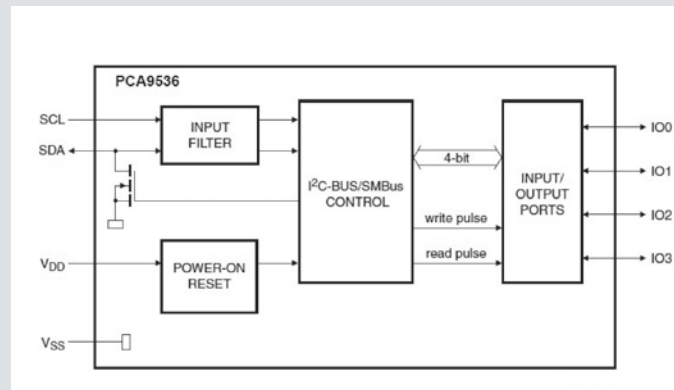
Recentemente abbiamo pubblicato lo Smart dimmer, un interruttore e varia-luce a controllo sia tattile che tramite assistente vocale Amazon Alexa o Google Assistant, utile per incominciare ad allestire o per ampliare l'impianto domotico di casa. Grazie al funzionamento trailing edge o leadig edge, permette di variare l'intensità luminosa di tutti i tipi di lampadine presenti in commercio. Rimanendo sulla stessa linea, in questo articolo vogliamo presentare il progetto di un interruttore smart, battezzato "Smart switch", proposto in due versioni: a relé elettromeccanico (EMR - ElectroMechanical Relay) ed a relé allo stato solido (SSR - Solid State Relay).

LA SCELTA DEL TIPO DI RELÉ

L'aspetto più importante nella scelta di un relé è sempre

In questo riquadro potete vedere lo schema a blocchi del port-expander PCA9536 utilizzato nello Smart switch e del doppio MOSFET BSD235C che nella versione EMR dello

Smart switch è utilizzato per pilotare la bobina di eccitazione del relé; il doppio MOSFET a canale N e P con diodo di riciclo che limita le sovratensioni da carico induttivo.



quello di considerare quale tipo di carico sarà applicato sui contatti. In base al tipo di carico, la corrente di spunto durante la chiusura dei contatti assume valori molto diversi. I carichi che possiamo applicare possono essere di tre tipi diversi resistivo, induttivo e capacitivo.

Nel caso del carico resistivo, la scelta del relé non presenta particolari difficoltà, mentre per quelli induttivi oppure capacitivi, si deve prestare molta attenzione al tipo di relé da utilizzare. Ad esempio per un motore elettrico la corrente di spunto può raggiungere fino 10 volte la corrente nominale e durante l'apertura dei contatti saranno generati archi voltaici che possono procurare diversi danni ad essi. Questo significa che un motore durante il funzionamento normale assorbe 1 A, durante la fase di partenza potrebbe assorbire fino 10A. Anche le lampade assorbono correnti elevate all'applicazione iniziale dell'energia. Il filamento di una lampadina a incandescenza ha un coefficiente di temperatura elevato. Quando è fredda, la resistenza del filamento spesso è cinque per cento inferiore della resistenza di una lampadina calda e quando è fredda assorbe una corrente venti

volte maggiore rispetto quando è calda. Durante il normale funzionamento, una lampadina a incandescenza da 60 W assorbe circa 300 mA ma alla prima accensione il filamento freddo assorbe una corrente di inserzione che può raggiungere i 10 A. Sebbene questo picco di corrente duri solo un decimo di secondo, i contatti dei relé comunque subiscono un degrado.

Poiché i carichi capacitivi presentano elevati picchi transitori di corrente all'accensione, commutare una tensione verso un condensatore scarico equivale a pilotare momentaneamente un cortocircuito. Questa corrente elevata all'accensione può fondere o addirittura saldare assieme i contatti del relé. Nei carichi induttivi invece, quando il carico viene interrotto, si verificano picchi transitori di tensione induttiva attraverso i contatti del relé. Il picco transitorio di tensione indotta può essere sufficientemente grande da formare un arco nei contatti del relé, che li degrada fondendo la superficie di contatto ogni volta che viene data e tolta l'alimentazione.

Per proteggere i contatti e prolungarne la vita, di solito si collega in parallelo ad essi uno snubber costituito da un condensatore posto in serie ad una resistenza. Questo oltre a prevenire l'arco elettrico sul contatto di commutazione sopprime le sovratensioni funzionando come un soppressore dei disturbi EMI (*electromagnetic interference*).

Una soluzione alternativa è quella di sostituire il comune relé elettromeccanico con uno più moderno allo stato solido SSR (*solid state relay*) dove la commutazione non avviene più con parti in movimento bensì con componenti elettronici, rendendo il relé più silenzioso e più veloce. Inoltre, grazie a circuiti interni per il rilevamento del passaggio della semionda per lo zero, permette di commutare il carico quando la tensione è minima, eliminando gli indesiderati disturbi di commutazione tipici dei relé tradizionali. Quindi a seconda del tipo di carico da pilotare, sceglieremo la versione dello Smart switch più adatto alle nostre esigenze, prediligendo la versione con relé allo stato solido per pilotare carichi induttivi o capacitivi.

SCHEMA ELETTRICO

Poiché nell'articolo vengono proposte due versioni dello Smart switch, che si differenziano soltanto nel tipo di pilotaggio del carico in uscita, analizzeremo contemporaneamente lo schema elettrico di entrambe le versioni e separatamente soltanto le differenze. Incominciamo partendo dal modulo ESP03, siglato U4 basato sul microprocessore

Espressif ESP8266 con architettura RISC a 32 bit, clock a 80 MHz ed equipaggiato con una memoria Flash da 1MB. Il circuito viene alimentato da un trasformatore switching da 5 V 1 A, nello schema elettrico siglato TR1, la cui tensione viene abbassata a 3,3 V dal regolatore di tensione NPC1117, nello schema elettrico siglato U3. La tensione a 5 V è filtrata dal condensatore C10 mentre quella a 3,3 V dai condensatori C8 e C9 indispensabili per evitare reset o crash di sistema casuali. I piedini 10 e 11 dell'ESP03, rispettivamente SDA e SCL, sono utilizzati per la comunicazione I²C con l'integrato PCA9536 un espansore di porte input output a 4 bit, nello schema elettrico siglato U1. Il piedino 1 di U1 è utilizzato come porta output mentre il piedino 5 è utilizzato come porta input. Quest'ultimo è mantenuto ad alto livello dalla resistenza R3 e il segnale, proveniente dal collettore del transistor presente nel fotoaccoppiatore U2, viene filtrato dal condensatore C1. Il fotoaccoppiatore U2 grazie al doppio diodo LED in entrata e le due resistenze R1 e R2, permette di disaccoppiare la tensione alternata a 220 V proveniente dal pulsante collegato al piedino 1 del connettore CON1. Le resistenze R4 e R5 usate come resistenze di pullup mantengono alto rispettivamente le linee SDA e SCL del segnale I²C.

Il pulsante SW1 collegato al piedino GPIO2 del modulo ESP03, mantenuto ad alto livello dalla resistenza R6, serve a reimpostare la password predefinita di sistema "123456789", in caso dimenticaste quella memorizzata. Il piedino 1 di U1, a seconda della versione, pilota direttamente un relé allo stato solido oppure un relé elettromeccanico mediante il doppio mosfet nello schema elettrico siglato con T1. Il diodo D2, nella versione a relé elettromeccanico, serve a sopprimere le extratensioni di segno opposto generate dalla bobina quando il relé viene diseccitato e serve a non far bruciare T1.

È possibile calcolare l'extratensione utilizzando la seguente formula:

$$V = L \cdot \frac{I}{t}$$

Dove V è la tensione da calcolare, L è il valore dell'induttanza della bobina, I è la corrente nominale della bobina e t corrisponde al tempo di rilascio dell'alimentazione.

Il carico viene collegato tra i piedini 2 e 3 e la tensione di rete neutro e fase viene collegata rispettivamente ai piedini 4 e 5 di CON1. Il circuito è protetto da sovratensioni e sovracorrenti mediante il

varistore VR1 e il fusibile F1, mentre il diodo transil unidirezionale D1 protegge il circuito da una tensione eccessiva in uscita dall'alimentatore in caso di guasto. Nella versione a relé elettromeccanico in parallelo ai contatti è stato inserito uno snubber RC formato dalla resistenza R11 e un condensatore C11, che ha una duplice funzione, sopprime l'arco sui contatti proteggendoli dalla precoce usura con carichi induttivi o capacitivi e riduce i disturbi elettromagnetici durante la commutazione. Nella versione a relé SSR, la rete snubber non è necessaria, poiché integrata nello stesso Solid State Relay; inoltre il carico viene acceso sempre al passaggio della semionda per lo zero diminuendo i disturbi e gli stress elettrici su di esso.

Nella versione EMR la bobina di eccitazione del relé viene pilotato da un doppio mosfet a canale N e P siglato T1 e in parallelo ad essa troviamo un diodo di riciclo (Flyback) che limita le sovratensioni induttive proteggendolo.

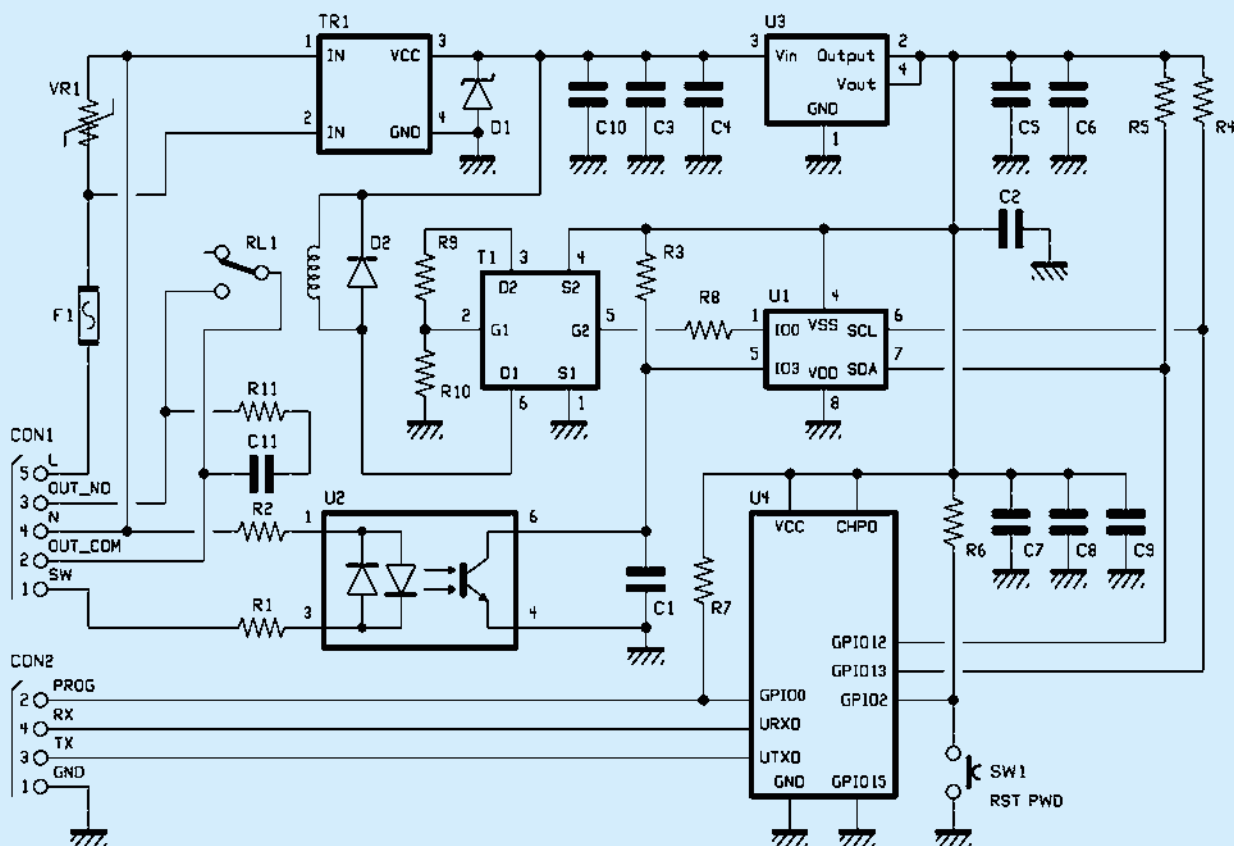
REALIZZAZIONE PRATICA

Passiamo ora alla realizzazione pratica che richiederà una conoscenza minima nel montaggio di componenti a montaggio superficiale. Come sempre per il montaggio dei componenti suggeriamo l'utilizzo di un saldatore stilo a punta sottile con potenza non troppo elevata (20 ÷ 25 watt) e del filo di lega saldante possibilmente non superiore a 0,7 mm di diametro. Consigliamo di usare un fluxante di ottima qualità che non lascia scorie residue utile soprattutto quando usate leghe saldanti senza piombo la cui temperatura di fusione è maggiore rispetto quelle tradizionali. Per posizionare i componenti aiutatevi con delle pinzette a punta fine possibilmente amagnetiche di acciaio inox oppure con le punte in plastica o ceramica, altrimenti magnetizzandosi i componenti saranno attratti dalle punte metalliche complicandovi il lavoro. Consigliamo sempre di iniziare a saldare i semiconduttori e successivamente tutti i componenti passivi. Fate attenzione al corretto orientamento dei semiconduttori come visibile nel piano di montaggio. Nel saldare i componenti passivi consigliamo di stagnare prima quelli più piccoli e successivamente passare a quelli con case più grande, come il varistore VR1, il condensatore C11 e il pulsante SW1. Il modulo ESP03, che nello schema elettrico è siglato U4, andrà prima stagnato sull'adattatore e successivamente, per ottimizzare lo spazio, montato verticalmente come visibile in **Fig. 1**. Passate ora al montaggio dei componenti a foro passante come i due connettori

CARATTERISTICHE TECNICHE

- Carico massimo 350W versione SSR e 550W versione a Relè
- Adatto per carichi resistivi o induttivi
- Protezione termica, da sovracorrenti e sovratensioni
- Timer spegnimento automatico
- Configurazione tramite interfaccia web
- Collegamento alla rete WiFi domestica
- Protezione con password
- Reset password tramite pulsante
- Compatibile con frutti da incasso Vimar, Gewiss o bTicino





CON1 e CON2, il fusibile F1, il trasformatore TR1 e per finire, a seconda della versione, il relé allo stato solido, nello schema elettrico siglato SSR1, oppure quello elettromeccanico, nello schema elettrico siglato RL1.

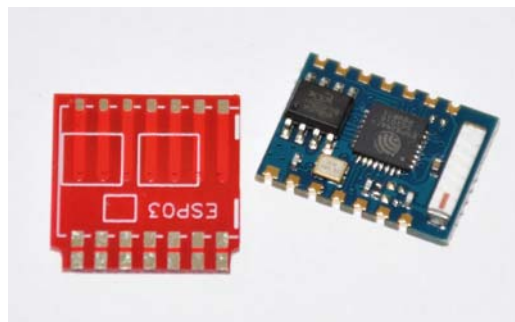
Una volta terminato il montaggio di tutti i componenti consigliamo di pulire il PCB dal fluxante con alcol isopropilico. Con una lente di ingrandimento controllate attentamente che non ci siano ponticelli di stagno tra i piedini dei componenti in particolar modo di U1. A questo punto potete alimentare il circuito, come visibile nel circuito elettrico,

collegando al connettore CON1 la fase al piedino 5 e 2, il neutro al piedino 4, il carico tra il piedino 3 e neutro e un pulsante o un interruttore collegato tra il piedino 1 e la fase. Facciamo presente di non toccare con le mani nude il circuito quando è allacciato alla rete elettrica a 230V, altrimenti rischiate di rimanere folgorati.

PROGRAMMAZIONE E COLLAUDO

La programmazione del modulo ESP03 deve essere effettuata mediante un adattatore USB per ESP8266 come quello a catalogo Futura Elettro-

Fig. 1
Montaggio del modulo ESP03 in posizione verticale sul PCB dello Smart switch.

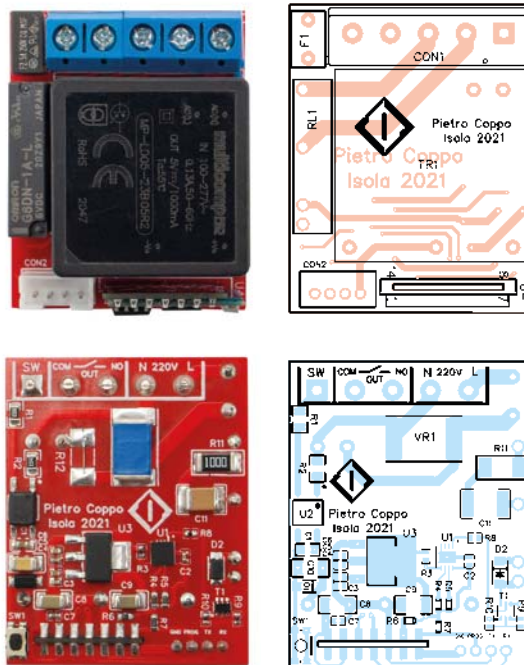


piano di montaggio

VERSIONE A RELÉ

Elenco Componenti:

R1, R2: 200 kohm 300 V (0805)	T1: BSD235CH
R3: 470 kohm (0402)	VR1: V430CH8
R4, R5: 2,2 kohm (0603)	TR1: RAC05-05
R6, R7: 10 kohm (0603)	F1: F 2,5A
R8: 120 ohm (0402)	U1: PCA9536TK
R9: 10 ohm (0402)	U2: TLP184
R10: 10 kohm (0402)	U3: NCP1117ST33T3G
R11: 100 ohm 230 V (2010)	U4: ESP-03
C1: 1 µF 6,3 V (0603)	RL1: G6DN-1A-SL DC5
C2, C3, C5, C7: 100 nF (0402)	CON1: Morsettiera 5 poli P - 5 mm
C4, C6: 10 µF 6,3 V (0402)	CON2: DIP-1x4 P - 2 mm
C8, C9: 100 µF 6,3 V (1206)	SW1: pulsante SMD 4.6X2.8mm
C10: 220 µF 6,3 V (1206)	
C11: 100 nF 500 V (1812)	Varie:
D1: SMF5.0A	- circuito stampato S1585 (41x32,5 mm)
D2: LL4148	



nica codice prodotto YM203, basato sul chip di conversione USB-TTL CH340G, oppure il modello ESP8266 Debugger codice prodotto ESP8266DEB, basato sul chip di conversione USB-TTL CP2014. In alternativa potete usare anche un qualsiasi convertitore USB TTL, purché compatibile con logica a 3,3 V, usato per programmare i moduli Arduino mini come quello a catalogo Futura Elettronica con codice prodotto USBTOUART, oppure quello usato da noi, che è il UC00A, sempre della Futura Elettronica.

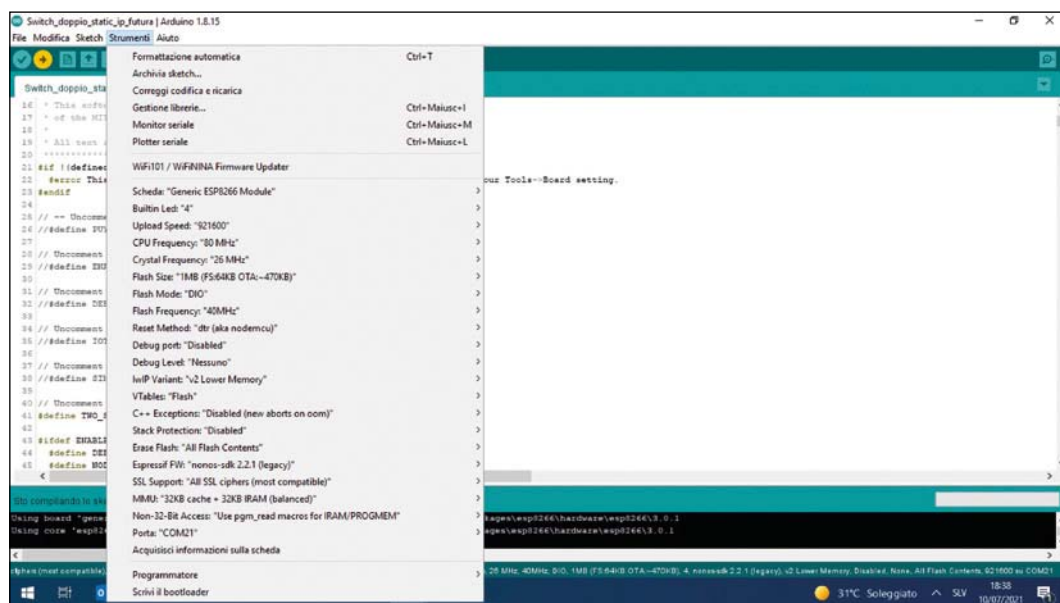
Per la programmazione occorre l'IDE Arduino in esecuzione su un PC, da impostare come mostrato nella **Fig. 2**.

Per il collegamento elettrico fate riferimento alla **Fig. 3**.

Ricordatevi di collegare l'uscita TX del convertitore all'entrata RX pin 3 del connettore CON2 e l'entrata RX del convertitore all'uscita TX pin 4 di CON2. Per mettere lo Smart switch in modalità di programmazione premete il pulsante Prog collegato al pin 2 di CON2 prima di allacciarlo alla corrente elettrica, rilasciandolo dopo un secondo. A differenza degli altri moduli ESP in commercio, l'ESP03 è sprovvisto

del piedino reset, quindi terminata la programmazione, per avviarlo, dovete togliere l'alimentazione staccando e riattaccando la corrente elettrica. In realtà ci sarebbe un metodo alternativo, che noi sconsigliamo, collegando al piedino CH_PD (chip power down) una resistenza di pull up al positivo e per resettare il microprocessore portare il piedino a basso livello. Questa soluzione rende il microprocessore instabile provocando reset e blocchi di sistema casuali. Poiché il modulo ESP03 monta un microprocessore ESP8266 potete programmarlo direttamente dall'ambiente di sviluppo Arduino Ide scaricando le definizioni della scheda dal sito http://arduino.esp8266.com/stable/package_esp8266com_index.json, indirizzo che dovete aggiungere tra gli "URL aggiuntivi per il gestore di schede" nel menu "impostazioni". Dal menu "sketch" selezionate "#include libreria" e cliccate su "Gestione libreria". Dalla finestra Gestione libreria cercate e installate le librerie "IoTWebConf" "SinricPro" e "SparkFun PCA9536 Arduino Library", necessarie per compilare lo sketch "smartswitch.ino" che andrete a scaricare dal sito web www.elettronica.in. A scopo informativo lo Smart switch è stato te-

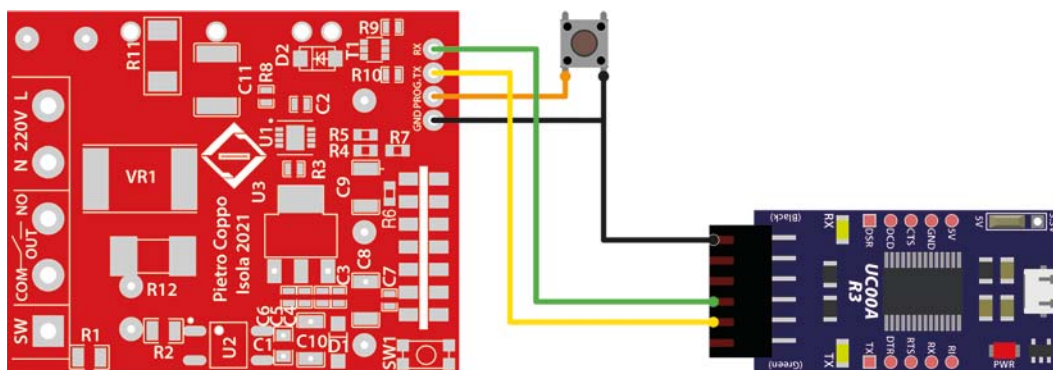
Fig. 2
Impostazioni di
programmazione
del modulo ESP03
in Arduino IDE.



stato compilando il sorgente utilizzando il package core ESP8266 versione 2.7.4 e le seguenti librerie "lotwebconf" versione 3.1.0, "SinricPro" versione 2.9.7, "SparkFun_PCA9536_Arduino_Library" versione 1.2.1 e "ArduinoJson" versione 6.18.3. Fate attenzione che la versione del package core dell'ESP8266 caricato in Arduino Ide sia aggiornato alla 2.7.4, altrimenti avrete problemi di funzionamento con la libreria *lotWebConf*. Al momento della stesura dell'articolo la versione del package core dell'ESP8266 è arrivata alla versione 3.0.2. Con questa versione abbiamo riscontrato problemi con la rete WiFi, probabilmente con gli aggiornamenti futuri saranno risolti. Lo sketch per comodità è stato scritto sia per programmare lo Smart switch, descritto in questo

articolo, sia il Double smart switch, dispositivo capace di comandare due carichi indipendenti, che verrà pubblicato sulla nostra rivista nei prossimi numeri. All'inizio del sorgente noterete alcune righe commentate, ad esempio se la memoria Flash montata sul modulo ESP03 è marchiata Puya, per abilitare il supporto di tale memoria in fase di compilazione, dovete togliere il commento davanti alla riga 31 "#define PUYA_SUPPORT 1". Se volete attivare il debug togliete la spunta davanti alla riga 34 "#define ENABLE_DEBUG", mentre se volete caricare lo sketch sul nostro Smart switch fate attenzione che la riga 43 "#define TWO_SWITCH" resti commentata. Per la configurazione di programmazione del modulo ESP03 vale la **Fig. 2**, che mostra quella da noi

Fig. 3
Cablaggio del
programmatore allo
Smart switch.



utilizzata. Ricordiamo che per caricare il firmware sul modulo ESP03 bisogna metterlo in modalità di programmazione, portando basso con un pulsante il piedino GPIO0 dell'ESP03 pin 2 Prog del connettore CON2, dare tensione al circuito e riportare alto il piedino 2, rilasciando il pulsante, dopo un secondo. Per il corretto cablaggio del pulsante sul connettore CON2 fate riferimento al solito schema di cablaggio di **Fig. 3**.

Una volta terminata la programmazione, con il computer o un altro dispositivo provvisto di scheda WiFi, fate una scansione delle reti wireless e selezionate quella con SSID FuturaElettronica, la password predefinita è "123456789". Lo Smart switch avvierà il "captive portal" all'indirizzo 192.168.4.1 e lancerà in automatico il browser predefinito di sistema visualizzando la pagina web di **Fig. 4**. Nel caso questo non avvenga in automatico fatelo manualmente. Negli smartphone e tablet Android, una volta scelta la rete SSID, il browser non si avvierà in automatico ma bisognerà cliccare sul pulsante accedi alla rete, oppure immettere nella barra degli indirizzi del browser l'indirizzo IP 192.168.4.1. Nella pagina Web iniziale viene visualizzato l'indirizzo fisico ossia l'indirizzo MAC (Media Access Control) dello Smart switch. L'indirizzo MAC è un numero a 48 bit dove i primi 24 bit dell'indirizzo rappresentano il numero ID del produttore dell'adattatore di rete e i secondi 24 bit rappresentano il numero di serie assegnato all'adattatore di rete dal produttore. Prendete nota di questo numero perché vi

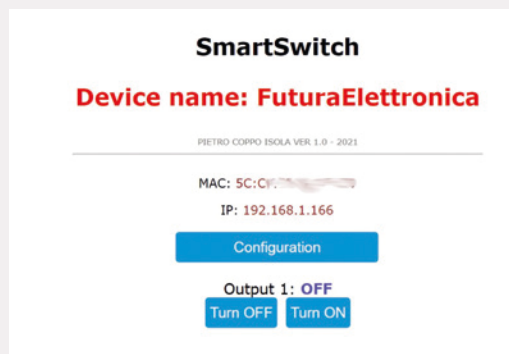


Fig. 4
Home page dello Smart switch, con la funzione web control attivata.

servirà in seguito.

Cliccando sul pulsante di configurazione, nella sezione "System configuration" (**Fig. 5**) potete modificare il nome del dispositivo, modificare la password di sistema, inserire il nome SSID e la password della vostra rete WiFi e lo "Startup delay". Quest'ultimo è il tempo in secondi che, ad ogni accensione o riavvio, lo Smart switch diventa un access point per permettere all'utente di raggiungere la home page visibile in **Fig. 4**.

Nella sezione "Device IP" selezionando la casella di spunta "Enable static IP" potete disabilitare l'assegnazione dell'IP dinamico mediante il DHCP server e impostare quello statico. Nella casella "IP address" scrivete il numero dell'IP statico che volete assegnare al dispositivo, nella casella "Subnet mask" inserite il numero IP della maschera di sottorete della vostra rete, nella casella "Gateway" inserite il numero IP del vostro router e

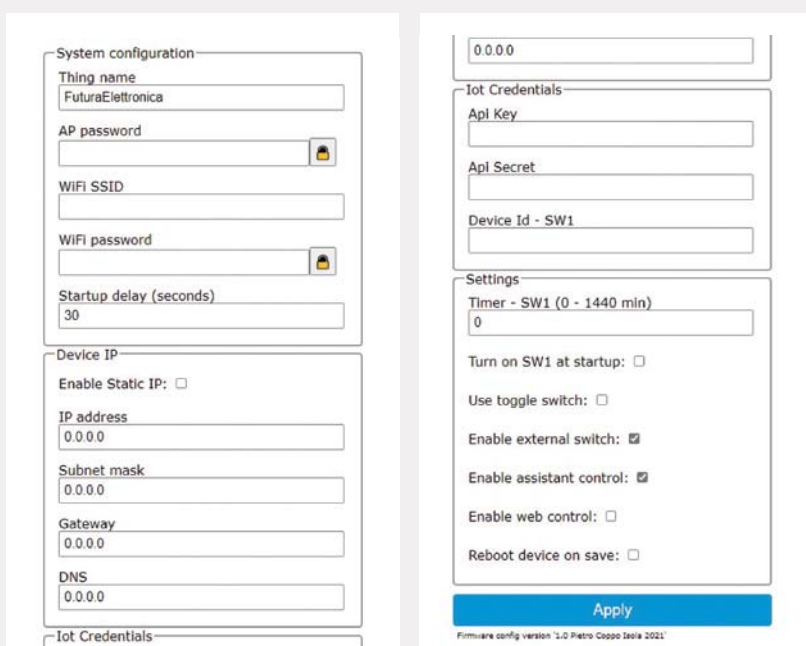
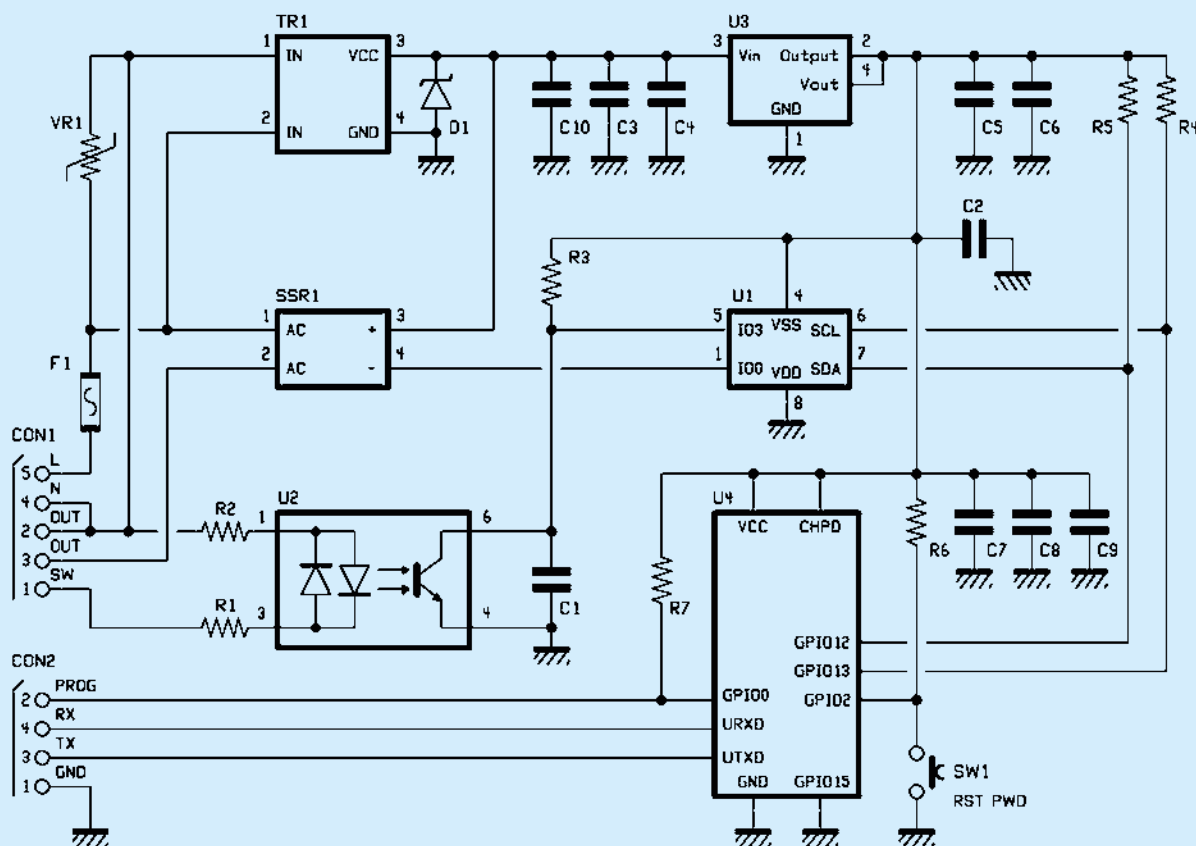


Fig. 5
Schermata della pagina web di configurazione dello Smart switch.





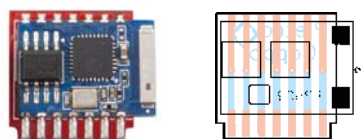
nella casella "DNS" inserite il numero IP del server DNS (domain name system) del vostro provider oppure il numero IP del vostro router. In alternativa potete utilizzare il numero IP di un servizio DNS pubblico come quello di Google, numero primario 8.8.8.8 e numero secondario 8.8.4.4, oppure quello di CloudFlare numero primario 1.1.1.1 e numero secondario 1.0.0.1, che tra i due risulta essere pure il servizio DNS pubblico gratuito con un ping medio

migliore.

Terminata la configurazione di sistema salvatela, cliccando sul pulsante Apply, e verificate se lo Smart switch si connette correttamente alla rete. Se avete scelto di ottenere il numero IP in modo dinamico, per conoscere quello che vi ha assegnato il server DHCP della vostra rete, avete bisogno del numero MAC che avete precedentemente annotato. Aprite il terminale "cmd" con i permessi di amministratore e digitate il comando "arp -d *" per cancellare tutti gli host memorizzati, poi fate un ping di broadcasting della vostra rete, ad esempio con il comando "ping 192.168.1.255", e infine lanciate il comando "arp -a". A terminale comparirà l'elenco dei numeri MAC di tutti i dispositivi presenti nella vostra rete, cercate quello corrispondente al numero MAC dello Smart switch, e annotate l'indirizzo IP che vi servirà per accedere alla pagina web del dispositivo. Lo Smart switch per funzionare con l'assistente vocale necessita delle credenziali lot. Nel nostro articolo spiegheremo come ottenere le credenziali lot per il servizio "Sinric pro" e quelle dell'assistente vocale Amazon Alexa. Con il browser aprite la pagina web <https://sinric.pro> vi registrate e fate il login, come visibile

piano di montaggio

ADAPTER



La scheda Adapter ospita un unico elemento: il modulo ESP03.

| piano di montaggio

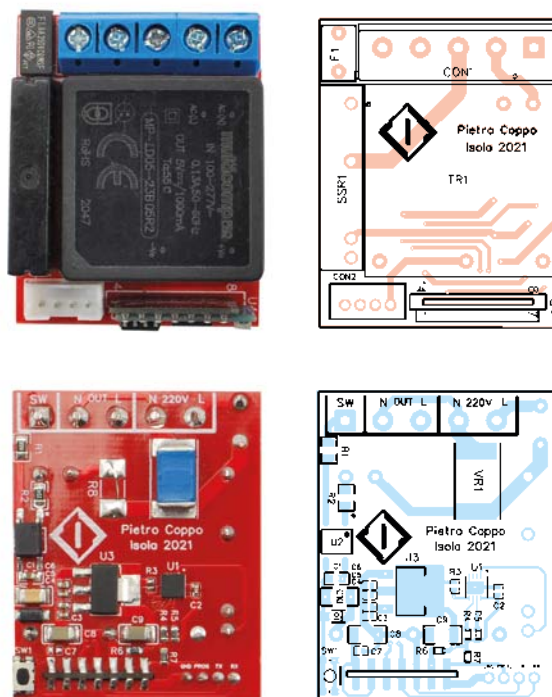
VERSIONE A SSR

Elenco Componenti:

R1, R2: 200 kohm 300 V (0805)
 R3: 470 kohm (0402)
 R4, R5: 2,2 kohm (0603)
 R6, R7: 10 kohm (0603)
 C1: 1 μ F 6,3 V (0603)
 C2, C3, C6, C7: 100 nF (0402)
 C4, C5: 10 μ F 6,3 V (0402)
 C8, C9: 100 μ F 6,3 V (1206)
 C10: 220 μ F 6,3 V (1206)
 D1: SMF5.0A
 VR1: V430CH8
 TR1: RAC05-05
 F1: 1,6 A
 U1: PCA9536TK
 U2: TLP184
 U3: NCP1117ST33T3G
 U4: ESP-03
 SSR1: G3MB-202P
 CON1: Morsettiera 5 poli P - 5 mm
 CON2: IP-1x4 P - 2 mm
 SW1: pulsante SMD 4.6X2.8mm

Varie:

- stampato S1586 (41x32,5 mm)



in **Fig. 6**. Una volta entrati, dal menu dispositivi cliccate sul pulsante "Aggiungi dispositivo", come visibile in **Fig. 7**. Si aprirà una nuova schermata nella quale inserirete il nome del dispositivo e una breve descrizione, dal menu a tendina "Tipo di dispositivo" scegliete "Switch", dal menu a tendina "Chiave di accesso al dispositivo" scegliete "default" e dal menu a tendina "Stanza" scegliete ad esempio "Living room", come visibile in **Fig. 8**.

Cliccando sulla pagina successiva potete attivare l'invio della notifica allo smartphone ad ogni accensione o spegnimento della luce.

Continuando potete impostare un timer di accensione o spegnimento e nell'ultima schermata potete attivare il calcolo approssimativo del consumo energetico. Dal menu "Credenziali" della pagina web Sinric.pro ricopiate le chiavi "App key" e "App Secret" e dal menu "Dispositivi" andate a ricopiare l'ID del dispositivo che volete comandare, come visibile in **Fig. 9**. Queste chiavi dovete inserirle nella pagina di configurazione dello Smart switch sezione "Iot credential" rispettivamente nei campi "Api Key", "Api Secrets" e "Device Id" (riferitevi alla **Fig. 5**). Nel vostro smartphone o tablet installate l'App Amazon Alexa che richiede un account

Amazon, se non lo avete creato. Entrati nell'App Amazon Alexa nel menu home cercate e installate la skill Sinric Pro. Spostatevi nel menu dispositivi e aggiungete un nuovo dispositivo cliccando sull'icona in alto a destra come visibile in **Fig. 10**.

Nella schermata successiva scegliete come

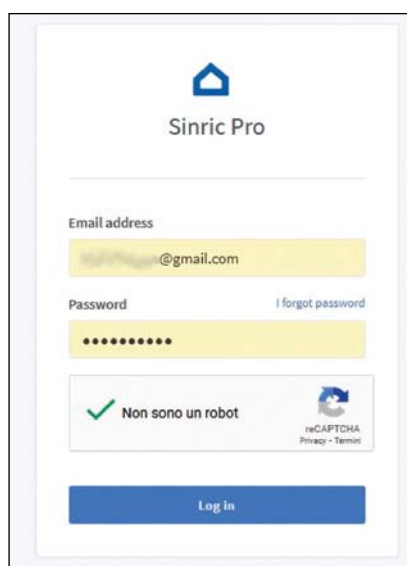
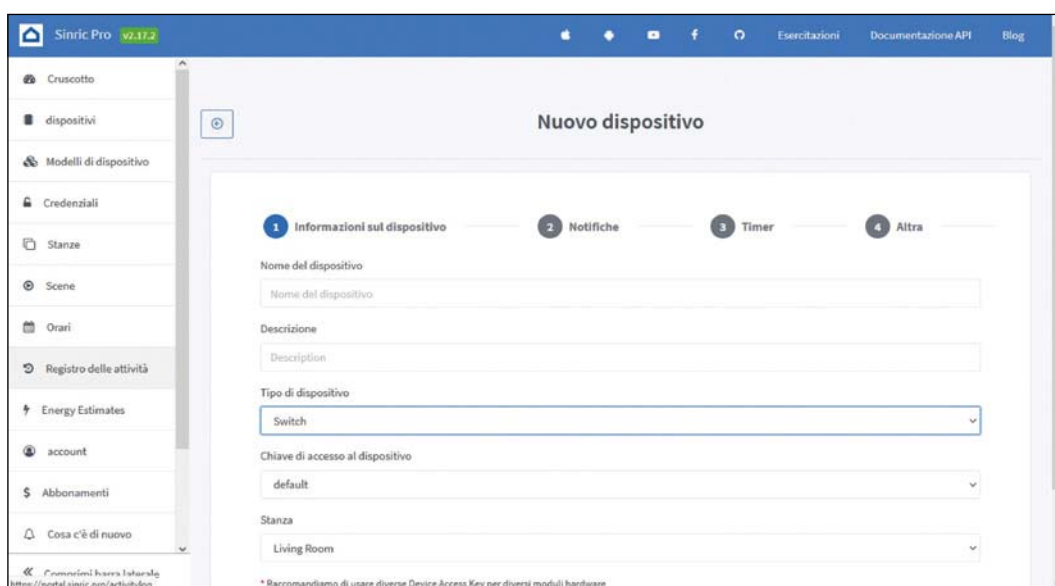


Fig. 6
Pagina di
login del sito
<https://sinric.pro>.

Fig. 7
Pagina con l'elenco
dei dispositivi
registrati sul sito
<https://sinric.pro>.



dispositivo "altro" e successivamente cliccate sul tasto "trova dispositivi", come visibile in **Fig. 10**, e attendete che il dispositivo sia rilevato. Selezionate la voce "Configura dispositivo" e, se desiderate tenere i dispositivi organizzati per gruppo, scegliete l'opzione "Seleziona gruppo". Potete scegliere tra Salotto, Camera da letto, Cucina, Sala da pranzo, Ufficio, ecc. oppure creare un gruppo personalizzato. Per terminare la configurazione seguite le istruzioni sullo schermo, vedi **Fig. 11**.

In alternativa all'app Amazon Alexa potete utilizzare quella di Google Home assieme all'assistente

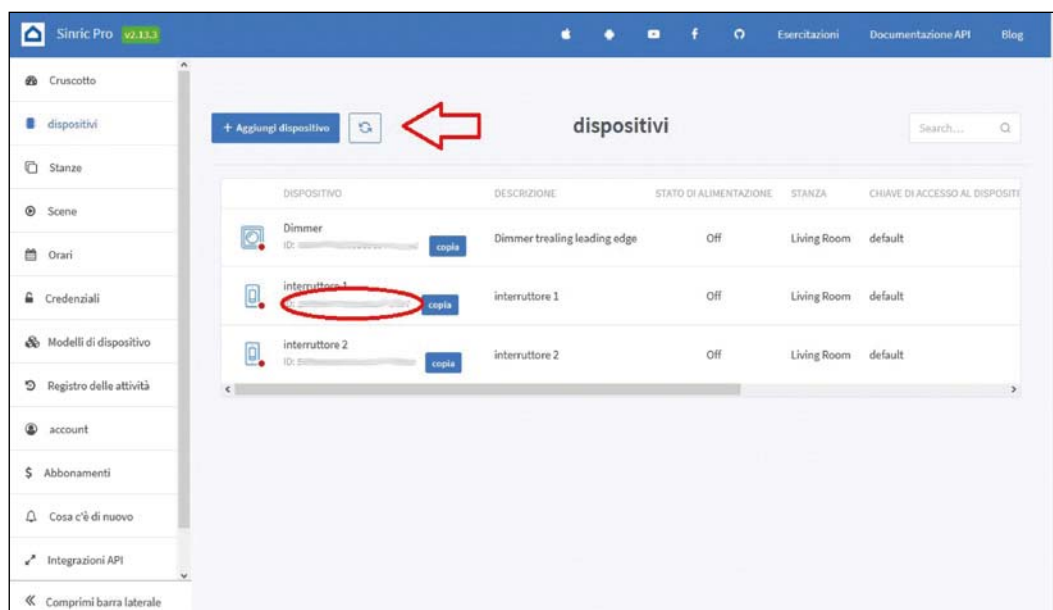
vocale Google assistant, **Fig. 12**, oppure l'app Sinric Pro come visibile in **Fig. 13**.

Inserite ora lo Smart switch in una scatola di derivazione oppure direttamente nella scatola da incasso dell'impianto domestico, seguendo il cablaggio riportato nello schema elettrico sul connettore CON1.

Facciamo presente che in base alla norma CEI 64/8 questa prevede una classificazione degli impianti elettrici secondo tre livelli:

- **Livello 1 - Base:** livello minimo previsto.
- **Livello 2 - Standard:** per unità immobiliari con

Fig. 8
Creazione di un
nuovo dispositivo
sul sito
<https://sinric.pro>.



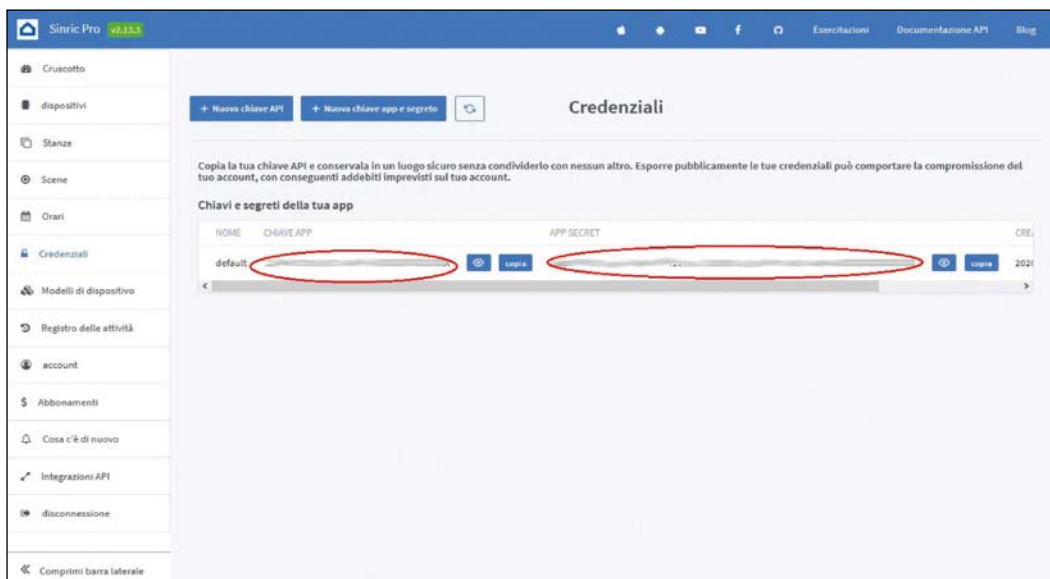


Fig. 9
Pagina delle
credenziali sul
sito <https://sinric.pro>.

una maggiore fruibilità degli impianti, tenendo conto delle altre dotazioni impiantistiche presenti.

- **Livello 3 - Domotico:** per unità immobiliari con dotazioni impiantistiche ampie e innovative, ad esempio per sistema bus, e in particolare, la domotica.

Se il vostro impianto civile elettrico non è a norma CEI 64/8 livello tre, nelle scatole da incasso rettangolari portafrutto manca sicuramente il cavo conduttore blu del neutro indispensabile negli

impianti domotici. Quindi per far funzionare il nostro dispositivo Smart dovete far arrivare questo conduttore dal quadro elettrico o dal punto luce che volete comandare alla scatola da incasso portafrutto. Nel caso ristrutturare la vostra abitazione e volete rendere l'impianto elettrico a norma CEI 64/8 livello tre, vi consigliamo di far installare all'eletttricista in ogni scatola da incasso anche il cavo conduttore del neutro, cioè quello di colore blu, così con una spesa irrisoria aumenterete anche il valore dell'immobile investendo per il futuro.

Tra i file scaricabili dal sito web www.elettronica.in.it

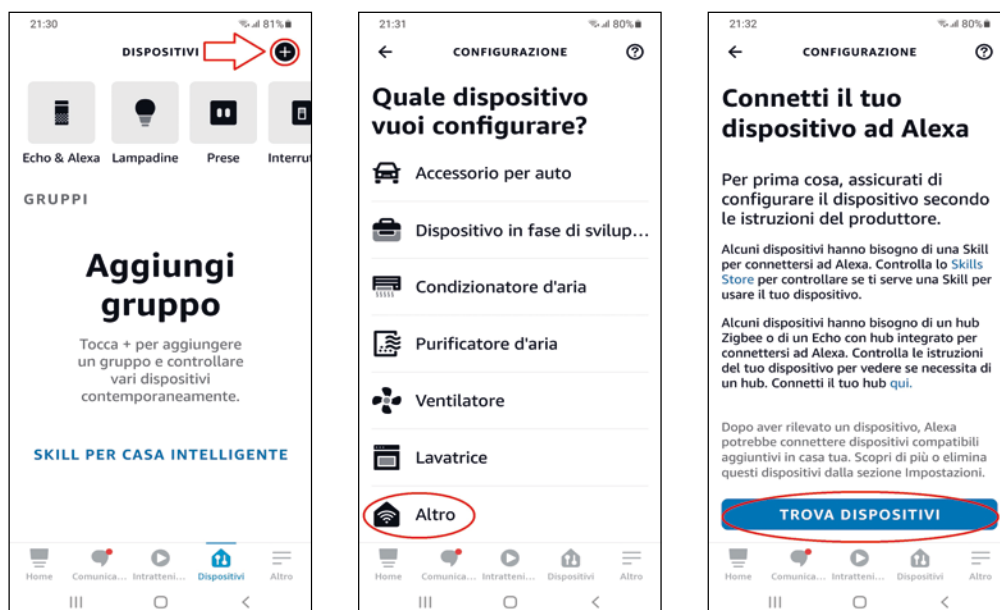
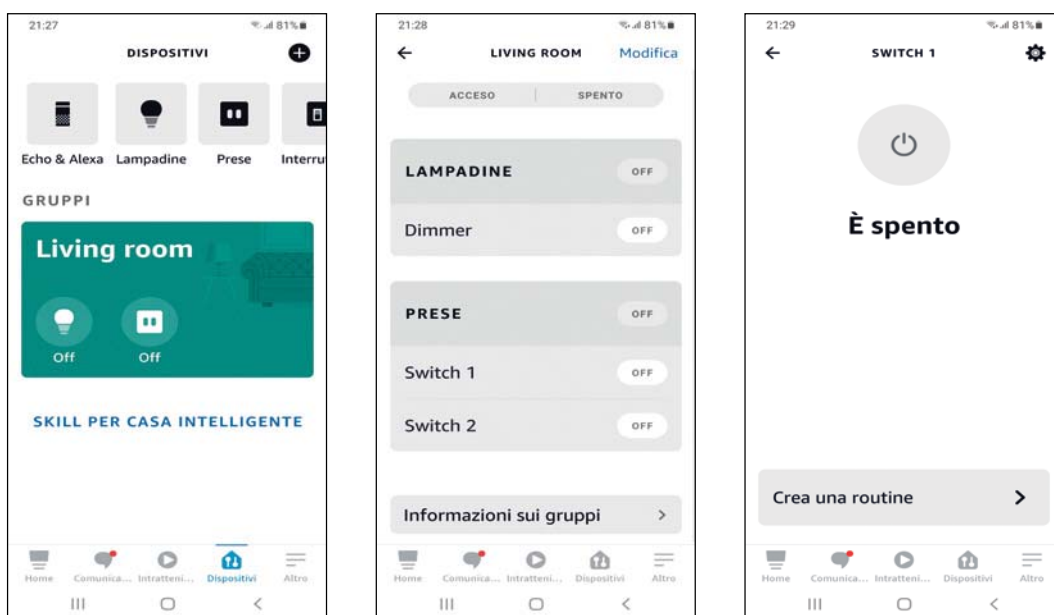


Fig. 10
Procedura per
aggiungere
un dispositivo
nell'App
Amazon Alexa.

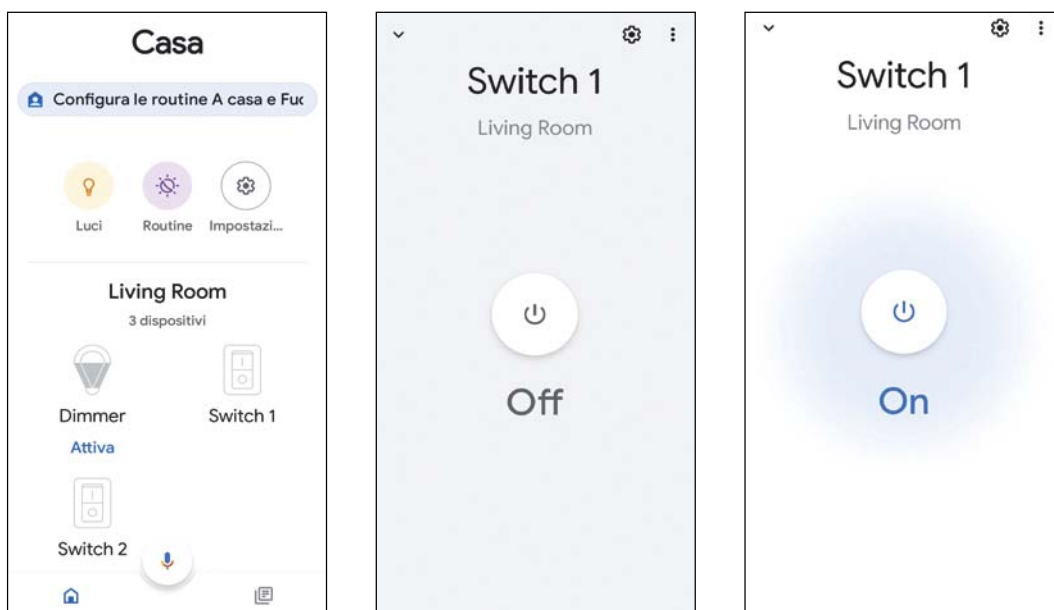
Fig. 11
Organizzazione
dei dispositivi in
gruppo e
interfaccia per
il controllo dello
Smart switch.



nella cartella STL trovate i file per la stampa 3D degli adattatori e frutti per i supporti Vimar Plana, Vimar Idea, Gewiss System, Gewiss Playbus e bTicino Living, che faciliteranno l'installazione dello Smart switch nel vostro impianto elettrico domestico. Tra i file STL troverete pure un contenitore di dimensioni contenute per il montaggio nelle scatole di derivazione oppure direttamente nelle scatole rettangolari da incasso. Facciamo presente che l'adattatore Vimar Idea viene montato su un copri foro originale al quale devono essere asportati precedentemente i segmenti di rinforzo come visibile in **Fig. 14**.

Aiutatevi con le tronchesine per tagliare i reofori dei componenti oppure un paio di pinze. Fate attenzione al corretto orientamento del coperchietto montato a incastro, in modo che il forellino sia posizionato sopra al pulsante SW1, per poterlo schiacciare. Tenendo premuto il pulsante per più di cinque secondi la lampadina collegata all'uscita lampeggerà tre volte come conferma. Il modulo si riavvierà con le impostazioni predefinite, impostando la password di sistema a "123456789" e cancellando l'SSID della rete memorizzata nonché le credenziali lot. A questo proposito vi informiamo

Fig. 12
Interfaccia dell'app
Google Home.



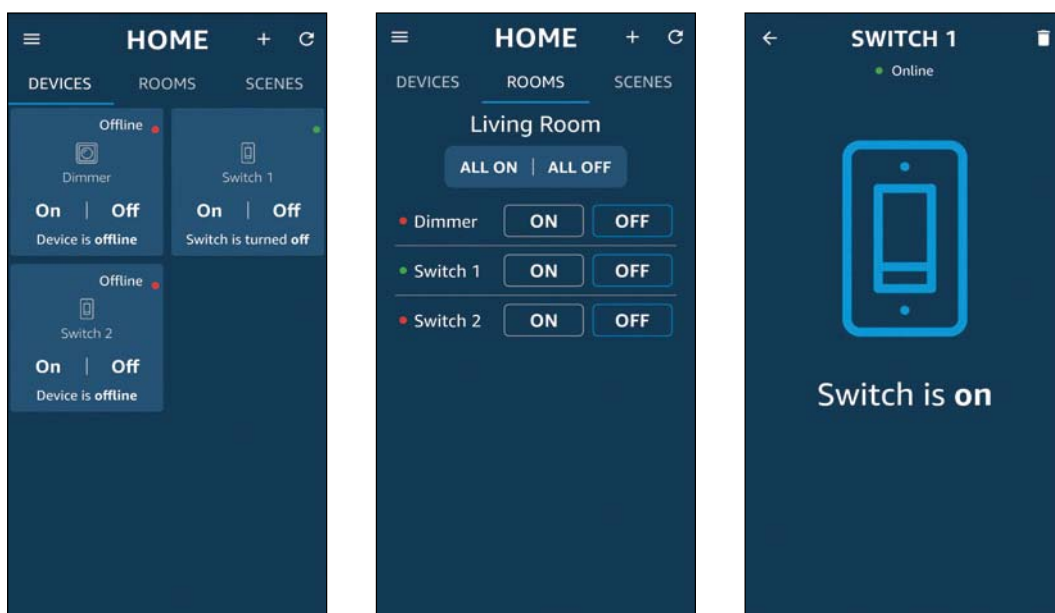


Fig. 13
Interfaccia
dell'app
Sinric Pro.

che la libreria `lotwebconf` di default visualizza il log sulla seriale in chiaro, compresi i parametri del vostro account lot. Per motivi di sicurezza vi consigliamo di disabilitare tale opzione inserendo all'inizio del file `lotWebConfSettings.h`, che si trova nella cartella SRC, la riga `"#define IOTWEBCONF_DEBUG_DISABLED"`. Per quelli che vogliono realizzare da soli il contenitore con la stampante 3D, consigliamo di impostare l'altezza del layer a un valore non superiore di 0,2 mm e, a seconda della versione del software di slicing, attivare l'utilizzo del riempimento a linee nel riempimento a griglia ad alta densità, altrimenti la parete del contenitore, spessa appena 1,5 mm, non verrà riempita correttamente presentando uno spazio vuoto. Benché con il filamento PLA si ottengono ottimi risultati, consigliamo comunque l'utilizzo di filamento in ABS che sopporta temperature di fusione più alte, oppure quello ABS V0 autoestinguente certificato dallo standard UL94 V0. In alternativa, per chi ha difficoltà a realizzare i contenitori, oppure non possiede una stampante 3D, può usufruire del servizio di stampa 3D messo a disposizione sul nostro sito all'indirizzo <https://www.futurashop.it/index.php?route=stampa3d/stampa3d> e chiedere un preventivo. In caso di conferma le stampe 3D verranno spedite nelle 24 ore successive al termine della stampa.

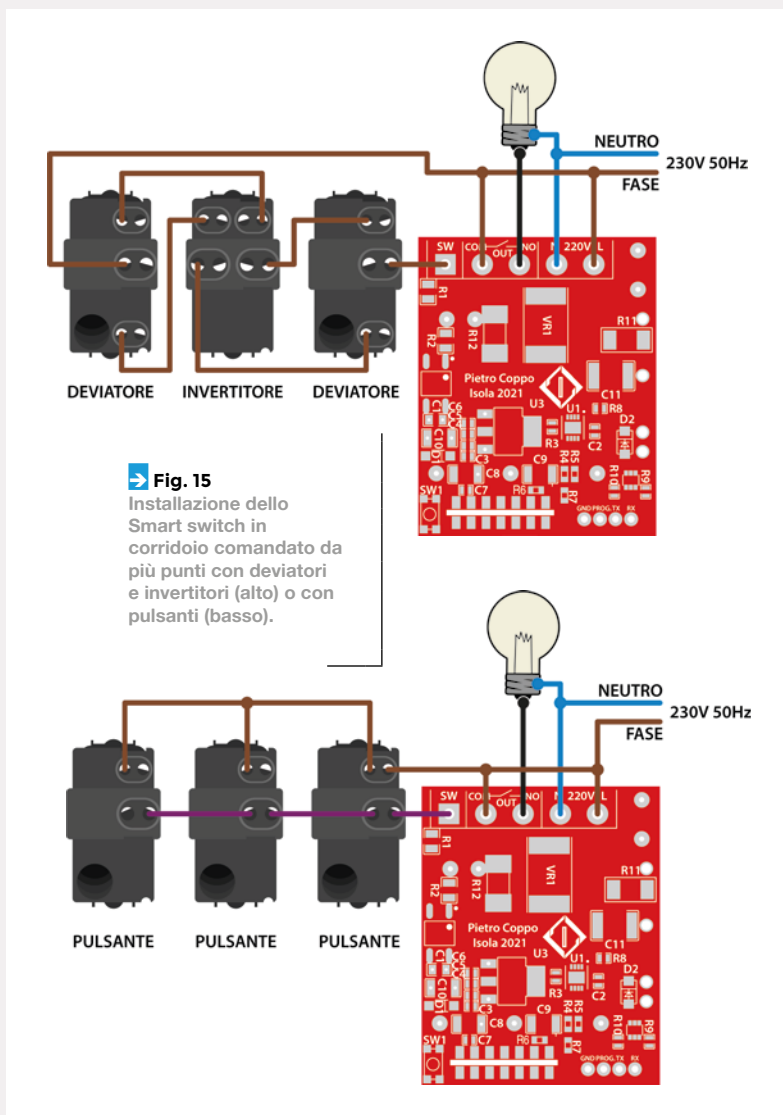
UTILIZZO DELLO SMART SWITCH

Per accedere alla pagina di configurazione dello Smart switch vi verrà richiesto il nome utente e la password. Come nome utente inserite "admin" e come password "123456789". Una volta

entrati nella sezione "Settings" potete impostare i parametri, come visibile in **Fig. 5**. In "Auto off timer" potete impostare il valore in minuti dell'auto-spegnimento della luce. Questa funzione è utile soprattutto per i più sbadati che dimenticano di spegnere la luce in garage o in cantina, rimanendo accesa anche per più giorni o settimane. Ad esempio impostando il timer a 240 minuti (4 ore), la luce si spegnerà da sola risparmiando sulla bolletta elettrica. Un altro utilizzo è quello di impostare il timer a 5 minuti e usare lo Smart switch per controllare l'accensione delle luci in un corridoio oppure sulle scale, rimpiazzando il vecchio temporizzatore meccanico, con un dispositivo più moderno e intelligente. La casella di spunta "Enable external pushbutton" vi permette di abilitare o disabilitare la regolazione mediante un pulsante collegato allo Smart switch. Abilitare tale funzione potrebbe essere utile durante un evento importante, per evitare che qualche maldestro ospite per sbaglio appoggiandosi sul pulsante spenga la luce,

Fig. 14
Le frecce indicano i
segmenti di plastica
da eliminare sul
copriforo.





oppure evitare che venga accesa se nella sala viene riprodotto un film. Viceversa l'attivazione della regolazione mediante il pulsante è utile soprattutto nelle installazioni senza connessione internet oppure, quando la connessione viene a mancare, permettendo la regolazione e lo spegnimento della luce che altrimenti rimarrebbe accesa.

La casella di spunta "Toggle switch" permette di definire se allo Smart switch collegherete un interruttore oppure un pulsante, se volete mantenere inalterato l'impianto elettrico domestico senza sostituire gli interruttori con pulsanti. Se volete installare lo Smart switch in una stanza, dove il punto luce è comandato da un interruttore oppure nel corridoio dove il punto luce è comandato da più posizioni lo potete fare. Alla fine dell'articolo nelle **Fig. 15** e **Fig. 16** trovate alcuni schemi esemplificativi di collegamento dello Smart switch.

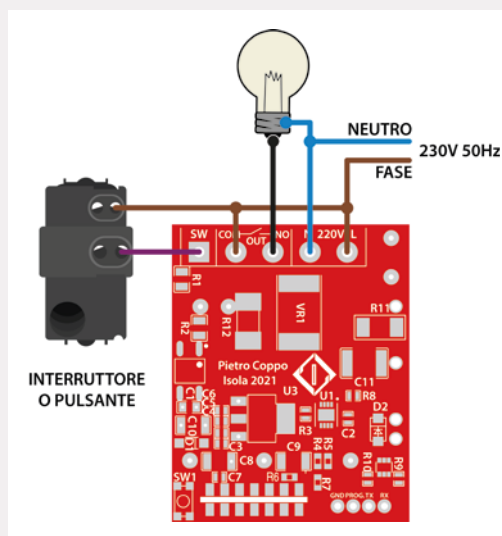
La casella di spunta "Enable assistant control" vi permette di abilitare o disabilitare il controllo mediante assistente vocale Amazon Alexa o Google assistance. La casella di spunta "Enable web control" vi permette di abilitare il controllo dell'accensione mediante pagina web. Nella casella di spunta "Turn on at startup" potete scegliere se la luce deve accendersi o rimanere spenta all'avvio. Se volete illuminare il portico o il cortile di casa e desiderate che la luce rimanga accesa tutta la notte anche dopo un black out abilitate questa funzione. L'ultima casella di spunta che ci rimane da analizzare è il "Reboot device on save", che permette di abilitare il riavvio del dispositivo dopo il salvataggio della configurazione con il tasto "Apply".

Di seguito riportiamo alcuni comandi riconosciuti dall'assistente vocale Amazon Alexa e Google assistant. Supponiamo che in <https://sinric.pro> al nostro dispositivo abbiamo assegnato il nome "Interruttore 1".

Per accendere la luce pronunciate: "Alexa, accendi Interruttore 1" oppure "Hey Google, accendi Interruttore 1".

Per spegnere la luce pronunciate: "Alexa, spegni interruttore 1" oppure "Hey Google, spegni Interruttore 1".

Per saperne di più e conoscere tutti i comandi vocali riconosciuti dall'assistente vocale Amazon Alexa, nell'App Amazon Alexa selezionate l'icona in alto a sinistra, scegliete la voce "Cose da provare" e cliccate l'opzione "Casa intelligente", se in casa avete un dispositivo Amazon echo dot, comandare lo Smart switch sarà ancora più comodo e facile. Per personalizzare l'esperienza e le impostazioni dell'assistente vocale Google Assistant, andate in



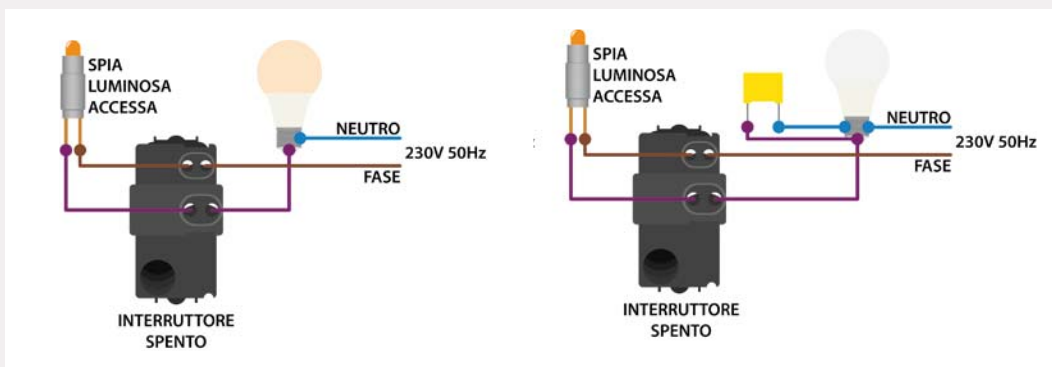


Fig. 17
Indesiderato fenomeno con le lampadine LED (sinistra), inserendo in parallelo alla lampadina LED un condensatore filtro il difetto scompare (destra).

Google Home e nel menu funzionalità del menu impostazioni cliccate su assistente Google. Oppure consultate la guida cliccando sull'icona in alto a destra. Prima di terminare vogliamo informarvi che non tutte le lampade a LED in commercio sono uguali. Quelle a basso costo con un alimentatore di bassa qualità se collegate in una configurazione come quella di **Fig. 17**, con in serie una spia luminosa, la lampadina rimarrà lievemente accesa oppure incomincerà a sfarfallare. Se sostituiamo la lampadina con una di qualità superiore come quelle dimmerabili, la lampadina rimarrà spenta. Per ovviare a questo inconveniente basta collegare in parallelo alla lampadina un condensatore filtro x2 da 0,22 uF max 0,33 uF. Poiché la versione dello Smart switch EMR possiede in parallelo ai contatti del relé uno snubber che li protegge dai carichi capacitivi e induttivi, provoca questo indesiderato fenomeno. Per correggere il difetto avete tre opzioni, sostituire la lampadina LED con una dimmerabile molto più costosa, eliminare lo snubber, oppure aggiungere un condensatore in parallelo al carico come raffigurato in **Fig. 18**. Con la versione a relé allo stato solido questo difetto non è presente.

CONCLUSIONI

Per chi fosse interessato a realizzare da solo lo stampato, facciamo presente che sul nostro sito web www.elettronica.in.it trovate disponibili per il download i file Gerber di entrambe i circuiti stampati a doppia faccia ottenibile per fotoincisione, nonché il firmware e la BOM (Bill Of Material) dettagliata. Con questi due dispositivi speriamo di aver soddisfatto tutti i nostri lettori che volevano realizzare da soli un interruttore smart affidabile, versatile, gestibile e comandabile da remoto con lo smartphone o con l'assistente vocale, capace di comandare carichi resistivi o induttivi, nonché essere di dimensioni contenute per facilitare l'installazione negli impianti domestici. Ricordiamo che il progetto Smart switch fa parte di

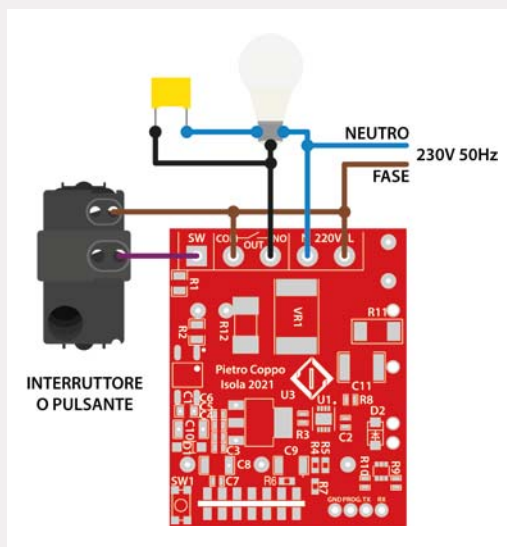


Fig. 18
Cablaggio del condensatore filtro in uscita allo Smart switch in caso di malfunzionamenti con le lampade a LED.

una serie di moduli e attuatori domotici progettati e realizzati dagli alunni dell'Istituto Tecnico Pietro Coppo Isola Slovenia che verranno pubblicati sulla nostra rivista nei prossimi numeri. Per ulteriori domande, richieste o delucidazioni agli autori del progetto rivolgetevi alla redazione di Elettronica In che provvederà a mettervi in contatto. □



Cosa occorre?

I componenti principali utilizzati in questo progetto sono disponibili presso Futura Elettronica. Il modulo ESP8266 (cod. ESP03) è in vendita a Euro 8,00, il programmatore/debugger (cod. YM203) è disponibile a Euro 2,50, il convertitore USB/TTL (cod. UC00A) costa Euro 15,50. I PCB possono essere realizzati utilizzando il servizio PCBPRODUCTION disponibile sul sito www.futurashop.it I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

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

I tuoi PCB a partire da

1€

Con il servizio PCB di Futura Elettronica, potrai realizzare facilmente il circuito stampato del tuo progetto!

Vai su www.futurashop.it, inserisci nella ricerca il codice "PCBPRODUCTION" e carica il tuo Gerber scegliendo le dimensioni e le caratteristiche preferite.

Prezzi IVA inclusa.



cod. PCBPRODUCTION

COLORE SERIGRAFIA



BIANCA



NERA

La serigrafia nera è disponibile solamente per PCB bianchi e gialli.

COLORE SOLDER



VERDE



ROSSO



GIALLO



BLU



BIANCO



NERO

SPESSORE (mm)

mm



0.2

0.4

0.6

0.8

1.0

1.2

1.6

2.0

2.4

2.8

3.2

NUMERO DI LAYERS

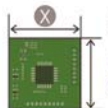
1 Layer

2 Layers

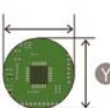
4 Layers

6 Layers

DIMENSIONI (cm)



Single Size



Pannel Size

- 5x5 - 5x10 - 5x15 - 5x20 - 5x25 - 5x30
- 10x10 - 10x15 - 10x20 - 10x25 - 10x30
- 15x15 - 15x20 - 15x25 - 15x30
- 20x20 - 20x25 - 20x30
- 25x25 - 25x30
- 30x30

ECCO QUALCHE ESEMPIO:

PCB con 1 layer
da 5x5 cm
spessore 1,6 mm
solder verde
serigrafia bianca
€ 1 cad.

PCB da 2 layer
da 10x10 cm
spessore 1,6 mm
solder verde
serigrafia bianca
€ 6 cad.

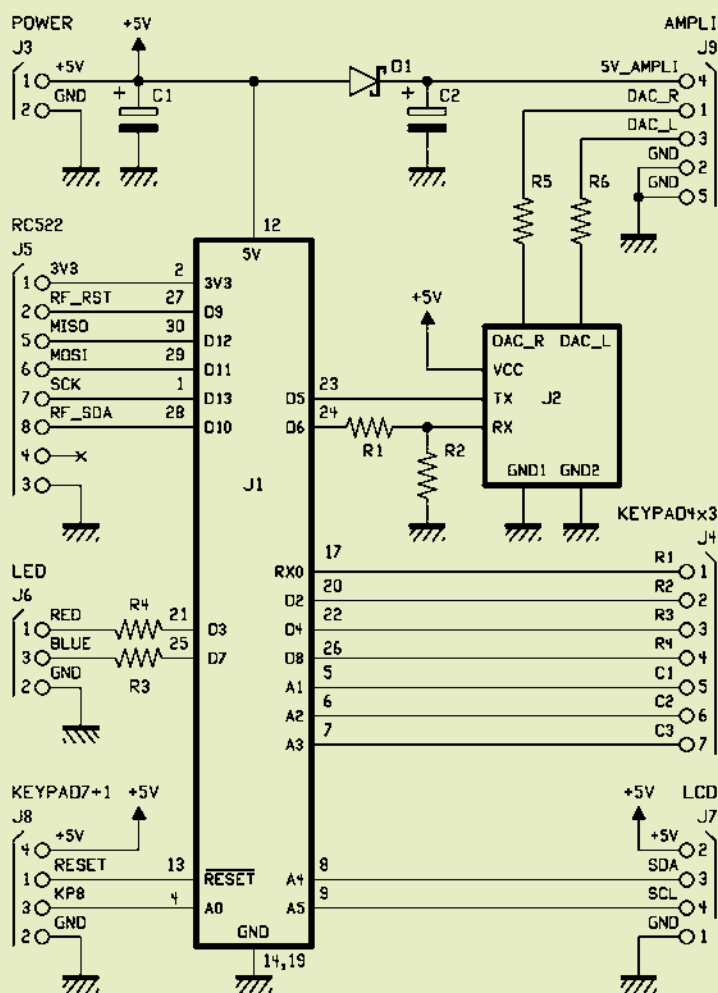
UN JUKE-BOX MP3



del Prof. **MICHELE MENNITI**

Realizziamo un Juke-box in chiave moderna, basato sul modulo riproduttore MP3 DFR0299 e gestito mediante identificazione RFID.

Chi ha vissuto la propria gioventù negli anni '60-'70-'80 non può non ricordare il mitico Juke-box, la macchina musicale a gettoni, presente in genere all'esterno dei locali pubblici, come bar e ristoranti, ma anche in alcune scuole ed Università, nelle sale di ristoro. Bastava inserire una moneta, selezionare il pulsante riportante il nome della canzone desiderata e, come per magia, dietro la lastra trasparente, in vetro o in plexiglass, si metteva in moto un meccanismo che faceva ruotare il box contenente i 45 giri in vinile, prelevava quello prescelto, lo depositava sul piatto rotante ed infine abbassava la testina, e partiva la musica. Niente cuffiette, ma due grandi altoparlanti integrati nel mobile, che diffondevano i brani più in voga in quel periodo, a beneficio di tutti i presenti, molti ascoltavano, qualcuno ballava, qualcuno si lamentava del volume troppo alto, ma la socializzazione era garantita.



Nel film di Lucio Fulci "I ragazzi del Juke-box" (Italia, 1959), Adriano Celentano cantava, nel suo tipico stile rock, la canzone "Juke-box" il cui ritornello faceva così:

...la felicità costa un gettone per i ragazzi del Juke-box, la gioventù, la gioventù la compra per cinquanta lire e nulla di più, basta un dolce blues e una canzone per i ragazzi del Juke-box, ballando qua ballando là ognuno trova la sua gran felicità..."

Il mito del Juke-box è racchiuso in queste quattro frasi e fu magistralmente rappresentato nella famosissima serie televisiva americana "Happy days", quella in cui il celeberrimo protagonista Fonzie sceglieva la canzone da ascoltare ma, invece di introdurre la monetina, di cui era perennemente sprovvisto, dava all'apparecchio un potente pugno, ottenendo comunque il risultato desiderato! Periodicamente il gestore apriva questa incredibile macchina, con decine di curiosi che si accalcavano

per vedere l'interno di questa meraviglia, e sostituiva parte dei dischi e le relative etichette dei pulsanti, per la gioia di alcuni ed il disappunto di altri. Il mobile, grandissimo, inizialmente realizzato in legno finemente lavorato, successivamente in metallo e plastica, aveva un pannello trasparente superiore che mostrava la meccanica, ed altri pannelli coloratissimi, illuminati internamente da comuni lampadine ad incandescenza che, nei modelli più recenti, lampeggiavano, replicando gli effetti dei flipper, i famosissimi giochi di abilità dell'epoca.

L'HARDWARE

Il progetto che presentiamo oggi è strettamente legato ai due articoli "Tecnologia RFID e il Modulo RC522" e "Modulo MP3 DFR0299" presentati rispettivamente sui numeri 258 e 259 della nostra rivista, in quanto abbiamo utilizzato tutti i componenti presentati in quei due lavori, aggiungendo una tastiera a membrana 4x3 e sostituendo Arduino Uno con Arduino Nano, in modo da poter realizzare un apposito PCB professionale, al quale collegare tutte queste periferiche.

L'aver affrontato tutte le problematiche relative alla tecnologia RFID (vedi anche l'articolo "Tecnologia RFID, NFC: teoria e pratica" sul n. 254), al modulo RFID-RC522, al keypad 7+1, ed avendo presentato uno studio approfondito sul modulo MP3 DFR0299, ci permette ora di dedicarci esclusivamente a questo progetto, dedicandoci solo alle sue specificità. Chiaramente vi invitiamo a dare una lettura agli articoli suindicati, se volete approfondire il funzionamento del circuito. L'unica piccola novità, rispetto all'articolo precedente, è l'aggiunta di due resistori da 100 Ohm in serie alle uscite audio stereo del lettore MP3, utili per limitare il carico sul modulo da parte dell'amplificatore BF.

Il nostro progetto si basa su un PCB con due coppie di slot header per l'innesto di un Arduino Nano (codice ARDUINONANO sul sito www.futurashop.it), che programmeremo opportunamente e di un modulo DFR0299 della DFRobot (codice DFR0299). Sono inoltre presenti sette connettori, uno per l'alimentazione (JST a 2 pin) e 6 per le periferiche esterne: uno strip a 8 pin per l'innesto della tastiera a membrana 4x3, due prese MK a 4 pin per il keypad 7+1 e per il display LCD 16x2 con interfaccia I²C, uno strip maschio a 8 pin per il modulo RFID-RC522 (codice VMA405N), una presa MK a 3 pin per i due LED di stato ed una presa MK a 5 pin per il collegamento al modulo amplificatore (PAM8403POT), che dovrà essere dotato di due altoparlanti da 0,5 Watt 8 Ohm (codice MLS1).

Il tutto potrà essere montato nel contenitore che abbiamo realizzato appositamente. Il nostro primo prototipo, è stato realizzato su breadboard, un lavoro non molto complesso, ma che ha richiesto attenzione a motivo dei numerosi collegamenti con cavetti jumper (**Fig. 1**). Sono visibili, da sinistra a destra, il modulo MP3, il modulo RFID-RC522 e Arduino Nano, mentre esternamente, collegati mediante cavetti, si vedono il keypad 7+1 (questa volta usato in forma completa), il display LCD e la tastiera a membrana. Da notare infine, subito sopra il keypad, la presa jack stereo volante, per il collegamento, nel caso del prototipo, ad una coppia di casse amplificate da PC. A proposito del modulo MP3, è importante chiarire che il nostro Juke-box funzionerà solo con il modulo DFPlayer Mini DFR0299 della DFRobot, cioè il lettore MP3 originale; abbiamo infatti testato alcuni moduli cinesi "compatibili" con esito negativo, perché i codici inviati in risposta alle richieste dello sketch sono diversi rispetto agli originali ed in alcuni casi uguali tra loro, rendendo quindi impossibile l'identificazione dello stato del lettore.

FUNZIONAMENTO GENERALE

Compatibilmente con le possibilità offerte dal semplice hardware che abbiamo utilizzato, abbiamo cercato di ricreare, in chiave moderna, il funzionamento degli apparecchi originali. All'accensione del circuito il nostro juke-box si trova in modalità **"Stop"**, la scelta del singolo brano può essere effettuata digitando sulla tastiera a membrana il relativo numero che, se inferiore a quattro cifre, dovrà essere seguito dal simbolo "#". Invece di un'ingombrante gettoniera per monete abbiamo previsto il lettore RC522, quindi il gettone sarà rappresentato da un qualsiasi TAG RFID (Card, portachiavi, supporto con adesivo, ecc.) che, una volta avvicinato al lettore, se sarà riconosciuto come autorizzato, farà partire il brano prescelto. Durante l'esecuzione sarà possibile regolare il **"Volume"**, mediante i due specifici pulsanti del keypad 7+1 oppure mettere l'ascolto in **"Pausa"**, mediante un altro pulsante, riprendendo il quale si potrà ritornare in **"Play"** (**Fig. 2**). Abbiamo pensato di creare una sorta di abbonamento, sfruttando due potenzialità della libreria, che approfondiremo in questo articolo: l'esecuzione di tutti i brani presenti, a ciclo continuo (**"Loop"**) oppure senza un ordine prefissato (**"Random"**). Per ottenere queste due condizioni sarà necessario disporre di un TAG RFID appositamente autorizzato per l'una o l'altra modalità. In questi due casi, oltre ai tre pulsanti già visti, funzioneranno anche quelli

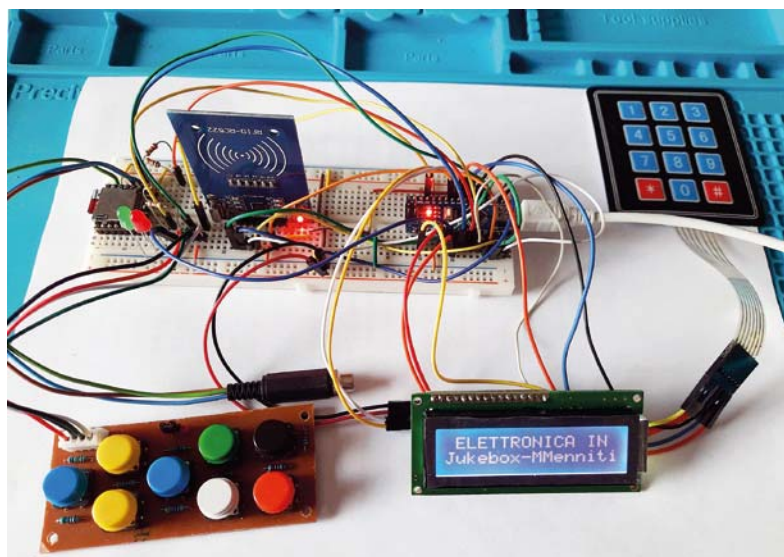
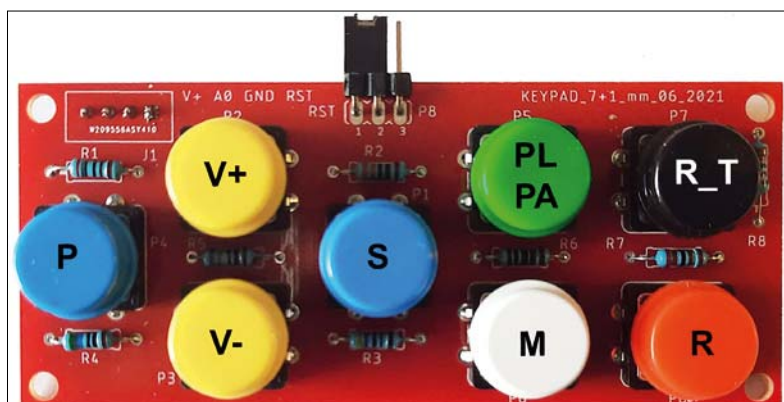


Fig. 1
Il primo prototipo su breadboard.

per cambiare brano prima della fine dell'esecuzione, sono i pulsanti **"Precedente"** e **"Successivo"**. Naturalmente i TAG dovranno essere preventivamente autorizzati, per questa operazione è prevista la necessità di un ulteriore TAG **"Amministratore"**, che avrà pieni poteri sul Juke-box: infatti, oltre a poter eseguire qualsiasi brano in modalità singola con **"Play/Pausa"**, mediante il pulsante **"Mode"** potrà selezionare a piacimento le modalità **"Loop"** e **"Random"**, e potrà, in qualsiasi momento, azzerare i permessi dei TAG e ricrearne altri. Il pulsante **"RESET"** è invece sempre disponibile per tutti e, resettando Arduino Nano, farà ripartire lo sketch, tornando in condizione **"Stop"**. Il nostro sketch prevede la possibilità di memorizzare nella EEPROM 4 TAG: un

Fig. 2
Il Keypad 7+1.

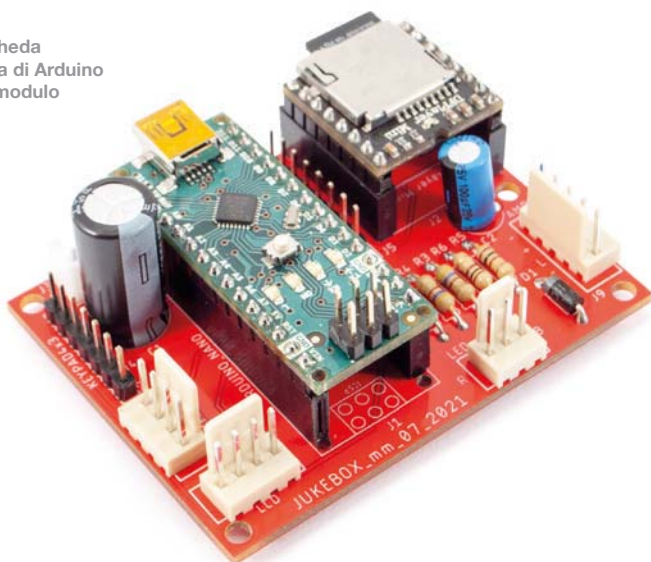


P =brano precedente - S =brano successivo
V+ =aumenta volume - V- =diminuisce volume
PL-PA =Play/Pausa - M =Mode (Loop/Random)
R_T =Reset TAG - R =Reset generale

Amministratore, un utente per brano singolo, un utente per esecuzione in loop ed un utente per esecuzione in random. I due LED servono per indicare la condizione di **"Stop"** oppure di rifiuto autorizzazione TAG (LED rosso) e la condizione di esecuzione di un brano in una qualsiasi modalità (LED blu). Il modulo amplificatore esterno è di tipo stereo, con potenza di 3+3 Watt e volume regolabile. Per quanto riguarda l'alimentazione, abbiamo previsto un ingresso specifico per un alimentatore esterno stabilizzato da 5 V, cioè una presa PLUG DC collegata al JST a 2 pin; l'alimentatore deve essere in grado di erogare una corrente fino a 550 mA garantendo una buona stabilità della tensione; consigliamo di adottare un modello da 2 Ampere nominali, possibilmente di tipo lineare, in quanto uno switching potrebbe provocare qualche rumore di fondo. Il circuito può essere alimentato anche mediante la presa USB di Arduino Nano, tramite collegamento ad un PC, ma tenendo presente che una porta USB normale eroga al massimo 500 mA. Qui di seguito riportiamo le caratteristiche tecniche e funzionali del nostro progetto Juke-box.

- Alimentazione: 5 V mediante PLUG DC oppure mediante porta USB; assorbimento massimo: 550 mA.
- Modulo MP3 DFR0299 con frequenze di campionamento supportate (kHz): 8 – 11,025 – 12 – 16 – 22,05 – 24 – 32 – 44,1 – 48, flash memory di 64 Mb ed equalizzatore a 6 livelli.
- Uscita DAC a 24 bit, gamma dinamica 90 dB e rapporto segnale-rumore (SNR) 85 dB.
- 255 file MP3 memorizzabili in una card SD micro (FAT 16 o 32), cartella "mp3".
- Modulo RFID-RC522 con funzione di "gettoniera" ed indicazione autorizzazione a LED.

→ La scheda completa di Arduino Nano e modulo MP3.



- 4 TAG RFID memorizzabili in EEPROM: Amministratore, Singolo, Loop, Random
- Keypad con 8 tasti.
- Pulsante TAG reset per l'azzeramento della EEPROM.
- Esecuzione brani in modalità: singola, totale in loop o totale in random con indicatore a LED.
- Scelta del file da eseguire mediante tastierino numerico a membrana.
- Regolazione Volume digitale + e - da 0 a 30, mediante due pulsanti.
- Pulsanti Precedente e Successivo.
- Pulsante Play/Pausa.
- Pulsante Mode per la selezione della modalità di esecuzione.
- Pulsante RESET per Arduino Nano.
- Visualizzazione delle informazioni di funzionamento su display LCD 16x2 I²C retroilluminato.
- Amplificatore audio con volume regolabile e coppia di altoparlanti da 0,5 Watt, oppure collegamento ad impianto audio esterno.

LA TASTIERA A MEMBRANA 4X3

L'unico componente di cui non abbiamo parlato negli articoli precedenti è la tastiera a matrice con tasti a membrana. La **Fig. 3** mostra la disposizione dei tasti e l'organizzazione della matrice, molto importante ai fini della realizzazione dello sketch. Le 4 righe sono colorate in blu e comprendono rispettivamente i gruppi di tasti "1 2 3", "4 5 6", "7 8 9" e "* 0 #", mentre le colonne sono colorate in rosso e comprendono rispettivamente i gruppi di tasti "1 4 7 *", "2 5 8 0" e "3 6 9 #". Il principio di funzionamento è semplice: nel momento in cui si preme un tasto, vengono messe in cortocircuito una riga ed una colonna. Qualche esempio: premendo il tasto "1" si cortocircuitano la riga R1 e la colonna C1, premendo il tasto "5" si cortocircuitano la R2 e la C2, premendo invece il tasto "0" si cortocircuitano R4 e C2. Come è facile intuire ogni tasto realizza una combinazione univoca riga-colonna che permette di identificarlo istantaneamente, quindi naturalmente sono necessari ben 7 pin di Arduino Nano per la sua gestione, le relative azioni vengono poi gestite dallo sketch. La libreria necessaria per l'uso di questa tastiera è disponibile sul playground di Arduino al link: <https://github.com/Chris-A/Keypad>, ma in ogni caso è disponibile nella sezione download del sito www.elettronica.in.

L'AMPLIFICATORE 3+3 WATT

Reperibile sul sito www.futurashop.it col codice PAM8403POT (**Fig. 4**), l'amplificatore audio che

abbiamo adottato è basato sul chip PAM8403, è un amplificatore stereo in classe D in grado di erogare una potenza massima di 3+3 Watt su 4 Ohm.

Dispone di potenziometro per la regolazione del volume ed interruttore on/off.

Alle sue uscite abbiamo applicato due altoparlanti da 0,5 Watt 8 Ohm, con diametro di 50 mm. Sono reperibili sul sito www.futurashop.it col codice MLS1 (Fig. 5).

Occorre fare attenzione al collegamento degli altoparlanti ai fili del cavetto MK, in quanto bisogna rispettare le polarità + e -.

Il sistema audio previsto da noi naturalmente non potrà garantire una qualità Hi-Fi, quindi, in alternativa ad esso, sarà possibile collegare le uscite audio stereo ad un sistema di casse amplificate esterne, prelevando il segnale direttamente dal connettore di uscita audio (J9) del PCB.

CIRCUITO PRINCIPALE

Diamo un rapido sguardo allo schema elettrico del circuito principale del Juke-box. Durante le nostre prove abbiamo riscontrato dei picchi di assorbimento, nell'esecuzione ad alto volume, da parte del modulo MP3, che influenzavano anche il comportamento dell'amplificatore audio; è stato quindi necessario separare le alimentazioni e fornire due serbatoi di energia per assorbire questi picchi. A tale funzione assolvono il condensatore elettrolitico C1, per il modulo MP3, ed il diodo schottky D1 con il condensatore elettrolitico C2, per l'amplificatore. Sempre per l'audio, abbiamo aggiunto, in serie alle uscite dei canali destro e sinistro (DAC_R e DAC_L) due resistori da 100 Ohm (R5 e R6) per separare il DAC del modulo DFR0299 dall'amplificatore ed evitare sovraccarichi.

La seriale UART del modulo MP3 lavora a livello logico TTL 3,3 V, nonostante il modulo sia alimentato a 5 V; questo comporta un problema di interfacciamento con Arduino Nano, che invece lavora con segnali di 5 V. I segnali con ampiezza di 3,3 V inviati dal pin TX del DFR0299 vengono correttamente riconosciuti da Arduino, quindi non rappresentano un problema; invece è stato necessario adeguare l'ampiezza dei segnali provenienti da Arduino e diretti al pin RX del DFR0299; per tale ragione abbiamo interposto il partitore resistivo costituito da R1 e R2, che riduce il livello del segnale a 3 V, come dimostra la formula

$$V_{rx} = \frac{V \cdot R2}{R1 + R2} = \frac{5 \cdot 15000}{10000 + 25000} = 3V$$

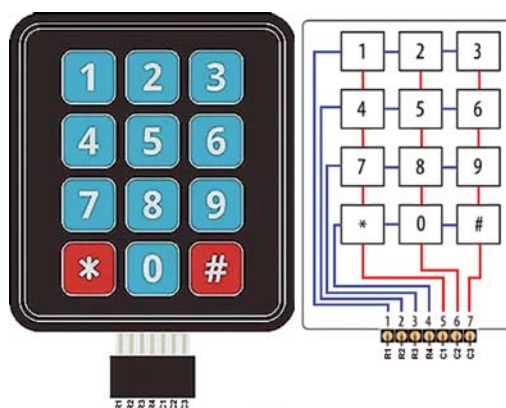


Fig. 3
La tastiera a membrana 4x3.



Fig. 4
L'amplificatore audio stereo.

in cui V è il livello del segnale in uscita da Arduino UNO (5 V) mentre V_{rx} è il livello in uscita dal partitore. Gli ultimi due resistori, R3 e R4, svolgono la tipica funzione di limitatori di corrente per il due LED blu e rosso.

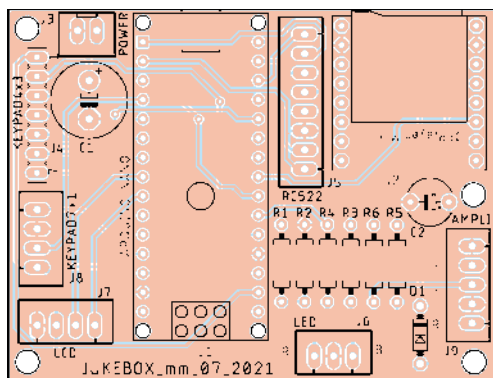
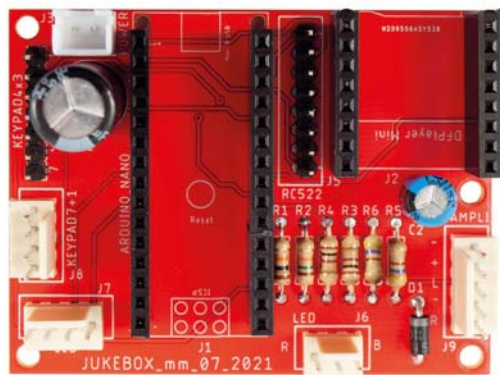
Gli altri componenti sono solo connettori, header, strip, JST o MK, tutti con passo 2,54 mm, per l'innesto di Arduino Nano e del modulo MP3, e per il collegamento con le periferiche esterne.

REALIZZAZIONE PRATICA

Per lo schema elettrico e il montaggio del keypad e dell'interfaccia I²C sul retro del display LCD, rimandiamo all'articolo presentato sul numero 259; ora ci soffermeremo esclusivamente sul circuito del

Fig. 5
I due altoparlanti da 0,5 Watt 8 Ohm.





Elenco Componenti:

- R1: 10 kohm
- R2: 15 kohm
- R3, R4: 100 ohm
- R5, R6: 470 ohm
- C1: 1000 μ F 16 VL elettrolitico
- C2: 100 μ F 16 VL elettrolitico
- D1: 1N5819 schottky
- J1: strip femmina 15 vie (2 pz.)
- J2: strip femmina 8 vie (2 pz.)
- J3: connettore JST 2 poli M 2,54 mm
- J4: strip maschio 7 vie
- J5: strip maschio 8 vie
- J6: connettore 3 poli MK maschio 2,54 mm
- J7, J8: connettore 4 poli MK maschio 2,54 mm
- J9: connettore 5 poli MK maschio 2,54 mm
- Varie
 - LED blu 5 mm
 - LED rosso 5 mm
 - Arduino Nano
 - modulo DFPlayer DFR0299
 - modulo RFID-RC522
 - modulo keypad 7+1 (v. elenco componenti separato)
 - modulo amplificatore 3+3 W PAM8403POT
 - display LCD 16x2 I2C
 - tastiera a membrana 4x3
 - altoparlante 3 W- 4 ohm (2 pz.)
 - presa DC 2,1 mm da pannello
 - contenitore
 - stampato S1589 (50x65 mm)

Juke-box. Vi ricordiamo che nella sezione download del sito www.elettronica.in.it troverete i file gerber per la realizzazione dei due PCB (keypad e Juke-box). Dopo aver montato i sei resistori R1÷R6, occorre aggiungere semplicemente le varie prese e connettori previsti sul PCB. Una particolare attenzione va posta solo nel montaggio delle due coppie di header destinate ad ospitare Arduino Nano ed il DFR0299, devono essere perfettamente perpendicolari rispetto al PCB, altrimenti potrebbero sorgere difficoltà nell'innesto della board e del modulo. Una volta completate le saldature si potranno inserire i due moduli nelle coppie di header, seguendo il piano di montaggio e facendo attenzione al verso di inserimento. Quindi occorrerà collegare i vari cavetti JST e MK alle periferiche, ed anche in questo caso servirà la necessaria attenzione in quanto l'inversione di uno o più connettori sicuramente provocherà malfunzionamenti e, in alcuni casi, si rischierà di provocare guasti importanti.

Per tale ragione abbiamo approntato uno schema pratico per i collegamenti tra PCB e periferiche, illustrato nello schema di cablaggio.

In particolare bisogna fare molta attenzione ai collegamenti tra il PLUG DC da pannello ed il cavetto a due poli con terminazione JST, un'inversione tra i due fili significherebbe bruciare sicuramente qualcosa; prima di innestare lo spinotto JST nella presa J3, consigliamo di collegare un alimentatore con le caratteristiche richieste al PLUG DC e misurare con un multimetro la tensione sui due fili: collegando i puntali rosso e nero sui corrispondenti fili dovrete rilevare una tensione positiva, in caso contrario significa che avete invertito i due poli sui pin del PLUG. I due LED andranno saldati ai tre fili del relativo cavetto MK, con questo criterio: l'anodo (pin più lungo) del LED rosso andrà saldato al filo collegato al pin 1 dell'MK, l'anodo del LED blu andrà saldato al filo collegato al pin 3 dell'MK, infine i due catodi (pin più corti) dovranno essere saldati tra loro e successivamente al filo collegato al pin 2 dell'MK. I 5 fili del cavetto MK collegato a J, andranno poi saldati sul modulo amplificatore. Attenzione infine al verso di inserimento della tastiera a membrana, fate riferimento alla **Fig. 3** oltre che allo schema di cablaggio.

L'ultima nota riguarda il keypad 7+1, per usarlo nella configurazione prevista per questo progetto, il cappuccio del jumper JP1 andrà inserito sulla posizione RST. Consigliamo di non montare subito tutto nel contenitore, meglio prima sincerarsi che sia tutto a posto. Naturalmente a questo punto bisognerà programmare opportunamente l'Arduino

69

```
// myDFPlayer.EQ(DFPLAYER_EQ_ROCK); //2
myDFPlayer.EQ(DFPLAYER_EQ_JAZZ); //3
// myDFPlayer.EQ(DFPLAYER_EQ_CLASSIC); //4
// myDFPlayer.EQ(DFPLAYER_EQ_BASS); //0
```

Nello sketch troverete attiva la versione "jazz", per cambiarla sarà sufficiente commentare la relativa riga e decommentare quella desiderata. Seguono tre istruzioni di impostazione e tre di lettura del modulo MP3:

```
myDFPlayer.enableDAC(); //abilita il DAC integrato
myDFPlayer.outputSetting(true, 15); //abilita l'uscita e
setta il guadagno a 15
myDFPlayer.volume(20); //Imposta il Volume iniziale
(tra 0 e 30)
lcd.print(myDFPlayer.readFileCounts()); //legge il
numero di brani presenti nell'SD
lcd.print(myDFPlayer.readEQ()); //legge l'impostazione
dell'equalizzatore interno
lcd.print(myDFPlayer.readVolume()); //legge il valore
del volume
```

Le ultime due servono in realtà per verifica delle impostazioni precedenti. Queste informazioni vengono poi visualizzate sul display per qualche secondo, quindi il Juke-box va in "Stop". In **Fig. 6** vediamo la sequenza dei messaggi iniziali visualizzati sul display.

Sezione Loop

Questa sezione, come ben sappiamo, viene eseguita continuamente e ciclicamente all'infinito e, ogni volta che rileva uno di questi eventi: lettura di un TAG RFID, inserimento di un numero con la tastiera a membrana, pressione di un pulsante del keypad 7+1, estrazione o inserimento della micro SD-card, fine dell'esecuzione di un brano, richiesta di reset dell'EEPROM da parte dell'Amministratore, vengono eseguite delle specifiche istruzioni.

SCHEDE

Diamo ora uno sguardo alle schede disponibili.

ADC_to_Number: contiene la routine di conversio-



Fig. 6
Messaggi iniziali
sul display.

ne del valore analogico generato dal keypad 7+1 in valore digitale tra 0 e 1023.

Amministratore: contiene le routine di creazione della Card Amministratore, di uscita dalla fase di amministrazione e di azzeramento dei dati contenuti nella EEPROM. I relativi messaggi sono illustrati nella **Fig. 7**.

Debug: contiene la routine che intercetta i messaggi inviati dal modulo MP3 tramite la Software Serial, cioè la condizione di rimozione o inserimento della CARD e la gestione della fine dell'esecuzione del brano (va in "Stop" nella modalità Utente Singolo, mentre legge il numero del brano successivo nelle modalità Utente Loop o Utente Random). I messaggi generati in questa scheda sono visualizzati nella **Fig. 8**.

Messaggi: contiene la routine "Play", attiva quando si esegue un brano, che indica la tipologia di Utente e la modalità Amministratore, mediante alcune lettere, la routine "Stop" che parte all'accensione e nel momento in cui il lettore si ferma dopo l'esecuzione di un brano in modalità singola; la routine "number" serve per leggere dal lettore il numero del brano in esecuzione, la "tag_request" gestisce la richiesta dei TAG RFID.

In **Fig. 9** abbiamo riportato tutti i messaggi collegati a questa scheda.

Questi messaggi sono strettamente collegati a quelli generati nella sezione Loop dello sketch, quando si usano i pulsanti "Precedente", "Successivo", "Play/Pausa", "Volume" e "Mode", e che abbiamo raggruppato nella **Fig. 10**, miscelando le varie modalità Utenti ed Amministratore.

Modalità: contiene le routine singleMode (Utente Singolo eventuale richiesta di selezione del numero del brano), loopMode (Utente Loop), randomMode (Utente Random) e OFFMode (spegnimento del lettore MP3 da parte dell'Amministratore). In **Fig. 11** sono visibili i messaggi di questa sezione.

rc522: in questa scheda è localizzata tutta la gestione dei TAG RFID, quindi rilevazione e memorizzazione iniziale dei TAG, riconoscimento del tipo di Utente (Amministratore, Singolo, Loop e Random). La **Fig. 12** illustra tutti i messaggi di amministrazione dei TAG RFID.

Tastiera: la routine kybrd si occupa di gestire l'inserimento del numero del brano tramite tastiera a membrana.

Volume: la routine getVolume legge l'impostazione di volume del modulo per la successiva visualizzazione sul display.

Per chi volesse approfondire l'utilizzo del lettore MP3, riepiloghiamo qui i comandi usati per le varie

modalità di esecuzione:

myDFPlayer.playMp3Folder(brano); //brano è il numero del file che sarà eseguito in modalità Utente Singolo

myDFPlayer.pause(); //va in PAUSA alla prima pressione del pulsante "Play/Pausa"

myDFPlayer.start(); //dopo la PAUSA, riprendendo "Play/Pause" riparte dal punto di arresto

myDFPlayer.enableLoopAll(); //esegue in loop tutti i brani, in modalità Utente Loop

myDFPlayer.randomAll(); //esegue in random tutti i brani, in modalità Utente Random

myDFPlayer.disableLoopAll(); //interrompe il loop
Non esiste un comando per l'interruzione dell'esecuzione random, l'unico modo è quello di attivare il loop e poi disattivarlo.

PREPARAZIONE DELLA MICRO SD

Per il funzionamento del Juke-box occorre predisporre una microSD-card con capacità massima di 32Gb, formattata FAT16 o FAT32, inserendovi fino a 255 file mp3. È importante seguire questi passaggi in sequenza, altrimenti potreste avere qualche difficoltà con l'esecuzione dei brani:

- Creare una cartella "mp3" sulla micro SD card;
- Selezionare sul PC i brani mp3 e rinominarli anteposendo al nome la codifica xxxx, iniziando da 0001, senza saltare numeri, per esempio: 0001_ Il cielo in una stanza.mp3, 0002_ Piazza Grande.mp3, ecc. La codifica a 4 cifre è giustificata dal fatto che in alcuni casi il numero di file potrebbe arrivare a 3000;
- Caricare i file sulla micro SD card, uno per volta, seguendo l'ordine numerico; un errore in questa fase significa perdere la corrispondenza tra la numerazione reale e quella visualizzata sul display, durante l'esecuzione del brano. Questa cosa è specificata nella wiki: in pratica l'esecuzione del file mediante chiamata diretta rispetta la



← Fig. 7
Messaggi
Amministratore.



← Fig. 8
Messaggi di
stato della SD
card.

numerazione, mentre l'esecuzione in sequenza dei file segue l'ordine di caricamento;

- Inserire la micro SD card nello slot del modulo DFR0299;
- In qualsiasi momento sarà possibile, successivamente, aggiungere altri brani alla cartella, ma sempre seguendo queste due semplici regole; quindi l'eventuale sostituzione di uno o più file richiederà la cancellazione di tutti i brani e il caricamento dell'intero nuovo elenco.

PREPARAZIONE DI ARDUINO NANO

Una volta montato il PCB del Juke-box e collegate



← Fig. 9
Messaggi
generici di
esecuzione e
stop.

→ **Fig. 10**
Messaggi della
sezione Loop dello
sketch.



→ **Fig. 11**
Messaggi di
attivazione
modalità.



tutte le periferiche esterne, naturalmente bisogna programmare il microcontrollore di Arduino Nano per poter iniziare ad usarlo. La prima operazione da fare è quella di aprire l'IDE, possibilmente la versione più recente (nel nostro caso è la 1.8.15). Bisogna poi scaricare ed aggiungere le varie librerie previste nello sketch: DFRobotDFPlayerMini-1.0.3.zip (modulo MP3), LiquidCrystal_I2C-1.1.2.zip (LCD I²C), vma405_library.zip (RFID), Keypad-master.zip (tastiera a membrana); queste librerie sono scaricabili dai rispettivi link, che vi abbiamo fornito nel numero precedente o all'inizio di questo articolo,

oppure le troverete nella sezione download del sito www.elettronicaain.it. Il comando da eseguire nell'IDE è Sketch - #include libreria - Aggiungi libreria da file .ZIP, selezionando poi il percorso nel quale sono state inserite le quattro librerie (**Fig. 13**).

Le librerie non menzionate sono già presenti di serie, quindi non vanno aggiunte. Una volta completata questa operazione bisogna scaricare e decomprimere la cartella "MP3_Jukebox_LCD_I2C_7T_RFID_Keyp.rar", posizionandola quindi nella cartella degli sketch dell'IDE. Quindi occorre aprire il file principale "MP3_Jukebox_LCD_I2C_7T_RFID_Keyp.ino" e fare una compilazione di verifica, se dovessero uscire errori, probabilmente sarà dovuto alla mancanza di qualche libreria, se invece l'operazione va a buon fine, allora si potrà procedere col caricamento dello sketch nella board.

USO DEL JUKE-BOX

A fine caricamento, dopo alcuni istanti, sul display appariranno in sequenza le due immagini viste in **Fig. 6**, quindi il Juke-box andrà in "Stop", si accenderà il LED rosso e apparirà la richiesta di TAG (**Fig. 9**). La prima operazione da fare sarà la memorizzazione dei TAG RFID, facendo attenzione

→ **Fig. 12**
Messaggi TAG
RFID.



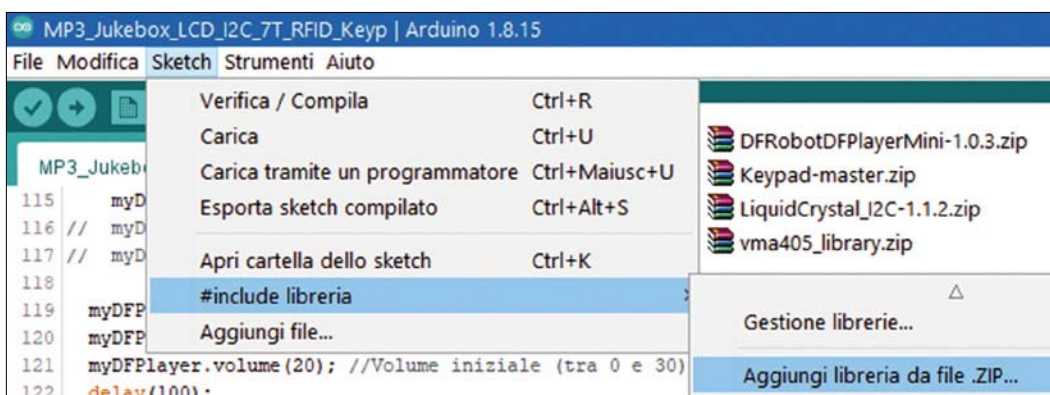


Fig. 13
Inclusione
delle librerie.

a questi criteri, imposti dallo sketch:

- Il primo TAG che si avvicina al Modulo RFID viene automaticamente memorizzato come Amministratore; dopo questa operazione il sistema esce dall'amministrazione (il LED rosso si spegne e poi si riaccende); per questa funzione abbiamo usato una CARD RFID, per quelle successive tre TAG portachiavi/gettone (**Fig. 14**).
- La seconda operazione da fare è avvicinare il TAG Amministratore per entrare nella corrispondente modalità (i due LED lampeggiano cinque volte mentre sul display, in alto a sinistra, appare la lettera "A"); a questo punto si può avvicinare un nuovo TAG, che sarà memorizzato come Utente Singolo; il sistema chiuderà automaticamente la modalità amministrazione;
- La terza operazione è identica alla precedente, solo che il nuovo TAG sarà memorizzato come Utente Loop;
- La quarta ed ultima operazione, sempre identica alle precedenti, servirà per memorizzare l'Utente Random.

Ogni volta che si entra in modalità Amministratore un messaggio indica quanti altri TAG sono memorizzati (da 0 a 3). I messaggi relativi a queste operazioni sono quelli visti in **Fig. 12**. A questo punto non sarà più possibile aggiungere altri TAG; nel normale funzionamento del Juke-box; avvicinando uno dei 4 TAG memorizzati si attiverà la specifica funzione, tutti quelli non memorizzati saranno rifiutati, senza alcun effetto sul sistema, eccetto il messaggio di mancata autorizzazione (**Fig. 9**). Naturalmente, disponendo di un numero inferiore di TAG, sarà possibile accedere solo alle funzioni nell'ordine in cui le abbiamo elencate; per esempio, disponendo di due TAG, il primo sarà Amministratore ed il secondo Utente Singolo.

In qualsiasi momento sarà possibile sostituire tutti i TAG, compreso quello Amministratore, effettuando

preventivamente l'operazione di TAG RESET; sarà sufficiente entrare in modalità Amministratore e premere tre volte il pulsante TAG_RST, con un intervallo di 2 secondi tra una pressione e la successiva, oppure premere e mantenere premuto lo stesso pulsante per oltre 5 secondi, il messaggio visto in **Fig. 7** indicherà il buon esito dell'operazione. Le tre pressioni consecutive dovranno essere effettuate nel tempo massimo di 20 secondi, altrimenti il contatore sarà azzerato. Il numero di pressioni e le tempistiche sono state impostate per garantire un buon livello di sicurezza contro reset accidentali.

Nota bene: in caso di smarrimento o danneggiamento del TAG Amministratore l'unico modo per azzerare l'EEPROM e poter poi ripetere le operazioni di memorizzazione, sarà quello di caricare su Arduino Nano lo sketch "EEPROM_canc" che troverete sempre nella sezione download del sito www.elettronicaain.it; la cancellazione sarà eseguita istantaneamente e sarà confermata dall'accensione del LED "L" presente sulla board Arduino Nano. Subito dopo occorrerà ricaricare lo sketch del Juke-box ed a quel punto, il primo TAG che si avvicinerà al



Fig. 14
I TAG usati
per le prove
del Juke-box.



Lo stupendo mobile realizzato per il prototipo.

lettore RFID diventerà il nuovo Amministratore. terminate le operazioni di memorizzazione dei TAG, saremo finalmente pronti ad usare il nostro Juke-box! Vediamo cosa succede nelle varie modalità.

Utente Singolo: vi si accede con il primo TAG memorizzato dall'Amministratore, prima di avvicinarlo occorre scegliere il brano da ascoltare digitando il relativo numero sulla tastiera a membrana; se il numero è di 4 cifre la conferma è automatica, se le cifre sono inferiori bisogna premere il tasto "#"; non è necessario usare gli zero iniziali, per esempio il brano 0014 si richiama mediante la digitazione 14#; il numero prescelto sarà istantaneamente mostrato sulla seconda riga del display; una volta riconosciuto il TAG l'esecuzione del brano partirà automaticamente (messaggi visibili in **Fig. 9**), con lo spegnimento del LED rosso e l'accensione di quello blu; inoltre il display mostrerà la scritta "S-PLAY" seguita dal numero in esecuzione. Se il numero inserito è 0 oppure maggiore del numero di file memorizzati nella micro SD card, viene ignorato. Al termine dell'esecuzione il Juke-box tornerà in "Stop" e resterà in attesa di un TAG. Per ascoltare un nuovo brano occorre inserirne il relativo numero e poi avvicinare nuovamente il TAG; l'inserimento del nuovo brano potrà essere effettuato anche durante l'esecuzione del precedente, che potrà essere interrotta avvicinando nuovamente il TAG, a quel punto sarà subito eseguito il nuovo brano. In questa modalità gli unici pulsanti attivi del keypad 7+1 sono "Volume +/-" e "Play/Pausa". Se durante

l'esecuzione di un brano si avvicina uno degli altri due TAG Utente, il Juke-box interrompe l'esecuzione del brano attuale, cambia modalità e ricomincia dal n.1.

Utente Loop: vi si accede con il secondo TAG memorizzato dall'Amministratore; in questo caso l'inserimento di un numero non serve a niente, visto che in questa modalità vengono eseguiti tutti i brani presenti nella micro SD card, partendo sempre dal numero 1, in ordine numerico crescente; l'esecuzione avverrà fino a quando non sarà bloccata dall'Amministratore oppure fino alla pressione del pulsante RESET, che farà ripartire lo sketch tornando in "Stop". In questa modalità, oltre ai pulsanti "Volume +/-" e "Play/Pausa", vengono attivati anche i due che permettono di passare in qualsiasi momento al brano precedente o successivo, rispetto a quello in esecuzione. L'avvio di questa modalità sarà indicato dallo spegnimento del LED rosso e accensione di quello blu; inoltre il display mostrerà la scritta "L-PLAY" seguita dal numero in esecuzione. Se durante l'esecuzione di un brano si avvicina nuovamente il TAG Loop, viene ignorato; se invece si avvicina uno degli altri due TAG Utente, il Juke-box interrompe l'esecuzione del brano attuale, cambia modalità e, nel caso dell'Utente Singolo, esegue il brano eventualmente preselezionato con la tastiera oppure chiede l'inserimento del numero ed un nuovo riconoscimento del TAG; invece nel caso dell'Utente Random ricomincia dal n.1.

Utente Random: vi si accede con il terzo TAG

memorizzato dall'Amministratore; vale tutto quanto descritto per l'Utente Loop, ad eccezione del fatto che i brani saranno eseguiti secondo un ordine casuale (random, appunto), talmente tanto casuale che potrà capitare che lo stesso brano venga ripetuto più volte prima che venga completata l'intera lista. Anche in questo caso l'esecuzione potrà essere interrotta solo dall'Amministratore o dalla pressione del pulsante RESET, tornando in "Stop". L'avvio di questa modalità sarà indicato dallo spegnimento del LED rosso e accensione di quello blu; inoltre il display mostrerà la scritta "R-PLAY" seguita dal numero in esecuzione. Se durante l'esecuzione di un brano si avvicina nuovamente il TAG Random, viene ignorato; se invece si avvicina uno degli altri due TAG Utente, il Juke-box interrompe l'esecuzione del brano attuale, cambia modalità e, nel caso dell'Utente Singolo, esegue il brano eventualmente preselezionato con la tastiera oppure chiede l'inserimento del numero ed un nuovo riconoscimento del TAG; invece nel caso dell'Utente Loop ricomincia dal n.1.

I messaggi relativi a queste tre modalità sono visibili in **Fig. 9**.

Amministratore: abbiamo già detto che questa modalità ha il pieno controllo del Juke-box e dei TAG Utenti. Come abbiamo visto, oltre a memorizzare TAG può anche cancellarli. Se si avvicina il TAG quando il Juke-box è in "Stop", si entra nell'amministrazione, con l'attivazione di tutti i pulsanti del Keypad 7+1; i due LED lampeggiano cinque volte e viene visualizzato il riconoscimento del TAG Amministratore. Se si seleziona il numero del brano e si preme il pulsante "Play/Pausa" il Juke-box va in modalità Utente Singolo, però con tutti i permessi dell'Amministratore, quindi funzionano tutti i pulsanti; se invece si preme il pulsante senza aver prima selezionato un brano, la modalità è sempre la stessa ma viene eseguito il n.1. Il pulsante "Mode" consente di passare istantaneamente alle modalità Utente Loop (prima pressione) e Utente Random (seconda pressione); una terza pressione del pulsante "Mode" spegne il lettore MP3 ed il Juke-box va in "Stop". Oltre che con i pulsanti "Play-Pausa" e "Mode", la modalità di esecuzione durante la fase di amministrazione può essere impostata anche semplicemente avvicinando il relativo TAG Utente. Se invece si entra in amministrazione mentre è attiva una delle tre modalità Utente, l'esecuzione del brano attuale continua senza alcuna interruzione, si resta nella modalità attuale, ma con i privilegi di Amministratore, riguardo l'uso dei pulsanti. A prescindere dalla condizione operativa, avvicinando nuovamente il TAG Amministratore, si esce dall'amministrazione, il lettore MP3 viene spento ed il Juke-box va in "Stop", con l'eventuale spegnimento del LED blu (se si era in fase di esecuzione) e l'accensione del LED rosso.

In base al tipo di alimentatore di cui sarà dotato il Juke-box, per evitare problemi a volume troppo elevato, consigliamo di fissare all'origine il limite massimo con una semplice operazione: si sceglie un brano che abbia una buona intensità sonora e lo si esegue in modalità singola, si ruota il potenziometro del volume al massimo e quindi, a tentativi, si agisce sulla riga "*myDFPlayer.volume(xx);*" dello sketch, fino a trovare per "xx" un valore che consenta di ottenere il massimo volume possibile senza distorsioni.

IL CONTENITORE

Una volta completati tutti i test, il PCB principale e tutte le altre periferiche potranno essere montati nel contenitore che abbiamo disegnato appositamente per questo progetto e che vedete nelle foto in queste pagine. Per chi dispone di una stampante 3D, nella sezione download del sito www.elettro-nicain.it abbiamo reso disponibile il file STL per la realizzazione del contenitore. Naturalmente sarà possibile qualsiasi altra soluzione, ma sempre tenendo presente che occorrerà posizionare il modulo RFID-RC522 in modo tale da favorire l'avvicinamento dei TAG, quasi a contatto con esso, altrimenti si avranno difficoltà nel loro riconoscimento. Non ci resta che augurarvi buon divertimento con questo strumento che vi permetterà di ricreare l'atmosfera dei mitici anni '60-'70-'80! ■



Cosa occorre?

I componenti utilizzati in questo progetto sono disponibili presso Futura Elettronica. La scheda Arduino Nano (cod. ARDUINONANO) è disponibile a Euro 25,00, il modulo lettore MP3 (cod. DFR0299) è in vendita a Euro 12,00, il lettore RFID con MFRC522 (cod. VMA405N) costa Euro 7,80, l'amplificatore audio (cod. PAM8403POT) è disponibile a Euro 6,50, il display LCD con connessione I2C (cod. LCD16X2A12C) è in vendita a Euro 7,00. Per realizzare il PCB è possibile utilizzare il servizio PCBPRODUCTION disponibile sul sito www.futurashop.it

Il materiale va richiesto a:

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



Purificatore d'aria a raggi ultravioletti

VERSIONE
DA PARETE

€179,00

Cod. MICROUVAIR-P

Alimentazione: 12 Vcc

Potenza assorbita: @ 3.6 W

Durata sorgente UV: @20000 ore

Capacità sterilizzazione: 60 m³/h

Dimensioni: altezza 25 cm x 10.5 cm x 7 cm

È un sistema di sterilizzazione dell'aria che assicura l'eliminazione di virus e batteri.

Riduce il rischio di contaminazione delle persone all'interno degli uffici.

Rende gli ambienti batteriologicamente sani eliminando anche le spore.

È in grado di effettuare un trattamento di disinfezione dell'aria in ambienti fino a 60 m³.

Può funzionare continuativamente anche in presenza di persone.

Abbatte il 99% della carica batterica aerea dopo circa un'ora di funzionamento.



MICROUVAIR è un sistema di sanificazione aria con ventilazione forzata a ciclo chiuso basato su led UVC. L'aria in ingresso viene purificata attraverso un filtro ad alta densità (30 PPI), quindi viene sterilizzata tramite luce ultravioletta emessa da led UVC in grado di effettuare la massima azione germicida. L'aria sanificata viene quindi filtrata ulteriormente da un filtro meccanico ad alta densità e quindi rimessa nell'ambiente.



VERSIONE
DA TAVOLO

€199,00

Cod. MICROUVAIR-T



Prezzi IVA inclusa.

SANIFICATORE UVC PER OGGETTI

Sanificatore automatico con LED UVC, particolarmente indicato in tutte le situazioni dove si rende necessario igienizzare, sanificare e comunque disinfettare oggetti potenzialmente depositari di particelle infettive (virus, batteri, ecc.) come mascherine chirurgiche, utensili, occhiali, chiavi, ecc. È costituito da un contenitore in plexiglass con un coperchio incernierato all'interno del quale si trova una strip a LED UV-C con quattro LED SMD UV-C tipo 3535 da 5 mW ciascuno con lunghezza d'onda della radiazione emessa nella banda dei 265 e i 278 nm, gamma ideale per sterilizzare e sanificare superfici e oggetti. Il set contiene tutte le parti in plexiglass, le parti stampate in 3D, l'elettronica e l'alimentatore.

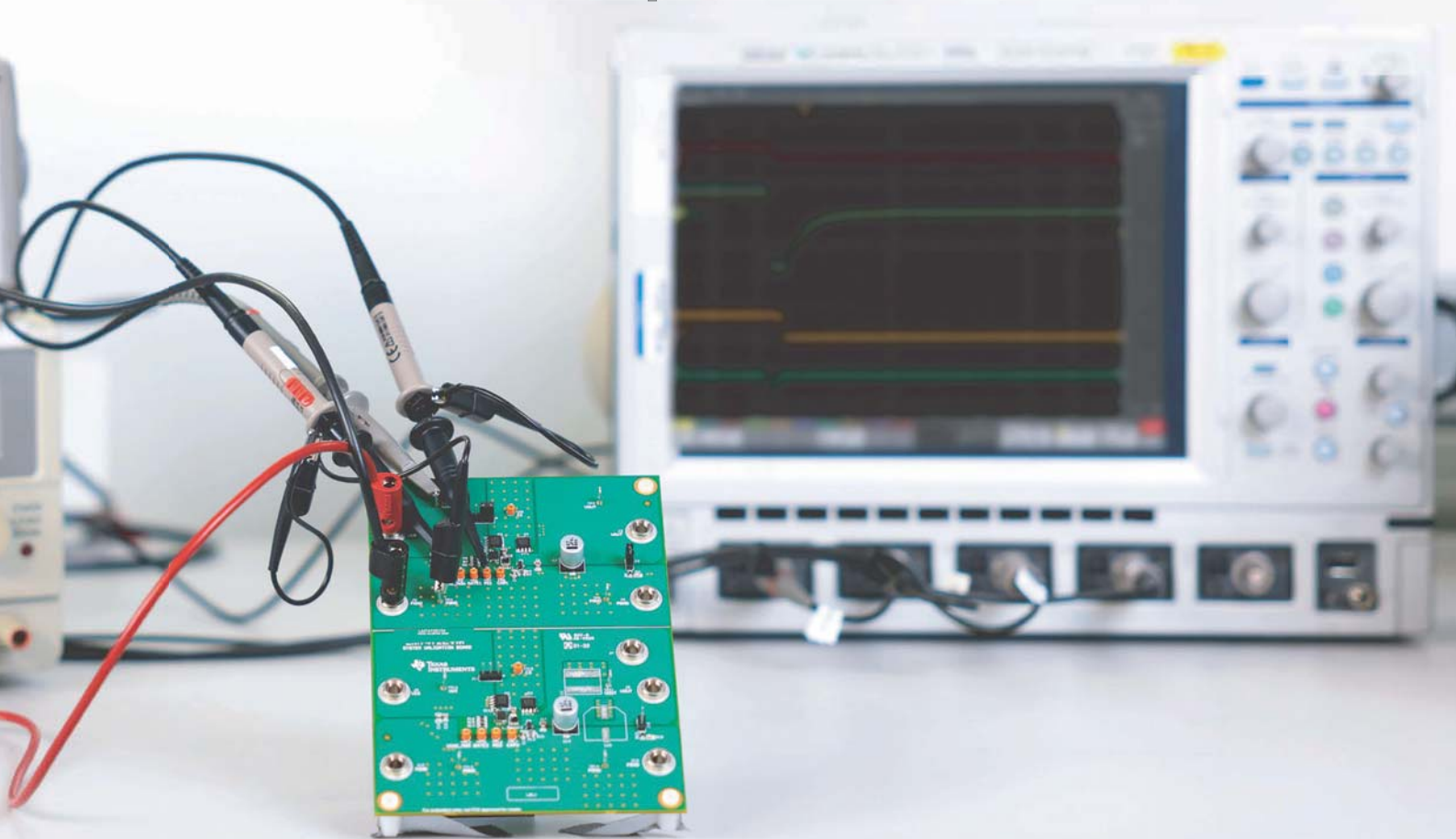


€80,00

Cod. UVCSAN



Riduce la corrente di riposo. Migliora le prestazioni. Nessun compromesso.



Nei sistemi a batteria, è essenziale ottenere un'elevata efficienza in condizioni di carico leggero o vuoto. Con la tecnologia e i prodotti di TI a bassissima IQ, puoi estendere la durata della batteria e sviluppare progetti futuri a basso consumo.

**Il partner che ti risolve i problemi sulla
corrente di riposo.**

► Scopri di più > [TI.com/lowiq](https://ti.com/lowiq)

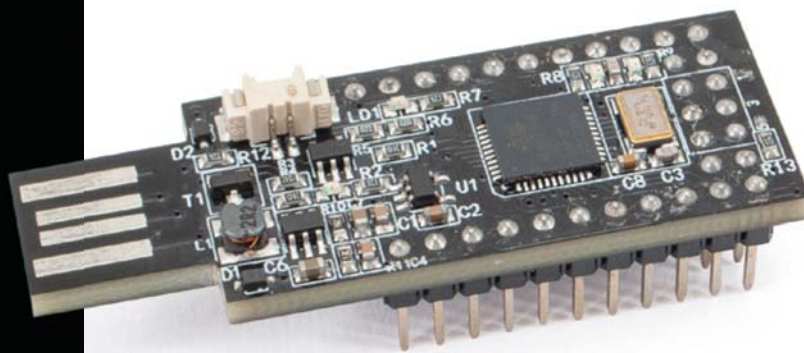


COMPUTER VISION **NEOPIXEL MAPPING**



di LUCA BELLAN

Rileviamo, tramite un algoritmo di computer vision, la posizione dei NeoPixel posti in ordine sparso e trasformiamoli in una matrice di LED virtuale cui applicheremo effetti luminosi.



Il Natale è ormai alle porte e, come da tradizione, addobbi luminosi di ogni tipo compaiono nei giardini e all'interno delle case; nel corso degli anni la tecnologia alla base delle luminarie è cambiata molto: dalle lampadine ad incandescenza ai LED, fino ad arrivare ai proiettori per le pareti esterne o alle luci controllate da

smartphone.

Il fattore che accomuna le luminarie, specialmente le catene di luci, è che sono quasi sempre posizionate in ordine sparso in modo da ricoprire l'oggetto da addobbare, e di conseguenza gli effetti luminosi vengono eseguiti in modo casuale. Immaginiamo di applicare una catena di LED indirizzabili (anche conosciuti come LED NeoPixel) su un albero di Natale: grazie alla mappatura in computer vision potremo trasformare la catena, posizionata in ordine casuale, in una sorta di schermo LED, rilevando la posizione di ogni singolo NeoPixel e disegnando successivamente la matrice di punti



su un frame video che sarà mostrato a schermo. In questo articolo vi guideremo nella creazione di un programma per PC in Python che, grazie al controllo via seriale dei LED e l'analisi con computer vision delle immagini provenienti dalla webcam, individuerà la posizione di ciascun LED delle luminarie, ottenendo così una matrice virtuale che successivamente si potrà indirizzare a piacimento.

Il secondo programma in Python leggerà la matrice di LED precedentemente salvata ed eseguirà degli algoritmi per creare effetti luminosi visualizzati su un frame video: operando ad una frequenza sufficientemente elevata, verrà analizzato il colore nella posizione di ogni singolo LED e, sempre comandando il microcontrollore via USB, invieremo i colori aggiornati a tutti i LED connessi. Questo sistema potrà essere utilizzato, inoltre, per mappare diversi oggetti NeoPixel a patto che siano tutti connessi in sequenza; ad esempio sopra l'albero di Natale potremo posizionare la stella NeoPixel collegandola alla fine della catena di LED, oppure avremo la facoltà di disporre diverse stelle NeoPixel su una parete, anche in ordine sparso, trasformandole in un unico display.

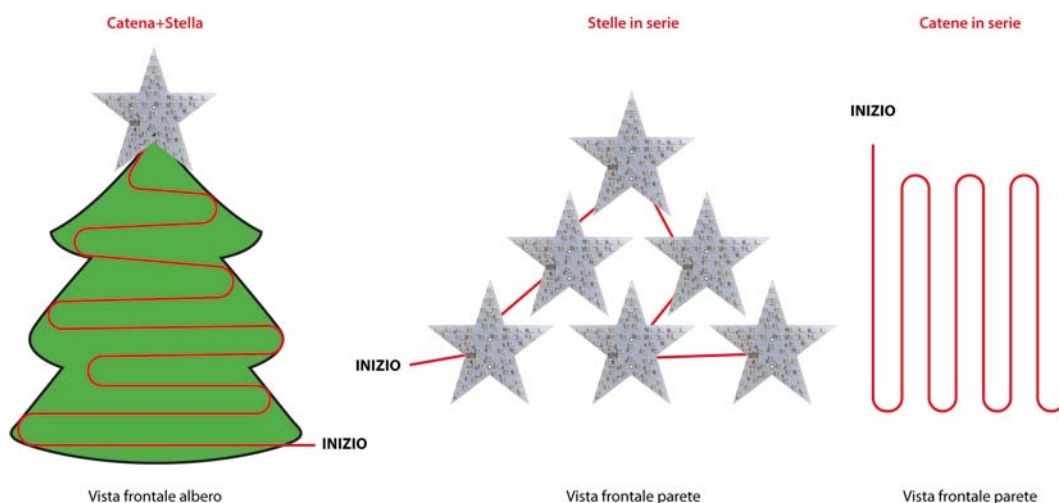
L'HARDWARE

Per realizzare questo progetto utilizzeremo principalmente un PC Windows, che eseguirà i nostri programmi Python per comandare i LED, una webcam USB, che servirà nella fase di mappatura e un alimentatore a 5 VDC con potenza adeguata al numero dei LED che andremo a controllare. Consideriamo che un singolo NeoPixel può arrivare ad assorbire fino a 60 mA quando è acceso alla sua massima luminosità a luce bianca,

ma, generalmente, la luminosità degli effetti che andremo a creare si mantiene al di sotto del 50%, quindi possiamo stimare, con un margine di sicurezza, 30 mA per NeoPixel; moltiplichiamo il numero dei LED (supponiamo di averne 200) per l'assorbimento diviso 1000 (nel nostro esempio: $200 \times 30 : 1000$) e otteniamo 6A di assorbimento totale (30 W).

Il microcontrollore che abbiamo selezionato per questo progetto è Miniduino: la scheda, dalle dimensioni estremamente contenute (19 x 40 x 3,5 mm), si connette direttamente alla porta USB del computer facilitandone la programmazione e il successivo controllo dei NeoPixel. Miniduino è basato sul microcontrollore ATmega32U4 che, grazie allo stesso bootloader di Arduino Leonardo, viene riconosciuto come tale da Arduino IDE una volta inserito nel PC; questa scheda incorpora anche la periferica USB, quindi non occorre utilizzare un convertitore USB/TTL per caricare lo sketch o comunicare in seriale. A bordo troviamo i LED che mostrano l'attività della seriale (LD1 e LD3) e il classico LED per applicazioni generiche (LD2); inoltre è disponibile un connettore (BAT) per batterie al litio da 3,7 VDC: la batteria viene caricata quando la scheda è alimentata tramite la USB del PC e la corrente di carica erogata è pari a 100 mA. La board accetta una tensione di alimentazione tra i 4,5 e i 5,5 V e assorbe circa 15 mA, dispone inoltre di 20 pin I/O digitali, 7 canali PWM, 12 input analogici, 1 UART, 1 BUS I²C e 1 SPI. La memoria flash ha le dimensioni di 32 kB mentre la SRAM è pari a 2,5 kB: queste caratteristiche, unite al formato da pen-drive, lo rendono estremamente utile nel caso in cui avessimo a disposizione poco spazio e non volessimo utiliz-

Fig. 1
Esempi di
installazione dei
LED.



La board Miniduino è basata sul microcontroller Atmel ATmega32U4, che è lo stesso utilizzato sulle schede Arduino Leonardo, Micro ed altre; non a caso una volta inserita nella presa USB di un Personal Computer viene riconosciuta come Arduino Leonardo.

Abbiamo scelto questo microcontrollore in quanto incorpora la periferica USB, quindi è possibile caricare lo sketch senza dover utilizzare un convertitore USB/TTL. Inoltre l'ATmega32U4 può anche simulare una tastiera o un mouse, funzione utile in molte occasioni.

Le dimensioni della nostra Miniduino sono molto contenute (19x40x3,5 mm esclusi pin-strip e connettore USB). Per ottenere tale risultato abbiamo ottimizzato la disposizione di alcuni pin, mettendoli accanto al connettore ICSP.

Questo è previsto in posizione centrale posteriore come nella Nano e accanto ad esso abbiamo spostato i pin GND, CTS, TX ed RX della seriale UART, così da accorciare il più possibile il circuito stampato e rendere minore il carico sul connettore USB se la scheda viene introdotta in una presa di un Personal Computer. Il PCB della scheda ha uno spessore da 2,4 mm che permette l'inserimento nella porta USB mantenendo stabile il contatto. I pin possono avere più funzioni, per semplificarne il riconoscimento nella figura a fianco trovate le possibili alternative di uso. La figura mostra la pin-out guardando la scheda da sotto (lato senza componenti).

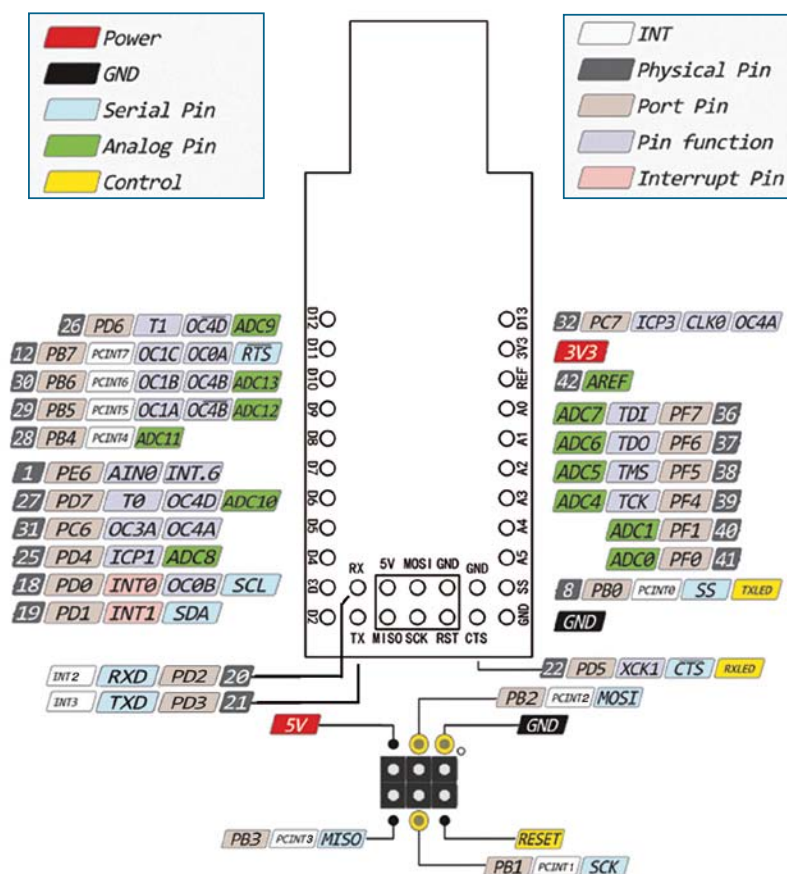
Per utilizzare la scheda è sufficiente inserirla in una presa USB del vostro computer e verificando che si accendano i due LED più vicini al fondo.

La scheda può essere alimentata da una batteria al litio a singola cella, da collegare

con il cavetto in dotazione (tali batterie vengono vendute con cavetto rosso-nero e connettore femmina volante a due poli a passo 2,54 mm) al connettore maschio previsto sul PCB; la batteria si ricaricherà nei periodi di connessione all'USB del computer (o di un alimentatore con uscita USB) e fornirà alimentazione alla nostra Miniduino una volta che verrà sconnessa.

Concludiamo precisando che se lo desi-

derate, potete inserire al board in un contenitore in materiale plastico (se ne può usare uno di quelli standard) lavorato per far uscire il connettore USB su PCB dalla parte anteriore e gli eventuali pin-strip dal coperchio, oltre che per rendere visibili i LED. Chi ha una stampante 3D può anche realizzare il contenitore utilizzando del filamento di PLA o PET trasparente, dandogli la sagoma che preferisce: ad esempio quella di una Pen Drive USB.



La pin-out della nostra Arduino USB vista da sotto.

zare un cavo USB per la programmazione e la comunicazione seriale. I tipi di LED che utilizzeremo dovranno essere tutti indirizzabili: strisce, catene o stelle luminose.

IL PROGETTO IN PRATICA

Procediamo ora con l'allestimento dell'hardware del progetto: per prima cosa occorre individua-

re l'oggetto da ricoprire con i LED e, quando effettueremo l'operazione di posizionamento, dovremo tenere presente che tutti i LED dovranno essere visibili quando si osserva l'oggetto frontalmente: questo è estremamente importante in quanto la webcam dovrà riuscire a vedere tutti LED durante la fase di mappatura. Possiamo aiutarci guardando alcuni esempi in **Fig. 1**.

Per ottenere una buona resa, consigliamo di non distanziare troppo i LED, ma di mantenerli a pochi centimetri tra loro: più l'area allestita sarà densa di LED, migliore sarà la resa degli effetti luminosi. Per allestire un albero di Natale utilizziamo una o più catene, partendo dal basso e procedendo a zig-zag in orizzontale, nella parte frontale dell'albero, fino a raggiungere la cima, rispettando il senso di controllo dei NeoPixel quando andiamo a giuntare; arrivati in alto colleghiamo la fine della catena con la stella NeoPixel da posizionare in cima all'albero (**Fig. 1**). Avendo invece a disposizione una parete, una siepe o un parapetto, posizioniamo i LED (catena o strisce) a tenda: partiamo da destra o da sinistra e procediamo verso l'altro lato a zig-zag in verticale, facendo sempre attenzione ad effettuare le giunte nel senso corretto; in alternativa potremo posizionare sulla parete diverse stelle NeoPixel, ravvicinate ma sfalsate tra di loro.

Terminata la fase di allestimento, procediamo ora con i cablaggi facendo riferimento alla **Fig. 2**: colleghiamo l'alimentatore da 5 V ai due capi della catena, al fine di evitare una diminuzione del voltaggio dato dalla resistenza dei LED e dei cavi;

se durante l'illuminazione notiamo che i LED centrali risultano meno luminosi, dovremo portare l'alimentazione anche nella sezione centrale della catena.

Sui morsetti dell'alimentatore consigliamo di inserire un condensatore (500–1000 μ F da 6.3 V o più) che si occuperà di fornire corrente durante i picchi di assorbimento che potrebbero avvenire a causa di repentini cambi di luminosità dati dagli effetti luminosi. Partendo dall'inizio della catena, mediante un cavo bipolare, portiamo al Miniduino (e quindi al PC) il segnale di controllo dei NeoPixel e il negativo dell'alimentatore; guardando Miniduino dal lato senza componenti, colleghiamo il segnale al primo pin in basso a sinistra (D2) e il negativo al primo pin in basso a destra (GND). È buona norma inserire, sul cavo del segnale, una resistenza tra i 300 e i 500 Ω per proteggere il primo LED della catena da eventuali picchi di voltaggio.

Accendiamo ora l'alimentatore a 5 V, inseriamo Miniduino e webcam in due porte USB del computer e passiamo alla parte software del progetto; con essa implementeremo la mappatura dei nostri LED indirizzabili, attraverso la webcam, e procederemo quindi all'applicazione di effetti luminosi generati da specifici algoritmi che creeremo.

Listato 1 – Sketch per Miniduino che riceve in seriale i colori per impostare i NeoPixel.

```
#include <Adafruit_NeoPixel.h>
// BEGIN SETUP
#define LEDS 56
#define PIN 2
// END SETUP
Adafruit_NeoPixel strip = Adafruit_NeoPixel(LEDS, PIN, NEO_GRB + NEO_KHZ800);
long unsigned int packetSize;
int r, g, b;
char packetBuffer[LEDS*3];
void setup() {
  Serial.begin(250000);
  strip.begin();
  strip.setBrightness(255);
  strip.show();
}
void loop() {
  while(Serial.available()) {
    packetBuffer[packetSize] = Serial.read();
    packetSize++;
    if (packetSize == LEDS * 3) {
      packetSize = 0;
      for (int i=0; i<LEDS * 3; i+=3) {
        r = (int)(byte*)(packetBuffer)[i];
        g = (int)(byte*)(packetBuffer)[i+1];
        b = (int)(byte*)(packetBuffer)[i+2];
        strip.setPixelColor(i/3, strip.Color(r, g, b));
      }
      strip.show();
    }
  }
}
```

IL SOFTWARE

Il software per il nostro sistema può essere scaricato dal repository GitHub che si trova all'indirizzo <https://github.com/open-electronics/OpenCV-NeoPixel-Mapping>; qui facciamo clic sul pulsante verde "Code", poi su "Download ZIP" e infine andiamo ad estrarre l'archivio.

Se invece siamo su Linux, assicurandoci di aver installato git, lanciamo il comando:

```
git clone https://github.com/open-electronics/
OpenCV-NeoPixel-Mapping.git
```

La cartella contiene i seguenti elementi:

- *NeoPy_Serial* è la cartella dello sketch per controllare i LED con il microcontrollore;
- *neopixel_serial.py* è la libreria Python per controllare i NeoPixel tramite seriale;
- *view.py* ci consente di visualizzare in una finestra ciò che inquadra la webcam per meglio allinearla agli oggetti da mappare;
- *map.py* è il programma di mappatura in computer vision dei LED;
- *show.py* esegue gli effetti luminosi sui LED

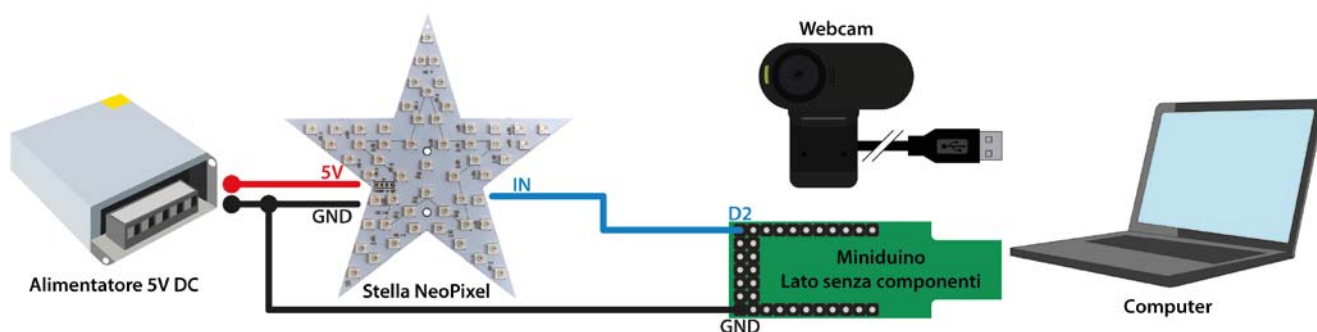


Fig. 2
Schema di
connessione
componenti.

precedentemente mappati;

- *effects* è la cartella che contiene un file Python per ciascun effetto luminoso disponibile: possiamo svilupparne altri a nostro piacimento.

Apriamo, con il programma Arduino IDE, lo sketch *NeoPy_Serial.ino* contenuto nell'omonima cartella, assicuriamoci di aver installato la libreria di gestione NeoPixel, in caso contrario dal menu Sketch, #include libreria, Gestione librerie... cerchiamo e installiamo "Adafruit NeoPixel". Lo sketch si compone di una parte iniziale di configurazione nella quale possiamo settare il numero totale di LED collegati al microcontrollore (come primo esempio utilizzeremo la stella NeoPixel cod. FT1300, quindi 56) e il pin del segnale di controllo (nel nostro caso il pin 2); segue poi l'inizializzazione dell'oggetto NeoPixel e la dichiarazione di alcune variabili che serviranno durante l'esecuzione del programma: la più importante è sicuramente l'array "packetBuffer" che verrà popolato con la sequenza di byte in arrivo sulla seriale dal programma Python (la sequenza è composta dai valori R, G, B di ciascun LED, dal primo all'ultimo).

Nel setup inizializziamo la connessione seriale e la strip NeoPixel settando il valore di luminosità al massimo: questo perché la luminosità effettiva verrà gestita dal programma Python sul PC.

Il loop consta di un ciclo infinito, eseguito alla massima velocità, che attende byte dalla seriale e li inserisce nell'array packetBuffer; quando la lunghezza dei byte ricevuti corrisponde precisamente ai byte che servono per accendere tutti i LED, viene eseguito un ciclo sull'array che, di tre in tre, ottiene i byte necessari a colorare un singolo NeoPixel, ne aggiorna il colore e passa a quello successivo.

Dopo aver settato tutti i LED con il nuovo colore, viene chiamato il metodo "show()" per aggiornare fisicamente i NeoPixel.

Durante la trasmissione seriale di pacchetti (so-

prattutto a velocità estremamente alte) potrebbero giungere al microcontrollore delle informazioni corrotte, come ad esempio una fila di byte non completa o alterata; questo si tradurrebbe in sfarfallii dei NeoPixel o cambi non previsti di colore: per ovviare a questo problema si potrebbe implementare un byte di controllo finale che viene calcolato con lo stesso metodo sia nello script Python che spedisce i byte (neopixel_serial.py) che nello sketch attualmente in uso; in questo modo i pacchetti non corretti verrebbero scartati dallo sketch.

Selezioniamo ora la scheda "Arduino Leonardo" dal menu Strumenti, Scheda (Miniduoino viene riconosciuto come tale), impostiamo la porta corretta e carichiamo lo sketch (lo vedete nel **Listato 1**) su Miniduoino. Passiamo ora all'installazione dei programmi necessari sul nostro PC Windows: per prima cosa navighiamo sul sito di Python e scarichiamo e installiamo l'ultima versione disponibile, che, al momento della scrittura di questo articolo, è la 3.10.0 che funziona su sistemi Windows 8 e 10 <https://www.python.org/downloads/>.

Per Windows 7 è possibile installare la versione precedente scaricandola da questo indirizzo: <https://www.python.org/downloads/release/python-368/>; scorriamo la pagina fino alla sezione "Files" e selezioniamo "Windows x86-64 executable installer".

In tutti i casi, durante l'installazione selezioniamo l'opzione per aggiungere Python al PATH di Windows.

Apriamo il prompt dei comandi di Windows (cmd.exe) e diamo il comando *python --version* seguito da invio; se l'installazione di Python è avvenuta correttamente visualizzeremo il numero di versione appena installata.

Sempre con il prompt dei comandi installiamo le librerie di Python necessarie per il progetto (seriale e OpenCV) dando questi due comandi in sequenza:

▼ Listato 2 - Script di mappatura LED con computer vision.

```
import cv2
import numpy as np
import neopixel_serial
import time
import json
LED_NUMBER = 56
LED_PORT = "COM3"
LED_SPEED = "250000"
LED_BRIGHTNESS = 10
CAMERA_ID = 0
DELAY_LED = 0.1
DELAY_PROCESS = 0.2
THRESH_MARGIN = 0.6
SHOW_VIDEO = True
Save = {}
Save["OffsetX"] = 0
Save["OffsetY"] = 0

strip = neopixel_serial.NeoSerial(LED_NUMBER, LED_PORT, LED_SPEED)
strip.SetBrightness(LED_BRIGHTNESS)
time.sleep(1)
cap = cv2.VideoCapture(CAMERA_ID, cv2.CAP_DSHOW)
ret, StartFrame = cap.read()
StartFrame = cv2.cvtColor(StartFrame, cv2.COLOR_BGR2GRAY)
StartCircleFrame = StartFrame.copy()
time.sleep(1)
TimerLED = time.time()
TimerProcess = time.time()
LEDSet = False
MapStop = False
Counter = 0
leds = []

while (True):
    ret, frame = cap.read()
    if(time.time() - TimerLED >= DELAY_LED and not LEDSet and not MapStop):
        print("")
        print("Switch on LED " + str(Counter))
        # Switch on LED
        strip.Set(Counter, (255, 255, 255))
        strip.Show()
        LEDSet = True
        TimerProcess = time.time()
    if(time.time() - TimerProcess >= DELAY_PROCESS and LEDSet and not MapStop):
        print("Process LED " + str(Counter))
        grey_img = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        grey_img = cv2.subtract(grey_img, StartFrame)
        minVal, maxVal, minLoc, maxLoc = cv2.minMaxLoc(grey_img)
        thresh = maxVal * THRESH_MARGIN
        retval, threshold = cv2.threshold(grey_img, thresh, 255, cv2.THRESH_BINARY)
        threshold = cv2.GaussianBlur(threshold, (75,75), 15)
        minVal, maxVal, minLoc, maxLoc = cv2.minMaxLoc(threshold)
        leds.append(maxLoc)
        if SHOW_VIDEO:
            #cv2.imshow('LED threshold', threshold)
            pass
        strip.Set(Counter, (0, 0, 0))
        strip.Show()
        TimerLED = time.time()
        LEDSet = False
        Counter += 1
        if (Counter >= strip.NumPixels()):
            MapStop = True
    if SHOW_VIDEO:
        for led in leds:
            cv2.circle(StartCircleFrame, led, 5, (120,120,120), 2)
        cv2.imshow("Mapping... (Press q for quit)", StartCircleFrame)
        if cv2.waitKey(1) & 0xFF == ord('q') or MapStop:
            break
    if (len(leds) == LED_NUMBER):
        Save["LEDS"] = [led for led in leds]
        with open("leds.json", "w") as outfile:
            json.dump(Save, outfile)
        print("Saved " + str(len(leds)) + " LEDs.")
    else:
        print("ERROR: not enough LEDs mapped.")

cap.release()
cv2.destroyAllWindows()
```



```
pip install pyserial
pip install opencv-python
```

Nel caso in cui utilizzassimo Linux, ad esempio Ubuntu, possiamo installare Python e le librerie necessarie lanciando da terminale questi tre comandi in sequenza:

```
sudo apt install python3 python3-pip
pip3 install pyserial
pip3 install opencv-python
```

Prima di utilizzare gli script Python dobbiamo scegliere un editor per modificare il codice; su Windows possiamo utilizzare l'editor integrato in Python "IDLE": il menu "File" ci consente di aprire un file di codice per modificarlo e per eseguirlo basterà premere il tasto F5.

Un altro editor molto interessante, disponibile sia per Windows che per Linux, è Visual Studio Code reperibile a questo indirizzo <https://code.visualstudio.com/download>; in questo caso, per eseguire il codice bisognerà aprire un prompt dei comandi o un terminale e digitare il comando "python percorso/del/file.py". Bene, a questo punto possiamo procedere con le tre fasi operative del nostro progetto.

Fase 1 - Posizionamento webcam

Apriamo il file *view.py* con l'editor che abbiamo scelto (in questo articolo utilizzeremo Visual Studio Code su Windows) e analizziamolo insieme: nella prima parte importiamo tutte le librerie che servono per il corretto funzionamento dello script, successivamente troviamo una sezione di configurazione dove possiamo impostare l'ID della webcam che vogliamo utilizzare: 0 indica la webcam di default e solitamente bisogna lasciare questo valore a meno che il PC non sia dotato di più webcam, come ad esempio un portatile con la webcam integrata e una connessa tramite USB.

Tramite il metodo `VideoCapture` di OpenCV apriamo uno streaming con la periferica (in questo caso la webcam) e all'interno del ciclo infinito "while (True):" catturiamo un frame alla volta e lo visualizziamo in una finestra.

Per lanciare il programma portiamoci, con il terminale, nella cartella dello script:

```
cd C:\Users\luca\Desktop\OpenCV-NeoPixel-Mapping\
```

E diamo il seguente comando:

```
python view.py
```

Questo programma ci aiuterà solamente a posizionare la webcam frontalmente rispetto all'oggetto da mappare, facendo attenzione che

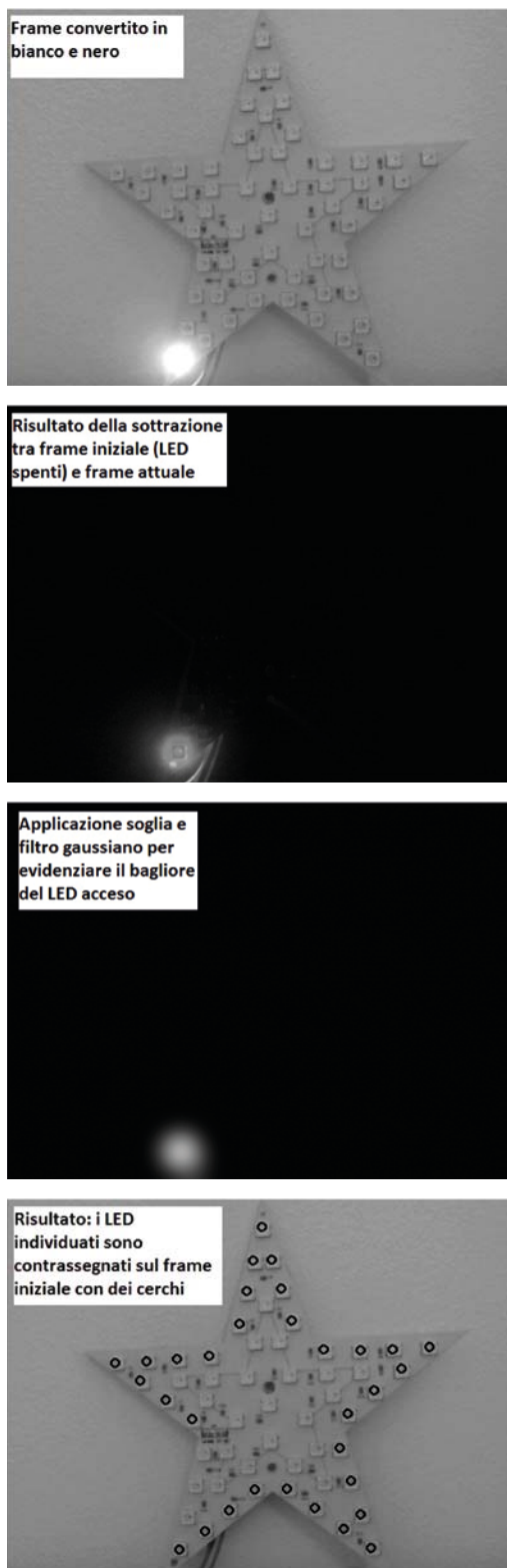


Fig. 3
Alcuni step della fase di mappatura.

▼ Listato 3 - Script per l'esecuzione di effetti luminosi, tramite Miniduino, sui LED mappati.

```
import numpy as np
import neopixel_serial
import time
import json
import os
import sys
sys.path.append("effects/")
from rainbow import Rainbow
from red_white_bars import RedWhiteBars
from color_top_down import ColorTopDown
from random_color_jump import RandomColorJump
# SETTINGS
LED_NUMBER = 56
LED_PORT = "COM3"
LED_SPEED = "250000"
LED_BRIGHTNESS = 6
FRAME_WIDTH = 640
FRAME_HEIGHT = 480
SHOW_VIDEO = True
# Initialize NeoPixel
strip = neopixel_serial.NeoSerial(LED_NUMBER, LED_PORT, LED_SPEED)
strip.SetBrightness(LED_BRIGHTNESS)
time.sleep(1)
leds = []
# Read all JSON file we find in current folder: put all LEDs into leds list with OffsetX and OffsetY
json_files = [pos_json for pos_json in os.listdir(".") if pos_json.endswith('.json')]
for json_file in json_files:
    with open(json_file) as leds_file:
        print("Read file: " + json_file)
        LEDGroup = json.load(leds_file)
        for led in LEDGroup["LEDS"]:
            leds.append((led[0] + LEDGroup["OffsetX"], led[1] + LEDGroup["OffsetY"]))
# Check if LED number is correct
if (LED_NUMBER != len(leds)):
    print("ERROR: wrong LED_NUMBER")
    exit(0)
# Initialize the light effect
effect = Rainbow(w = FRAME_WIDTH, h = FRAME_HEIGHT, speed = 0.01)
#effect = RedWhiteBars(w = FRAME_WIDTH, h = FRAME_HEIGHT, speed = 0.01, thickness = 100)
#effect = ColorTopDown(w = FRAME_WIDTH, h = FRAME_HEIGHT, speed = 0.002, color = (200, 200, 200))
#effect = RandomColorJump(w = FRAME_WIDTH, h = FRAME_HEIGHT, speed = 0.002)
# Start the show
TmrUpdateLEDs = time.time()
while (True):
    # Get frame effect
    frame = effect.GetFrame()
    if SHOW_VIDEO:
        f = frame.copy()
        mask = np.zeros(f.shape[:2], dtype="uint8")
        # For each LED...
        c = 0
        for led in leds:
            # Get ROI frame color and send to neopixel
            roi = frame[led[1]-2:led[1]+2, led[0]-2:led[0]+2]
            color = cv2.mean(roi)
            strip.Set(c, (int(color[2]), int(color[1]), int(color[0])))
            c += 1
            # Print circles on mask
            if SHOW_VIDEO:
                cv2.circle(mask, (led[0], led[1]), 4, (255,255,255), -1)
                cv2.circle(f, (led[0], led[1]), 4, (255,255,255), -1)
        # Update NeoPixels (low frequency)
        if (time.time() >= TmrUpdateLEDs):
            TmrUpdateLEDs = time.time() + 0.1
            strip.Show()

    # Show debug frame on video
    if SHOW_VIDEO:
        masked = cv2.bitwise_and(f, f, mask=mask)
        cv2.imshow("Frame (Press q for quit)", f)
        # If press q: exit the cycle
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
strip.SetAll((0, 0, 0))
strip.Show()
cv2.destroyAllWindows()
```



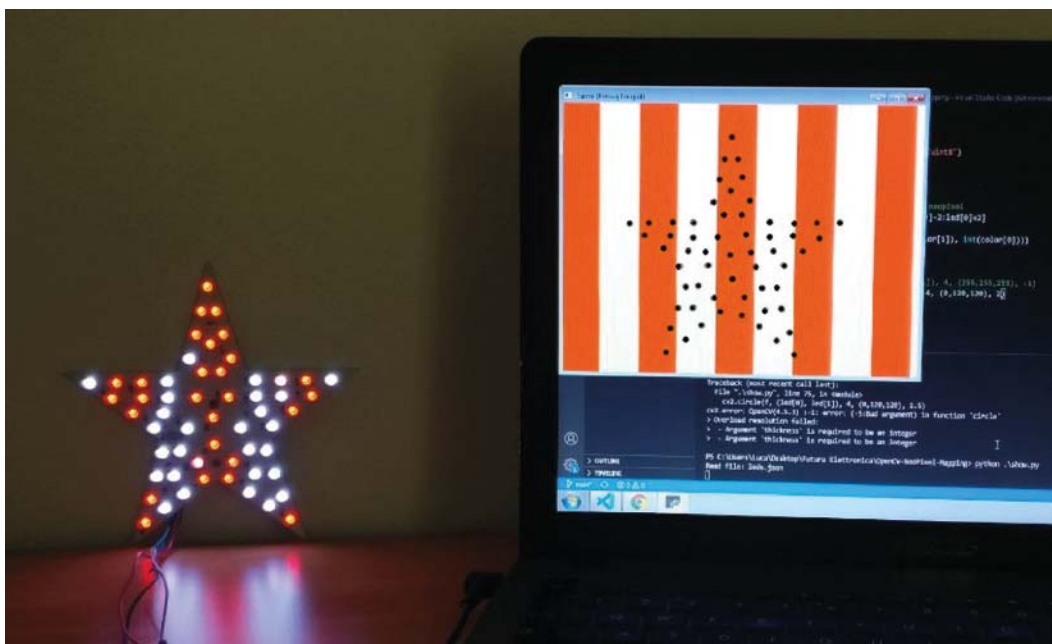


Fig. 4
Effetto luminoso su matrice di LED (cerchi neri) e applicazione su NeoPixel.

sia centrata sia in verticale che in orizzontale; non sono presenti comandi per controllare i LED. Una volta trovata la posizione corretta premiamo il tasto "q" per terminare il programma, avendo cura di non muovere più la webcam.

Fase 2 - Mappatura dei LED

Passiamo ad analizzare il programma più importante dei tre: *map.py*. Lo trovate nel **Listato 2**. Nella sezione iniziale troviamo i seguenti parametri di configurazione:

- LED_NUMBER: totale dei LED connessi a Miniduinò, lo stesso numero che abbiamo inserito nello sketch;
- LED_PORT: porta seriale a cui è connesso Miniduinò (su Windows "COMx" dove x rappresenta un numero);
- LED_SPEED: velocità di comunicazione seriale, uguale a quella impostata nello sketch in "Serial.begin()";
- LED_BRIGHTNESS: luminosità dei LED, da 0 a 100 (durante la fase di mappatura consigliamo di mantenere un valore basso come 10);
- CAMERA_ID: come nel file *view.py*, default 0;
- DELAY_LED: ritardo di accensione del LED rispetto a quello precedente, default 0.1 secondi;
- DELAY_PROCESS: ritardo di processamento del LED, rispetto a quello precedente, default 0.2 secondi;
- THRESH_MARGIN: coefficiente di soglia per individuare il punto più luminoso nell'immagine catturata, default 0.6;

- SHOW_VIDEO: True/False, durante la fase di mappatura mostra una finestra con le posizioni dei LED individuati nell'inquadratura della webcam, utile per capire se il programma sta funzionando correttamente con i parametri impostati.

Subito dopo la parte di configurazione troviamo l'inizializzazione dell'oggetto NeoPixel che si occupa di comunicare a Miniduinò tutte le variazioni dei LED e l'inizializzazione dell'oggetto VideoCapture che, come nel file *view.py*, si occupa di catturare frame dalla webcam.

Il primo frame catturato, con i LED tutti spenti, viene salvato nella variabile StartFrame e ci servirà per trovare le differenze quando avremo catturato un frame con un LED acceso. Tutti i frame che acquisiremo dalla webcam verranno trasformati in bianco e nero in modo da ottenere una matrice di pixel in cui ogni singolo punto può valere da 0 (nero) a 255 (bianco): così facendo viene semplificata e velocizzata la parte di calcolo, in quanto a noi interessa trovare il puntino bianco su uno sfondo grigio-nero che si viene a creare quando accendiamo un LED.

All'interno del ciclo "while (True):" si svolge tutta la fase di mappatura, composta da una serie di operazioni sul frame catturato dalla webcam, che possiamo semplificare con i seguenti step (alcuni dei quali rappresentati graficamente in **Fig. 3**):

1. accendiamo il primo LED di bianco;
2. catturiamo un frame dalla webcam e lo trasformiamo in bianco e nero;

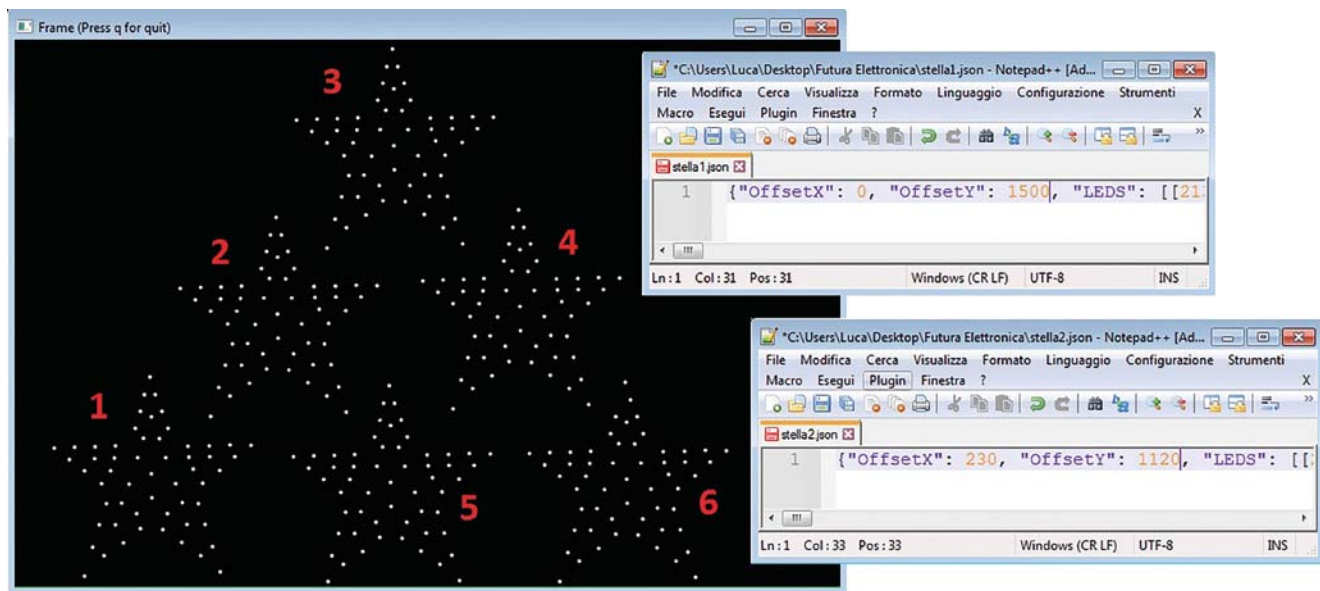


Fig. 5
Diversi stelle
NeoPixel mappate
in momenti diversi
e caricate nello
stesso frame degli
effetti.

3. sottraiamo al frame corrente lo StartFrame: in questo modo otteniamo un frame con le differenze tra un frame e l'altro (ossia solo i pixel bianchi del LED appena acceso);
4. con il metodo minMaxLoc troviamo il punto più luminoso del nuovo frame e lo salviamo nella variabile maxVal, successivamente moltiplichiamo THRESH_MARGIN alla variabile appena trovata per abbassare leggermente il valore del punto più luminoso (creiamo una soglia);
5. tramite il metodo threshold e il filtro gaussiano otteniamo un frame completamente nero con il solo cerchio bianco corrispondente alla posizione del LED;
6. passiamo al metodo minMaxLoc il frame risultante e troviamo così le coordinate x,y del LED, (il centro del cerchio bianco) che salviamo in una lista;
7. spegniamo il primo LED e ripetiamo tutto il procedimento con i LED successivi.

Il ciclo di mappatura si interromperà automaticamente quando saranno stati processati tutti i LED, ma sarà sempre possibile interrompere il ciclo premendo il tasto "q"; alla fine del programma, se tutti i LED sono stati individuati, la lista di coordinate x,y viene salvata in un file JSON dal nome "leds.json".

Le condizioni di illuminazione dell'ambiente, la posizione della webcam e i parametri di configurazione influenzano profondamente la buona riuscita della mappatura: potrebbe essere necessario ripetere più volte il procedimento fino a quando i risultati non saranno soddisfacenti.

Consigliamo di effettuare questa operazione in un ambiente poco illuminato (ma non completamente buio), inquadrando solo l'oggetto da mappare, senza riprendere pavimenti, finestre e specchi che potrebbero ingannare con il riflesso l'algoritmo di riconoscimento della posizione del LED; inoltre la luminosità del LED non dovrà essere troppo elevata per evitare l'effetto "accecamento" della webcam e il conseguente errore di individuazione dello stesso.

Possiamo avere un riscontro in tempo reale settando il parametro SHOW_VIDEO a True prima di eseguire il programma; in questo modo si aprirà una finestra in cui compariranno dei cerchi sopra a ciascun LED individuato: più sarà precisa la posizione del cerchio sopra al LED e migliore sarà la mappatura.

Fase 3 - Effetti luminosi

Dopo aver ottenuto il file "leds.json", siamo pronti a provare i primi effetti luminosi; in questa fase la webcam non è più necessaria in quanto controlleremo direttamente i LED con Miniduino tramite script Python. Il codice corrispondente è quello riportato nel **Listato 3**.

Apriamo il file *show.py* e impostiamo i parametri LED_NUMBER, LED_PORT e LED_SPEED esattamente come nel programma precedente, mentre possiamo impostare LED_BRIGHTNESS a un valore più alto rispetto a quello utilizzato durante la mappatura.

Consigliamo di impostare i parametri FRAME_WIDTH e FRAME_HEIGHT con la risoluzione

della webcam (ad esempio 800 x 600) in modo da andare a creare una finestra che contenga tutti i LED appena mappati (possiamo lanciare il programma più volte per trovare le dimensioni adatte); lasciamo il parametro `SHOW_VIDEO` a `True` in modo che venga visualizzata la finestra che mostra l'andamento dell'effetto luminoso. Dopo l'inizializzazione dell'oggetto per il controllo dei NeoPixel e il caricamento delle posizioni dei LED, possiamo trovare la dichiarazione dell'effetto che vogliamo applicare: dobbiamo dichiararne solo uno alla volta mentre lasciamo gli altri con il simbolo `#` davanti; ad esempio, se volessimo inizializzare l'effetto `ColorTopDown` togliamo il `#` all'inizio della riga e otterremo:

```
effect = ColorTopDown(w = FRAME_WIDTH, h = FRAME_HEIGHT, speed = 0.2, color = (200, 200, 200))
```

Come possiamo notare, stiamo passando dei parametri tra parentesi: senza mai modificare i parametri `w` e `h` possiamo comunque personalizzare la velocità (in secondi) dell'effetto e il colore (valori B, G, R da 0 a 255 ciascuno).

La logica del programma è quella di creare un frame video in cui, step dopo step, vengono eseguiti degli effetti colorati in movimento tramite delle operazioni matematiche sui pixel; proiettando i LED sul frame troviamo il colore che dovrebbero avere in quell'esatto istante per ciascuna posizione `x,y` e lo settiamo fisicamente ai LED tramite Miniduino.

Nel dettaglio, all'inizio del ciclo chiediamo un frame aggiornato al nostro effetto invocando il metodo `"GetFrame()"`; andiamo a scorrere l'elenco di tutti i LED e li proiettiamo sul frame, poi per ciascun LED calcoliamo la media del colore e la impostiamo tramite l'istruzione `"strip.Set()"`. Dopo aver settato tutti i LED, alla fine del ciclo, utilizziamo `"strip.Show()"` per aggiornare tutti i LED contemporaneamente: tutte queste operazioni, eseguite ad alte velocità, creeranno l'effetto luminoso in movimento. Anche in questo caso, se abbiamo lasciato `SHOW_VIDEO` a `True`, è possibile interrompere il programma con il tasto `"q"` oppure possiamo interromperlo in tutti i casi con i tasti `CTRL + C` sul terminale.

In **Fig. 4** vediamo l'effetto `"RedWhiteBars"` applicato sulla stella NeoPixel e, sul monitor del PC, la finestra che contiene il frame generato dall'algoritmo dell'effetto luminoso; notiamo anche dei cerchi scuri che rappresentano le posizioni dei LED da cui vengono prelevati i colori inviati a Miniduino.

Nel caso in cui avessimo diversi tipi di NeoPixel

connessi tra loro, dovremo posizionare la webcam in modo tale da inquadrarli tutti: potrebbe capitare però di non riuscire a mapparli tutti insieme a causa della troppa distanza tra la webcam e i LED oppure a causa della scarsa risoluzione della webcam.

In questo caso è possibile mappare un oggetto NeoPixel alla volta e rinominare il file JSON che viene creato alla fine di ogni esecuzione di `map.py` in modo da ottenere, alla fine di tutte le mappature, un elenco di file del tipo: `stella1.json`, `stella2.json`, `catena.json`, ... Il programma `show.py`, una volta lanciato, caricherà in memoria tutti i file JSON che si trovano nella sua stessa cartella, mostrando però, nella finestra video, tutti i gruppi di LED sovrapposti. Per ovviare a questo problema possiamo aprire ciascun file JSON con un editor di testo (come Notepad++ o il Blocco Note di Windows) e andare a modificare in esso i valori di `"OffsetX"` e `"OffsetY"` per traslare gli oggetti nello spazio della finestra. Lanciamo `show.py` ad ogni modifica di valori e parametri, fino a quando non saremo soddisfatti delle posizioni di tutti gli oggetti caricati; ricordiamoci di agire anche sui parametri `FRAME_WIDTH` e `FRAME_HEIGHT` per allargare la finestra e creare lo spazio necessario a ospitare i vari oggetti di LED (**Fig. 5**).

CREAZIONE DI EFFETTI LUMINOSI

Gli algoritmi per creare effetti luminosi sono quattro: `color_top_down`, `rainbow`, `random_color_jump`, `red_white_bars`; per creare un nuovo effetto, possiamo copiare uno dei file esistenti e rinominarlo con il nome del nuovo effetto (esempio: `"nuovo_effetto.py"`), successivamente apriamo il file con il nostro editor e modifichiamo la definizione della classe avendo cura di scrivere il nome senza spazi (esempio: `"class NuovoEffetto:"`). Possiamo notare tre metodi:

- **`__init__`**: inizializza la classe del nostro effetto e riceve i parametri essenziali per il proprio funzionamento, come le dimensioni del frame video in cui verrà creato l'effetto, la velocità di esecuzione, eccetera...;
- **`GetFrame`**: chiama il metodo di creazione del nuovo frame, calcola il tempo impiegato a generarlo e, se necessario, aggiunge un ulteriore delay per restituire il frame esattamente alla velocità impostata in fase di inizializzazione;
- **`MakeFrameEffect`**: metodo che si occupa effettivamente, tramite un algoritmo, di generare l'effetto frame dopo frame.

All'interno dell'ultimo metodo dovremo inserire le



istruzioni per disegnare e traslare forme geometriche colorate mediante funzioni grafiche di OpenCV e calcoli matematici.

Consideriamo il nostro frame come un piano cartesiano dove l'origine degli assi è in alto a sinistra (0, 0), il valore massimo della X è FRAME_WIDTH mentre il valore massimo della Y è FRAME_HEIGHT e analizziamo ora alcune funzioni grafiche utili per disegnare sul frame:

- `cv2.line(frame, (x1,y1), (x2,y2), (b, g, r), spessore):` consente di disegnare una linea sul frame indicato, partendo dalle coordinate x1,y1 fino alle coordinate x2,y2 con il colore (b,g,r) e lo spessore in pixel indicato;
- `cv2.rectangle(frame, (x1,y1), (x2,y2), (b, g, r), spessore):` consente di disegnare un rettangolo sul frame indicato partendo dalle coordinate x1,y1 (vertice in alto a sinistra) fino alle coordinate x2,y2 (vertice in basso a destra) con il colore (b,g,r) e lo spessore di bordi indicato (se lo spessore è negativo, il rettangolo viene disegnato pieno a tinta unita);
- `cv2.ellipse(frame, (x1,y1), (ax,ay), ang, a0, a1, (b,g,r), spessore):` consente di disegnare un'ellisse sul frame indicato con il centro in x1,y1, asse in ax,ay con inclinazione ang, angolo iniziale a0, angolo finale a1 del colore (b,g,r) e dello spessore indicato (se lo spessore è negativo, l'ellisse viene disegnato pieno a tinta unita).

Dopo aver creato il nostro effetto, è necessario includerlo nel file `show.py` inserendo il comando di importazione accanto a quello relativo agli altri effetti:

```
from nuovo_effetto import NuovoEffetto
```

Inoltre, nella sezione di inizializzazione effetti, è necessario andare a inserire anche quella del nuovo effetto:

```
effect = NuovoEffetto(w = FRAME_WIDTH, h = FRAME_HEIGHT, speed = 0.2, parametro1 = 123)
```

Se volessimo lasciare in esecuzione l'effetto luminoso senza tenere acceso il PC, possiamo utilizzare un mini-computer Linux come Raspberry Pi. Allo scopo, scarichiamo l'immagine del sistema operativo senza interfaccia dal web, all'indirizzo <https://www.raspberrypi.com/software/operating-systems/> e prima di estrarre la MicroSD dal PC, creiamo un file vuoto nella cartella principale della

scheda e rinominiamolo in "ssh" (senza estensione): questo ci servirà per abilitare la connessione SSH utile per l'installazione del software tramite terminale.

Creiamo anche un secondo file "wpa_supplicant.conf" che permetterà al nostro Single Board Computer Raspberry Pi di connettersi alla nostra rete WiFi e all'interno inseriamo le seguenti righe, modificandole opportunamente:

```
country=IT
ctrl_interface=DIR=/var/run/wpa_supplicant
GROUP=netdev
network={
    ssid="INSERIRE_NOME_RETE_WIFI"
    psk="INSERIRE_PASSWORD_RETE_WIFI"
    key_mgmt=WPA-PSK
}
```

Inseriamo la MicroSD-Card nell'apposito socket della nostra scheda Raspberry Pi, alimentiamolo e connettiamoci in SSH per preparare la parte software (download file progetto da GitHub, installazione di Python e delle relative librerie) come abbiamo visto nel paragrafo "Software"; ricordiamoci di trasferire i file JSON della mappatura all'interno della cartella appena scaricata, colleghiamo Microduino a una porta USB e aggiorniamo i parametri come la nuova porta seriale e SHOW_VIDEO, che va posto a False in quanto non disporremo di uno schermo per visualizzare la finestra.

Lanciamo lo script `show.py` e godiamoci gli effetti luminosi creati sui nostri LED: allieteranno le Festività di fine anno! □



Cosa occorre?

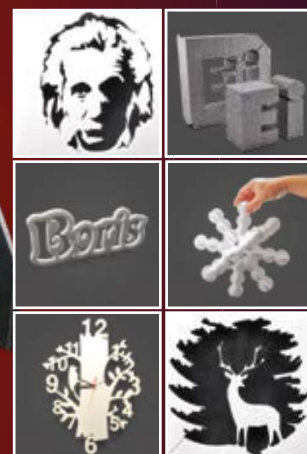
I componenti utilizzati in questo progetto sono disponibili presso Futura Elettronica. La board Miniduino (cod. MINIDUINO) è in vendita a Euro 22,00, la stella Neopixel (cod. FT1300M) è disponibile a Euro 35,00, la strip da 150 LED NeoPixel (cod. STRIP150LED) costa Euro 46,00, mentre la versione da 300 LED (cod. STRIP300LED) è disponibile a Euro 74,00. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

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



DAI FORMA ALLE TUE IDEE!



cod. FOAMCUTTER

€ 320,00

PLOTTER TAGLIA POSTIROLO - IN KIT

Realizza, intagliandole a caldo, sagome di ogni genere partendo da lastre di polistirolo, polistirene ed altri materiali espansi a bassa temperatura di fusione. Collegalo subito al PC e utilizzalo per realizzare tutto quello che vuoi: modelli di vario genere, scritte, scatole, sagome per decorazioni, lettere scatolate, loghi, simboli, puzzle 3D e molto altro!

CARATTERISTICHE TECNICHE:

- **Tecnica di taglio:** a filo caldo
- **Materiali lavorabili:** espansi a bassa temperatura di fusione
- **Dimensioni:**
 - **Plotter:** 775x690x220 mm
 - **Area lavorabile:** 500x480 mm
 - **Spessore lavorabile:** 120 mm
- **Risoluzione:** 0,5 mm
- **Tolleranza X Y:** 0,012 mm
- **Velocità di taglio:** 300 mm/minuto (varia in funzione della tipologia di materiale lavorato)
- **Alimentazione:** 12 VDC / 3 A
- **Temperatura filo:** regolabile da software
- **Gestione:** da PC tramite interfaccia USB

Prezzo IVA inclusa.

**FUTURA
ELETTRONICA®**

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

www.futurashop.it

LED COB

LIGHTS

**100 LUMEN
WATT**



Realizza lettere, numeri per pannelli pubblicitari, illumina al meglio i prodotti nella tua vetrina e tanto altro con le barrette a LED COB. Ricoperte da gomma siliconica sono realizzate su supporto in alluminio per una migliore dissipazione di calore. Utilissime per svariati utilizzi!



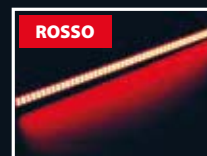
Cod. LEDCOB200B



Cod. LEDCOB200W



Cod. LEDCOB200V



Cod. LEDCOB200R

BARRETTE DI LED COB 12V-15W LUNGHEZZA 20 CM

Tipo di LED: COB chip
Angolo del fascio: 120°
Flusso luminoso: 100 lumen/watt
Vita media: 50.000 ore
Alimentazione: 12 VDC
Potenza: 15W
Consumo: 900 mA
Temperatura di funzionamento: da -20°C a +60°C
Dimensioni: 200x10x3 mm
Peso: 11 grammi

€5,90
A BARRETTA

**Collega
insieme più
barrette
di colori
differenti!**

Prezzi IVA inclusa.

Alimentale con:

**MAX 1
barretta di
LED COB**

ALIMENTATORE SWITCHING
12V - 15W - 1,25A

€5,00

Cod. POS15W12V



**MAX 2
barrette di
LED COB**

ALIMENTATORE
SWITCHING
12V - 36W - 3A

€8,50

Cod. ADL-40-12



**MAX 6
barrette di
LED COB**

ALIMENTATORE
SWITCHING
12V - 100W -
8,5A

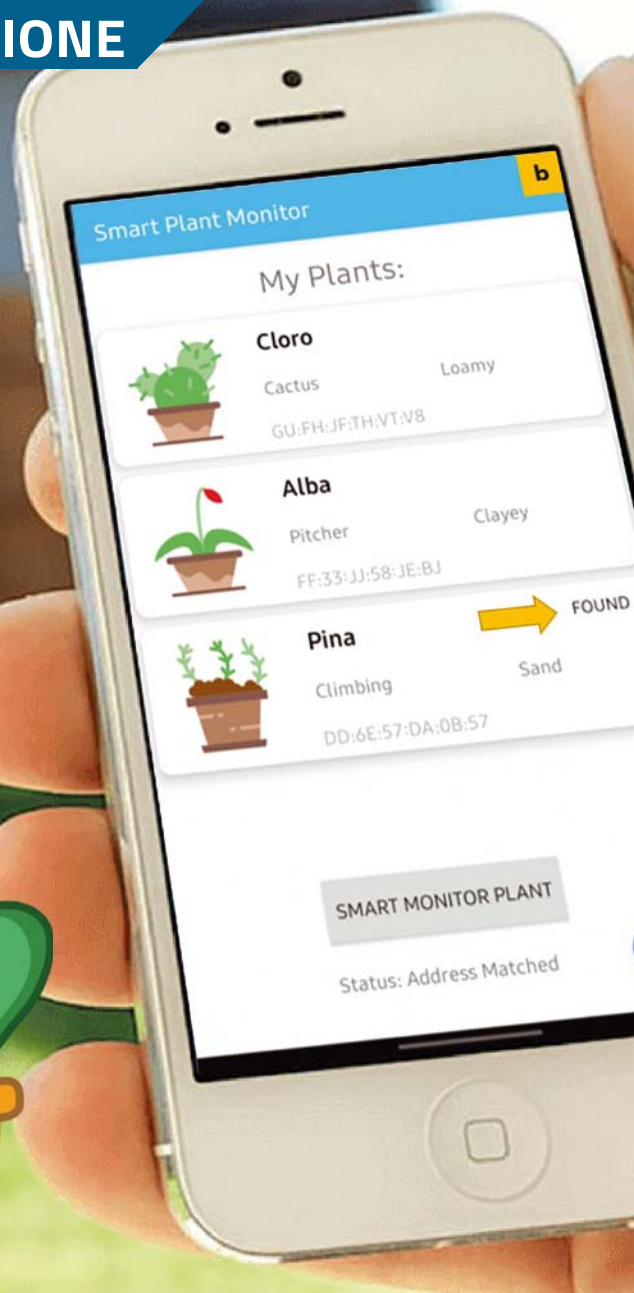
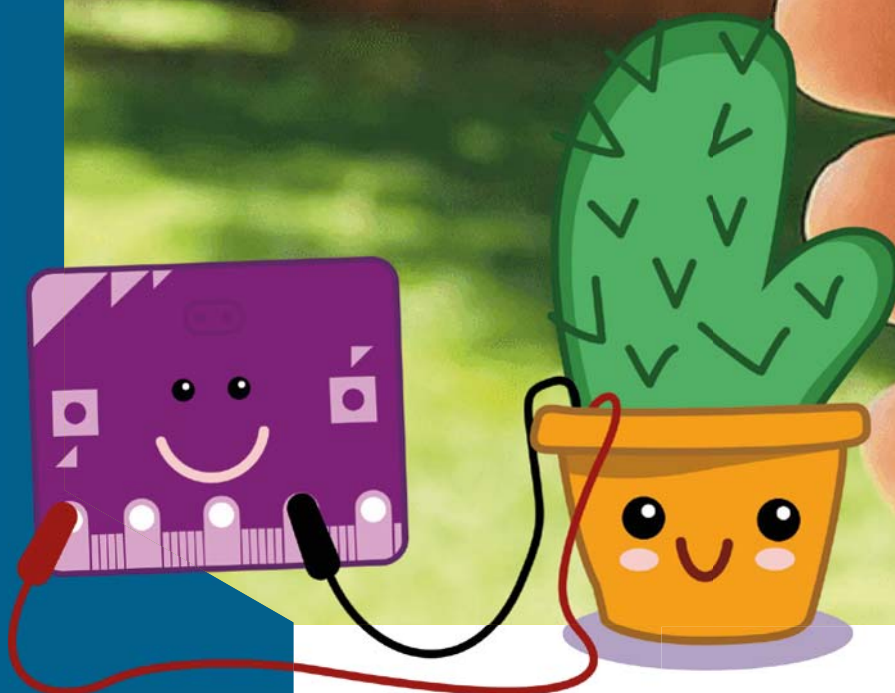
€20,00

Cod. MW10012



PLANT SMART MONITOR

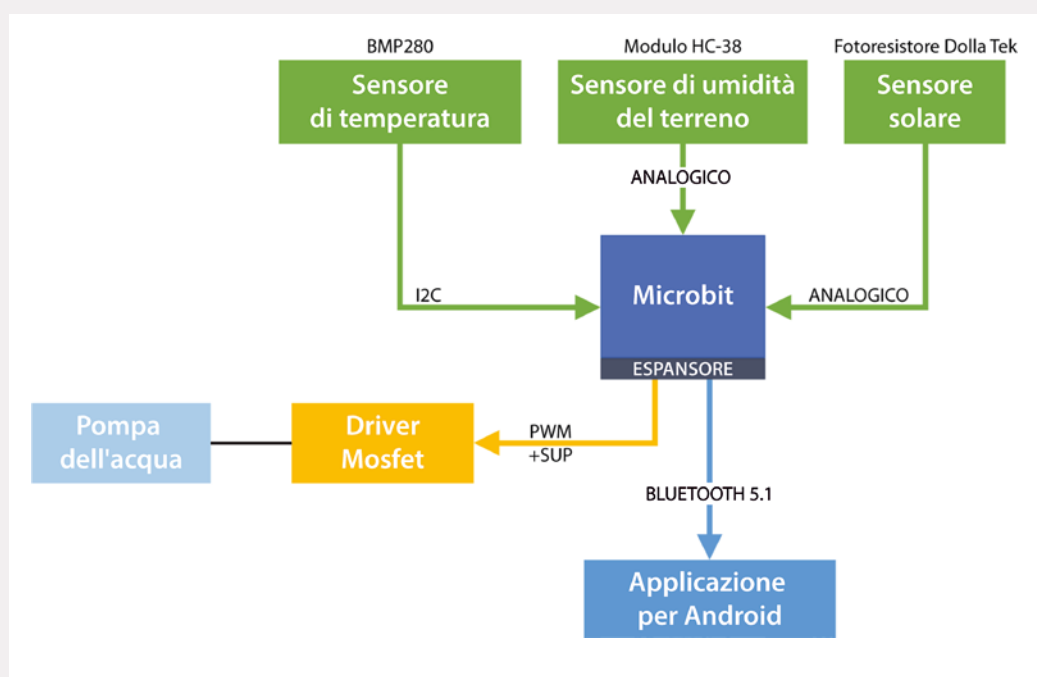
degli ing. **VINCENZO GERMANO**
e **MARCONATALE PARISE**



Utilizziamo la flessibilità e potenza di micro:bit per realizzare un sistema di irrigazione intelligente delle piante, portatile ed interconnesso.

In un mondo in completa evoluzione, in cui si fa sempre più attenzione al futuro del pianeta, si mixano tecnologie come l'IoT e l'intelligenza artificiale per cercare di ottimizzare i consumi di acqua a 360 gradi. Si sviluppano sistemi per monitorare da vicino il modo in cui le piante rispondono a stress, lesioni, infezioni e alla luce, utilizzando sensori realizzati con nanotubi di carbonio. In questo contesto ci sentiamo di dare il nostro piccolo contributo, proponendo un sistema di monitoraggio intelligente delle piante (per comodità di seguito utilizzeremo l'acronimo **PSM** dall'inglese **Plant Smart Monitor – Monitoraggio Intelligente della Pianta**) per fornire la possibilità all'utente di monitorare l'habitat della pianta in ambito domestico/lavorativo e permettergli una gestione automatica o manuale del sistema d'irrigazione. In estrema sintesi, l'idea nasce dalla necessità di irrigare al

Fig. 1
Schema a blocchi
PSM.



meglio le piante in modo automatico, non solo per ridurre gli impegni a cui siamo chiamati nella vita “frenetica” di ogni giorno ma anche per consentire a qualsiasi utente di prendersi cura delle proprie piante ottimizzando l’acqua necessaria per il corretto sostentamento, grazie al supporto di svariati parametri a disposizione che affronteremo e mostreremo durante l’articolo.

Vi guideremo nella realizzazione del prototipo dal punto di vista hardware nella descrizione dei componenti e nella realizzazione del setup, ma anche dal punto di vista software nella scrittura del codice necessario per l’implementazione delle funzioni sopra elencate. Oltre a questo, per rendere il tutto il più flessibile possibile è stata realizzata un’applicazione per Android che permette la configurazione del sistema inserendo la tipologia di pianta e il tipo di terriccio in cui è piantata, in modo tale da permettere una configurazione immediata anche ai meno esperti. Perciò dopo la presentazione del dispositivo stand-alone, verrà presentata l’applicazione in linea con le richieste di mercato odierno, volte a garantire l’interconnessione tra i dispositivi con un occhio verso possibili e future applicazioni.

DESCRIZIONE HARDWARE

Andiamo per gradi, il PSM ha come cuore di tutto il sistema la scheda micro:bit, collegata opportunamente ad alcuni sensori per rilevare lo stato della pianta e permettere di intervenire per garantirne lo

stato ottimale. Com’è facile immaginare per poter controllare i parametri di una pianta, i sensori coinvolti devono essere almeno relativi a temperatura d’ambiente, umidità del suolo e radiazione solare per comprendere se l’irrigazione avviene di giorno o di notte. Per permettere un facile reperimento del materiale per l’implementazione di questo progetto ed evitare la realizzazione di board dedicate, in questo articolo mostreremo alcune soluzioni di schede elettroniche integrate “pronte all’uso” a costi contenuti che permettono la realizzazione delle funzioni d’interesse. Visto che il progetto non è solo un monitor per il controllo dello stato dell’ambiente in cui si trova la pianta, alla sensoristica viene aggiunto un motore gestito attraverso un driver direttamente dal micro:bit che avrà il compito di implementare la funzione della pompa dell’acqua (qualora le condizioni lo dovessero richiedere) e quindi annaffiare la pianta per garantire una crescita continua e costante.

Per poter meglio comprendere il sistema, partiamo dall’architettura schematizzata mediante un diagramma a blocchi, per poi passare ai vari dettagli d’implementazione hardware/software.

Facendo riferimento alla **Fig. 1**, come sottolineato in precedenza, il compito principale è affidato alla scheda micro:bit, che come illustrato acquisisce le grandezze fornite dai sensori (in figura riportate per semplicità con delle frecce verdi uscenti dai sensori ed entranti nel micro:bit), pilota il driver del

LA SCHEDA D'ESPANSIONE

Ci sono alcune soluzioni possibili sul mercato ma tra le varie proposte abbiamo scelto un "*Port Expander*" acquistabile sullo store *futurashop.it* con il codice MICROBITPROTO in grado di rendere di facile utilizzo tutte le porte di input/output della board.

Ulteriori dettagli sono visibili, ovvero la scheda ha la possibilità di essere alimentata tramite i due connettori rossi, con tensioni variabili fino a 24V o fino a 6V (a seconda del connettore scelto), per sfruttare uno dei due regolatori stabilizzati di tensione da 5V o 3.3V. Altra possibilità è data dal fatto che la scheda può anche essere alimentata direttamente tramite la porta USB di micro:bit. Essendo puramente una scheda d'espansione non ha molte altre caratteristiche se non un LED di alimentazione e vari pin header con passo 2,54mm, con le seguenti funzionalità:

- colore giallo e blu per le porte di I/O e protocolli di comunicazione (I2C, SPI e Seriale)
- colore blu parzialmente per il riferimento GND
- colore rosso per le tensioni d'alimentazione

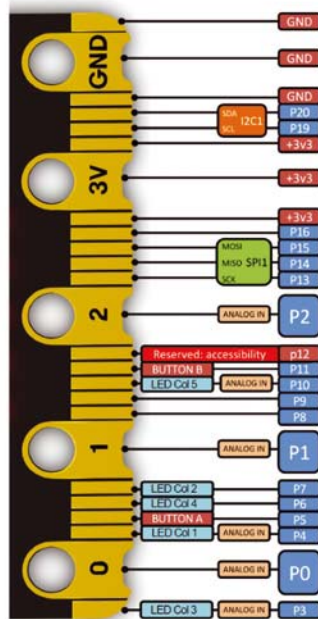
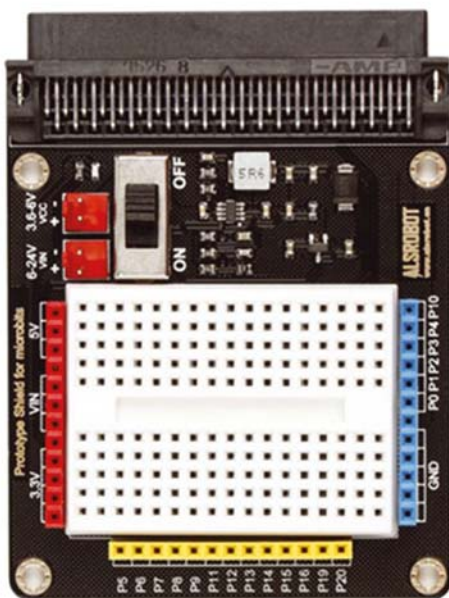
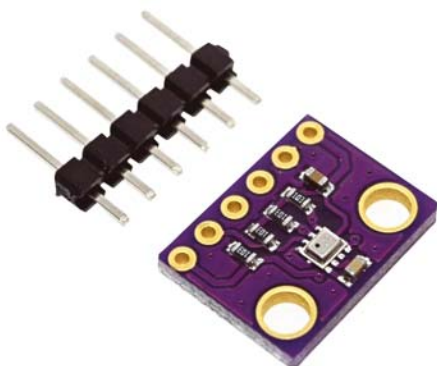


Fig. 2
Port
Expander e
pinout.

Fig. 3
Scheda BMP280.



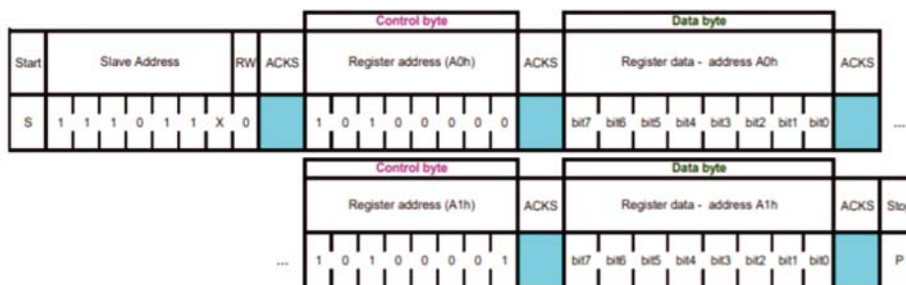
Ulteriore vantaggio è dato dalla disponibilità di un interruttore a due posizioni (on/off) che permette di accendere e spegnere direttamente la scheda. È facile immaginare come l'utilizzo di questa board risulta essenziale quando si vogliono connettere immediatamente componenti esterni al connettore micro:bit, come possono essere schede di sensori che richiedono specifici collegamenti fisici magari complessi da collegare direttamente al micro:bit.

IL SENSORE DI TEMPERATURA

Con riferimento alla **Fig. 1**, avendo evidenziato l'utilizzo di differenti moduli sensore, possiamo

all'analisi di ognuno di essi iniziando da uno dei più complessi, ovvero il BMP280 (cod. BMP280 su *futurashop.it*), che verrà impiegato per rilevare la temperatura d'ambiente. Questo è un sensore di pressione barometrica assoluta, particolarmente adatto per applicazioni mobili, le sue piccole dimensioni e il suo basso consumo energetico ne consentono l'implementazione in dispositivi alimentati a batteria come telefoni cellulari, moduli GPS o orologi. Si basa sulla comprovata tecnologia di sensori di pressione piezoresistivi di Bosch, caratterizzata da elevata precisione e linearità, nonché stabilità a lungo termine ed elevata robustezza EMC (Electromagnetic Compatibility o compatibilità elettromagnetica). Perciò possiamo considerarlo come un nuovo sensore ambientale per temperatura e pressione barometrica che insieme al BME280, sono l'evoluzione della precedente serie BMP085/BMP180/BMP183. Questo prodotto è in grado di effettuare misurazioni molto "fini" con un'accuratezza assoluta di ± 1 hPa per la pressione e di $\pm 1.0^\circ\text{C}$ per la temperatura, grazie alle quali può essere utilizzato anche come altimetro con un'accuratezza di 1m. Numerose opzioni di funzionamento del dispositivo garantiscono la massima flessibilità, in più è ottimizzato in termini di consumo energetico, risoluzione e prestazioni.

Write Operation



Read Operation

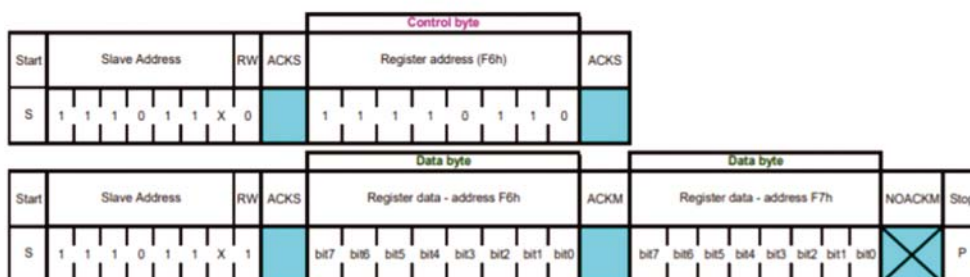


Fig. 4
Protocollo I2C.

Per dare un'idea, il chip consente di effettuare letture con un campionamento selezionabile e una risoluzione che varia da 0,0050 °C a 0,0003 °C, con consumi dipendenti da una delle seguenti possibili modalità operative:

- **Sleep mode**: non vengono effettuate misure e i consumi sono ridotti al minimo
- **Force mode**: viene effettuata una singola misura con successiva transizione in sleep mode
- **Normal mode**: consiste in cicli di lettura continui interrotti da un periodo di inattività

Senza addentrarci troppo in tecnicismi, dovendolo utilizzare come sensore di temperatura ambientale, passiamo rapidamente al suo utilizzo ovvero alla possibilità di connetterlo con il micro:bit. In commercio esistono varie schede già realizzate e acquistabili con costi ridotti come quella in **Fig. 3**. La scheda presenta un regolatore a 3.3V per l'alimentazione stabilizzata del sensore e un convertitore di livello per funzionare sia con logica a 3.3V che a 5V; è possibile effettuare letture di pressione e temperatura abilitabili individualmente, che vengono rese disponibili tramite opportuni registri al quale è possibile accedere mediante i protocolli di comunicazione SPI (Serial Peripheral Interface) oppure I2C (Inter Integrated Circuit). Per questa implementazione viene scelta l'interfaccia I2C, abilitabile tramite il pin CS (Chip Select), grazie ad un opportuno collegamento alla tensione di alimentazione tramite una resistenza di 1KΩ. Come in tutti i dispositivi con questa tecnologia di comunicazione, è possibile interrogare il dispositivo tramite un determinato indirizzo (indirizzo slave), l'indirizzo slave si compone di 7bit (111011x) di cui 6 statici e uno dinamico che varia in base al desiderato, ovvero collegando il pin SDO a GND si genera l'indirizzo 0x76(1110110) oppure collegandolo alla tensione di alimentazione (ad esempio tramite una resistenza di 1KΩ) si genera l'indirizzo 0x77(1110111). Nel nostro caso l'indirizzo scelto del dispositivo sarà 0x76.

Per completezza, le linee SCK (Serial Clock) ed SDI (Serial Data Input) dovranno essere collegate ai rispettivi pin lato micro:bit, con l'accortezza di connettere la linea SDI alla tensione di alimentazione tramite una resistenza (va bene anche un valore dell'ordine di 1KΩ). I *frame* di lettura e scrittura tramite protocollo I2C sono illustrate in **Fig. 4**. Vediamoli brevemente entrambi. L'operazione di scrittura consiste nell'inviare:

- l'indirizzo del dispositivo (slave) con il bit RW impostato a 0

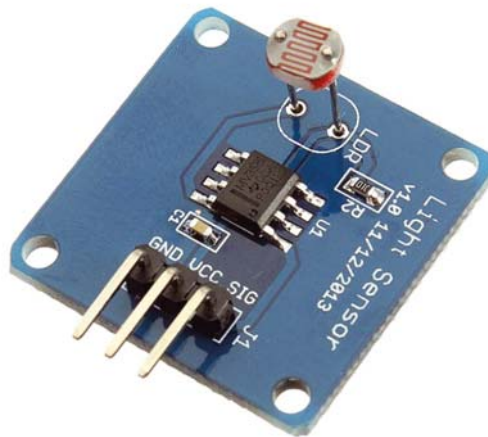


Fig. 5
Modulo
Sensore luce.

- la coppia di registri denominati "Register address" che rappresenta il registro che si vuole scrivere e il registro "Register data" che dovrà contenere il dato che dovrà essere scritto nel "Register address".

La procedura termina quando è intercettata la condizione di stop (P) caratterizzato da una transizione da basso ad alto del SDA mentre il SCL è alto. Mentre, per l'operazione di lettura:

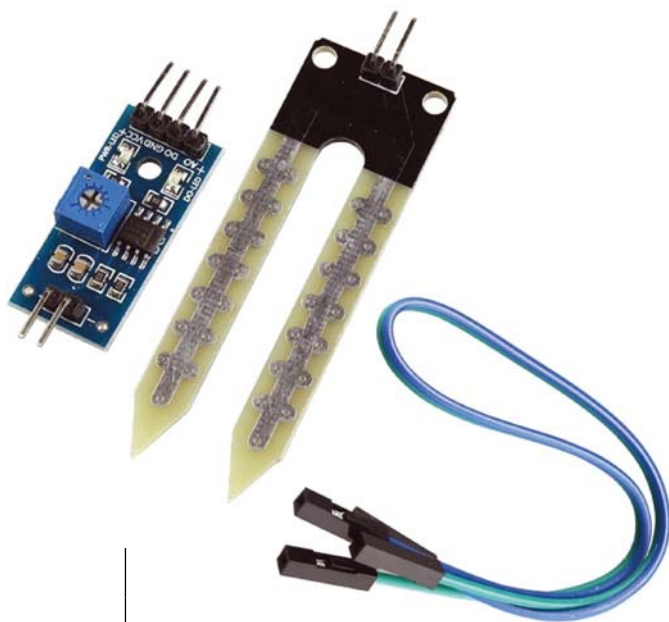
- Si configura il bit RW a 0 e si invia il registro denominato "Register address" da leggere
- Con una nuova condizione di start "S", si configura il bit RW a 1
- Successivamente il dispositivo invierà il dato del registro
- La procedura terminerà con un NOACKM ed una condizione di stop (P)

Le diciture ACKM, ACKS sono definite come *Acknowledge* lato master oppure lato slave che può indicare l'inizio di un'operazione identificata da una forzatura bassa della linea dati durante l'impulso alto del clock (SCL). I registri interessati per la lettura temperatura sono 0xFA, 0xFB e 0xFC. Per i meno esperti, fortunatamente, come vedremo successivamente le librerie software utilizzate ci astrarranno da tutti questi dettagli e impostazioni, permettendoci di semplificare enormemente lo sviluppo software e rendendo la sua configurazione immediata, illustrando le funzioni più importanti per implementare la lettura dei dati d'interesse.

IL SENSORE LUCE

Per rilevare l'intensità luminosa, tra i vari moduli disponibili sul mercato la scelta è ricaduta sulla scheda LIGHTSENSOR, visibile in **Fig. 5**. Questa board ha una resistenza fotosensibile (la fotoresistenza o il fotoresistore è una resistenza composta da uno strato semiconduttore che reagisce alla

 **Fig. 6**
Modulo
Sensore
umidità.



radiazione luminosa, sia emessa dal sole che da altra fonte luminosa) che consente di rilevare la luminosità della luce ambientale.

Il funzionamento di una fotoresistenza, è molto semplice perché varia il valore da 0Ω alla sua resistenza massima 10, 50, 100K Ω , in base alla tipologia di fotoresistenza, quando passa da una condizione di illuminazione piena al buio.

In pratica se la luce è sufficientemente forte il suo valore in ohm è tendente a zero, mentre man mano che la luce diminuisce, il suo valore aumenta fino alla resistenza massima.

Quindi riassumendo è un componente elettronico la cui resistenza è inversamente proporzionale alla quantità di luce che lo colpisce e ciò comporta che la corrente elettrica che transita attraverso tale componente è proporzionale all'intensità di una sorgente luminosa. In tale maniera si realizza una sorta di potenziometro attuabile tramite la luce, anziché tramite forze meccaniche o segnali elettrici. Come apprezzabile dalla **Fig. 5**, la scheda ha un pin header da 3 pin, ovvero:

- **VCC**: alimentazione 3.3-5V
- **GND**: zero volt
- **SIG**: uscita del segnale da 0V (valore) a 5V (valore alto)

Tale segnale è possibile collegarlo direttamente al micro:bit per poter rilevare i cambiamenti dell'intensità della luce ambientale.

Questa scheda, come altre, ha anche dei comodi fori di fissaggio per una facile installazione mediante delle viti.

IL SENSORE DI UMIDITÀ

Per rilevare il contenuto di umidità del terreno abbiamo utilizzato la scheda IGROSENS visibile in **Fig. 6**: si può osservare come il modulo si compone di una sonda (collegata alla scheda principale mediante un doppio connettore), che a contatto con il terreno rileva le informazioni d'interesse riguardo l'umidità e le trasferisce direttamente al modulo sensore a cui è collegata; tali informazioni vengono poi "gestite" dalla scheda e tradotte in un segnale analogico e digitale.

La scheda, oltre ai due pin di ingresso della lettura del sensore, dalla parte opposta ha un pin header da 4 vie, con i seguenti segnali come da numerazione dell'immagine in **Fig. 6**:

1. **VCC**: tensione stabilizzata di funzionamento a 3.3V-5V
2. **GND**: zero volt
3. **DO**: uscita digitale (Digital Output), che può assumere un valore basso o alto (0 e 1)
4. **AO**: uscita analogica (Analog Output)

Entrambe le uscite possono essere collegate direttamente al microcontrollore per rilevare il segnale d'interesse dell'umidità ma com'è facile immaginare i due contenuti informativi sono differenti e quindi assumono scopi diversi; ad esempio nel caso del DO, essendo un valore discreto alto/basso, impostando una soglia mediante il potenziometro presente sulla scheda, si può scegliere un valore per abilitare/disabilitare direttamente un relè, magari in grado di pilotare un modulo buzzer a formare un sistema di allarme di umidità del suolo; oppure leggerlo direttamente da microcontrollore per altri scopi, certo è che una rilevazione di questo tipo è una "lettura di soglia".

Nel nostro caso, volendo essere più "fini" nella lettura dell'umidità del suolo, abbiamo scelto l'utilizzo del segnale analogico sul pin AO, che permette di rilevare tutto il range di valori e non una singola soglia d'allarme. Come da **Fig. 1**, questa scheda viene collegata (mediante pin AO) direttamente al micro:bit che tramite un convertitore analogico digitale consente di ottenere tutti i valori d'interesse.

LA POMPA AD ACQUA E IL DRIVER

Infine, l'ultimo modulo utilizzato è quello di attuazione della pompa. Dopo aver rilevato e analizzato lo stato dei sensori, micro:bit "decide" se irrigare il suolo o meno ma anche quanto irrigarlo.

Nella scelta di sistema tra le varie tipologie di motori, abbiamo deciso di utilizzare direttamente una pompa ad acqua, visto che abbiamo immaginato di

irrigare più piante (pensando ad un giardino). La mini pompa a immersione scelta (cod. DC12V3WPUMP) è acquistabile su sito *futurashop.it*, ed è in grado di far risalire un liquido in superficie con una portata di 200 litri/ora (circa 3,3 litri/minuto).

Realizzata in ABS, dotata di motore in corrente continua, albero motore in acciaio inossidabile, statore e circuito sigillati con resina epossidica. Nella parte frontale è presente un foro con apertura di aspirazione regolabile, mentre nella parte superiore è disponibile l'attacco per il tubo di uscita del liquido.

Tra le varie caratteristiche abbiamo, piccole dimensioni, alta efficienza, basso consumo e basso rumore (meno di 35 dB), eccellente durata: più di 30.000 ore.

A questo punto, con anche la pompa ad acqua a disposizione, l'unico elemento mancante è un circuito di pilotaggio della stessa, che per semplicità è stato scelto dalla componentistica comunemente utilizzata anche per il modellismo dinamico, ovvero un Driver basato su MOSFET (*Metal Oxide Semiconductor Field Effect Transistor*), nient'altro che un circuito elettronico integrato in grado di regolare la velocità del motore (nel nostro caso la pompa ad acqua) in base ad un segnale di pilotaggio (cod. MOSDRIVER), ma vediamo meglio come funziona. Le interconnessioni possono essere schematizzate come nella **Fig. 8**, ovvero per semplicità possiamo dire che il blocco Driver ha a bordo l'elettronica che pilota mediante le due fasi il motore della pompa ad acqua e ne varia la velocità a seconda del segnale d'ingresso proveniente dal micro:bit, perciò sono previsti:

- un'alimentazione esterna proveniente generalmente da una batteria a 12V in grado di erogare fino a 1A, con la quale viene alimentato direttamente il motore;
- un connettore con un segnale di controllo e le alimentazioni per l'interfaccia digitale.



Fig. 7
Pompa ad acqua.

Il segnale di controllo è identificato come PWM (Pulse Width Modulation), una tecnica di modulazione che consente di variare la larghezza dell'impulso con periodo costante; perciò il driver acquisisce questo segnale e genera un segnale d'uscita destinato a regolare la velocità del motore che collegato alla pompa dell'acqua consentirà di irrigare le piante. L'ingresso PWM del regolatore dovrà essere collegato al pin 9 del micro:bit che si occuperà del controllo e quindi della generazione del PWM come vedremo a breve nel setup di sistema. Per completezza e per dare un'idea di quanto appena illustrato, si riporta in **Fig. 9** il Driver di esempio.

DESCRIZIONE SETUP

Dopo aver illustrato le caratteristiche di ogni componente del progetto, prima di passare all'implementazione software su micro:bit e alla presentazione dell'app Android, vediamo brevemente le interconnessioni fisiche presentate a livello logico nel diagramma a blocchi di **Fig. 1** e il setup di sistema tra il micro:bit e le varie periferiche. Riconsiderando il port expander, con riferimento all'immagine di **Fig. 2**, abbiamo scelto di utilizzare la tensione di alimentazione V1, il riferimento G ed i connettori gialli per le periferiche. Partendo dalle considerazioni sulle tensioni di alimentazione è importante impostare la linea V1 su 5V collegando i due jumper verdi come mostrato in figura. Per

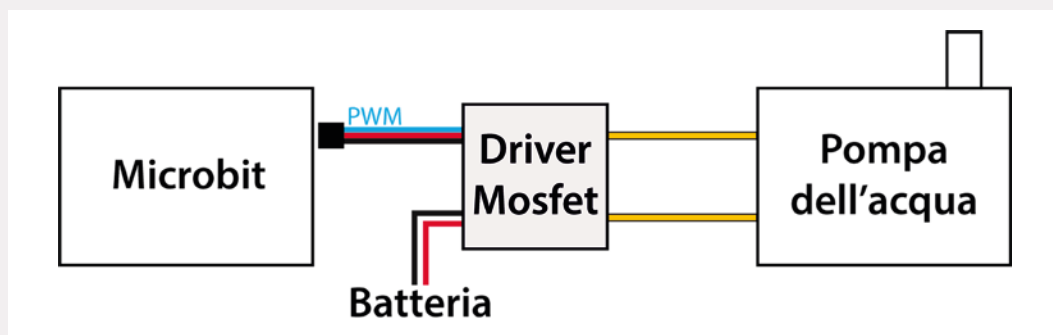


Fig. 8
Schema Controllo Motore Pompa Acqua.

Tabella 1

Connessioni fisiche tra i pin header delle varie schede.

		PIN EXPANDER BOARD MICRO:BIT						
PIN		V1	G	0	1	9	19	20
	Sensore di Temperatura	Vin	GND				SCK	SDI
		CS	SDO					
		SDI						
	Sensore Umidità	Vcc	GND	AO				
	Sensore Solare	Vcc	GND		DO			
	DC Motor	Vcc	GND			PWM		

semplicità i pin d'interesse destinati alle periferiche vengono riportati in **Tabella 1** ed esplicitate le varie connessioni tra schede attraverso i loro pin header, ma vediamo insieme. Partiamo dalla considerazione del sensore di temperatura che richiede un setup dedicato per la selezione e configurazione del protocollo di comunicazione I2C. I pin utilizzati sono:

- **Vin e GND**: che rappresentano alimentazione e zero volt
- **SDO**: collegato al riferimento G per definire l'indirizzo dello slave nella comunicazione I2C
- **CS**: (chip select) collegato a V1 attraverso una resistenza di 1KΩ in configurazione pull-up
- **SCK e SDI**: necessari per la comunicazione (particolare attenzione per SDI che dovrà essere collegato in pull-up con una resistenza di 1KΩ)
- **3V**: non collegato

Riguardo il sensore di umidità, non richiede configurazioni particolari come mostrato in **Fig. 6**, infatti:

- **Vin e GND**: rappresentano le alimentazioni
- **DO**: non verrà utilizzato
- **AO**: deve essere collegato al pin 0 del micro:bit

Così come il sensore umidità, anche quello sensore solare richiede un pin (DO), che dovrà essere collegato al pin 1 del micro:bit con le rispettive alimentazioni Vcc e GND. Infine il Driver per il controllo motore avrà per la parte digitale di pilotaggio motore in PWM il pin 9 del micro:bit e due pin per le alimentazioni come mostrato in **Fig. 8**.

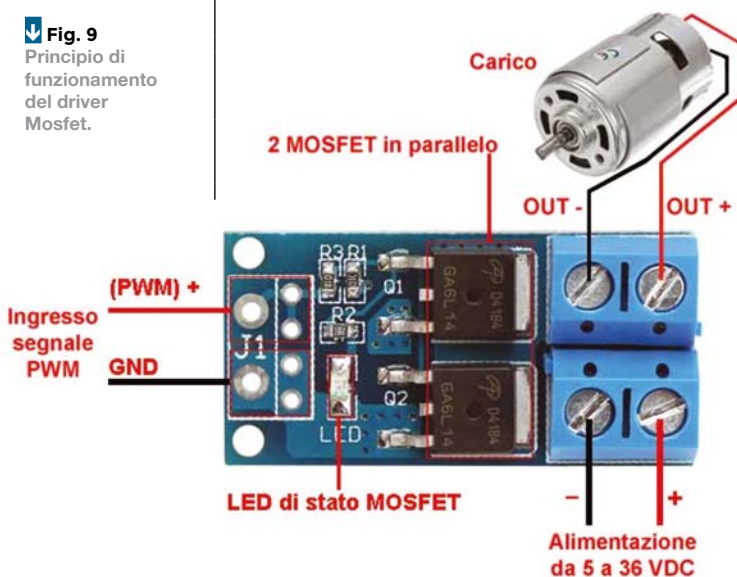
SVILUPPO SOFTWARE

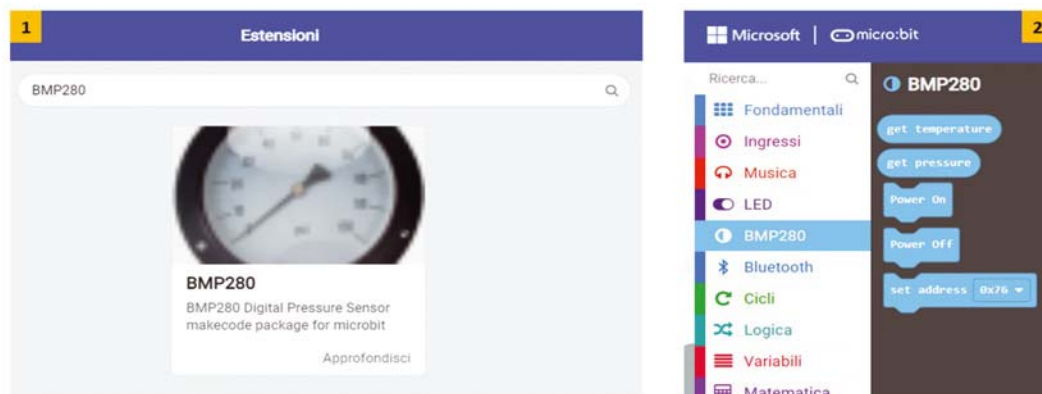
Ora che il nostro setup è completo, possiamo procedere con l'attività di sviluppo software così da "animare" il sistema. Come analizzato in precedenza la funzionalità principale è monitorare lo stato/ambiente in cui si trovano le piante ed eventualmente irrigarle, perciò pensando a questa funzionalità come un algoritmo, abbiamo che le funzionalità principali dovranno essere:

- Lettura temperatura ambiente
- Lettura umidità del terreno
- Lettura irradiazione luminosa
- Pilotaggio motore per la pompa dell'acqua
- Logiche di monitoraggio pianta
- Trasmissione informazioni via Bluetooth Low Energy (BLE) verso APP Android specifica del progetto

Come è immediato immaginare dall'ultimo punto, si avrà anche una funzione in remoto per il monitoraggio e l'irrigazione che vedremo a breve. Per lo sviluppo del software verrà utilizzata la tool chain MakeCode della Microsoft, descritta nel precedente articolo, che tramite una programmazione a blocchi rende la piattaforma accessibile e sem-

Fig. 9
Principio di funzionamento del driver Mosfet.





← Fig. 10
Estensioni
Sviluppo
Software.

plice da utilizzare per tutti. Prima di passare alla scrittura del codice è necessario importare tutte le librerie necessarie e in particolare tramite l'opzione estensioni presente in "Avanzati" è possibile cercare e aggiungere la libreria BMP280 come mostrato in Fig. 10 (a sinistra). Nel pannello delle librerie verranno messe a disposizione tutte le funzioni necessarie per la gestione del componente d'interesse come evidenziato nella Fig. 10 (a destra). Il primo passo da seguire per la scrittura del codice è la creazione e l'inizializzazione delle variabili necessarie per eseguire le funzioni sopra elencate e quindi con riferimento alla Fig. 11, abbiamo pensato di creare le seguenti variabili:

- "EnableBLETx" utilizzata per verificare se il dispositivo è connesso via BLE
- "BMP280_temp" per la lettura della temperatura
- "LandHum" per la lettura dell'umidità terreno
- "Pump_sts" per indicare lo stato della pompa dell'acqua
- "Sun_rad" per la lettura dell'irradiazione solare
- "cmd_motor" per definire la posizione dell'impulso del segnale PPM
- "time_50ms" usata per temporizzare il sistema

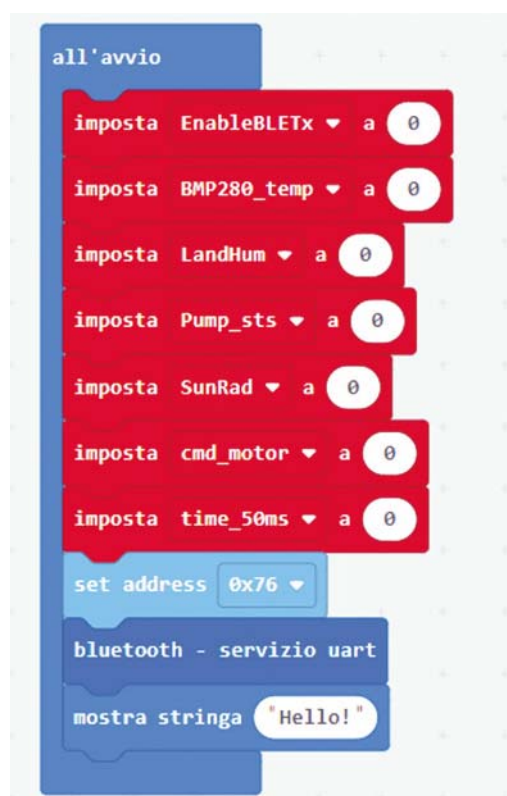
Sempre in Fig. 11, dopo la creazione e l'inizializzazione delle variabili abbiamo impostato il registro indirizzo per il componente BMP280 (0x76 come definito precedentemente), tramite la funzione disponibile nell'apposita libreria.

Successivamente è stato inizializzato il servizio di bluetooth disponibile nella sezione "Bluetooth – Altro" e definita Stringa "Hello" da visualizzare su matrice led disponibile nella sezione "Fondamentali".

A questo punto, se le variabili sono state inizializzate correttamente, si può procedere con la verifica del dispositivo quando viene connesso via bluetooth, perciò dalla sezione Bluetooth si deve selezionare e aggiungere il blocchetto "quan-

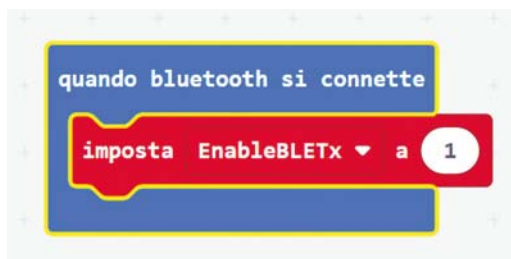
do bluetooth si connette" e infine impostare la variabile "EnableBLETx" al valore 1 utilizzando la sezione "Variabili" come in Fig. 12.

Finita la parte di setup del micro:bit, procediamo con l'implementazione delle varie "routine", cominciando con la lettura dei sensori e l'eventuale pilotaggio della pompa dell'acqua. Per avere una lettura temporizzata abbiamo impostato la variabile "time_50ms" al valore di 50 in modo tale che in tale tempo si può procedere con la gestione delle periferiche. La funzione incremento è disponibile nella sezione Matematica mentre dalla sezione Logica è possibile trascinare ed aggiungere il blocco di condizione "if – else" (se – allora), all'in-



← Fig. 11
Inizializzazio-
ne Variabili.

Fig. 12
Rilevazione Ble.



terno della quale è necessario azzerare la variabile "time_50ms". Con riferimento alla **Fig. 13**, è stata impostata la gestione dei carichi come segue:

- Nella sezione Pin, aggiungere il blocco di scrittura del segnale analogico selezionando il pin P9 del micro:bit e collegare la variabile "cmd_motor" opportunamente scalata (la funzione mappa è disponibile nella sezione Matematica)
- Sempre nella sezione Pin configurare il segnale analogico in scrittura sul pin P9 per la generazione di un segnale periodico con frequenza 1KHz e duty cycle associato alla variabile "cmd_motor" che dipende dalle logiche di controllo.
- Impostare la variabile "LandHum" al valore acquisito dal segnale analogico P0
- Impostare la variabile "SunRad" al valore acquisito dal segnale digitale P1
- Impostare la variabile "BMP280_temp" al valore acquisito tramite la funzione "get temperature" della libreria BMP280

I blocchetti "chiamata plant_logics" e "chiamata BLE_write" sono due funzioni che consentono di definire rispettivamente le logiche di funzionamento del dispositivo e la trasmissione dei dati

via Bluetooth LE, perciò si rimanda alla guida delle librerie per maggiori dettagli.

Dato che non sempre sono presenti nelle librerie tutte le funzioni d'interesse, è possibile crearne di nuove partendo dalla sezione Funzioni che vi consentirà di aggiungere un nome, i parametri e se necessario di un valore di ritorno, tuttavia nel nostro codice le funzioni saranno di tipo void per cui non verrà restituito alcun valore. Con tale funzionalità è stata creata la funzione "plants logics" assegnati tre parametri (identificati come numero); per semplicità di lettura codice faremo riferimento ai nomi assegnate in **Fig. 14**, ovvero: Temperature, Radiation e Humidity, che consentiranno di definire le logiche di funzionamento del dispositivo. La logica implementata prevede che in base ai valori acquisiti dai sensori se i limiti relativi ad umidità del terreno, esposizione solare e temperatura superano determinate soglie, allora è necessario che il comando del driver di pilotaggio pompa d'acqua venga aggiornato insieme allo stato della pompa. Se la funzione è stata realizzata correttamente sarà presente nella sezione Funzioni, pronta per essere selezionata e trascinata all'interno del ciclo d'interesse (si veda **Fig. 14**), per poi assegnare le variabili relative ai sensori acquisiti.

La seconda funzione da realizzare (come appena analizzato per la precedente funzione) riguarda la scrittura dati via Bluetooth che in questo caso richiede quattro parametri come si vede in **Fig. 15**. Anche in quest'occasione conviene assegnare nomi legati alla funzionalità (Temp, Hum, Sun, Pum); successivamente conviene aggiornare lo stato della matrice di Led con "C" se il dispositivo

Fig. 13
Gestione Periferiche.



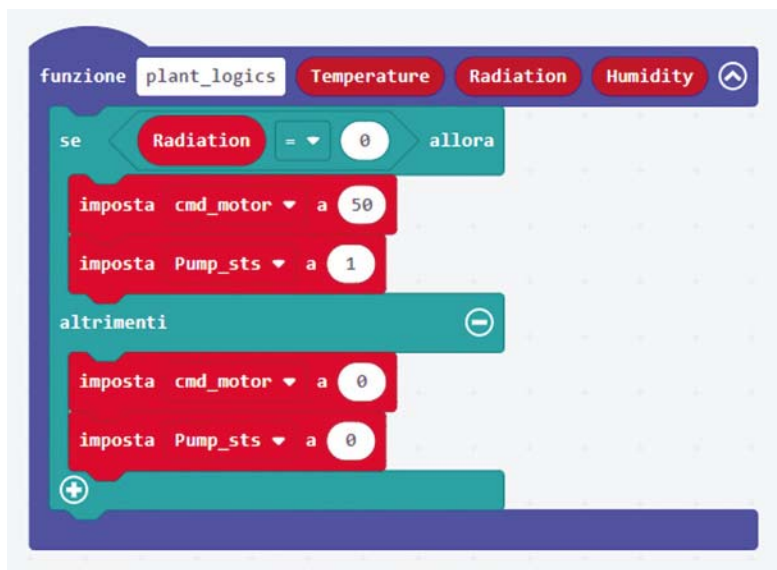


Fig. 14
Logiche di
Controllo.

si è accoppiato correttamente oppure in caso contrario con "D", in base allo stato della variabile "EnableBLETx". Se il pairing (accoppiamento) è andato a buon fine, allora effettuiamo la conversione dei dati in stringhe separate dal carattere underscore ("_").

Se non vengono segnalati errori di coding allora è possibile collegare il micro:bit e procedere con l'aggiornamento del firmware appena realizzato. Per verificarne le funzionalità (lettura periferiche e invio BLE) è possibile utilizzare un'applicazione open source, ovvero Android Serial Bluetooth Terminal scaricabile dalla piattaforma Play Store. Perciò la prima cosa da fare è verificare che il dispositivo micro:bit sia accoppiato (paired) correttamente allo smartphone mediante il Bluetooth per poi avviare l'applicazione Android Serial Bluetooth Terminal. Il risultato è mostrato in Fig. 16:

Per quanto appena illustrato, il micro:bit trasmette via BLE i dati acquisiti dai sensori, perciò considerando la prima sequenza "_28_761_0_1_" il dispositivo legge una temperatura di 28 gradi, l'umidità del terreno è alta, è sottoposto a sorgente luminosa e la pompa è attiva.

APP DEL PROGETTO

L'ultimo passaggio del progetto Smart Monitoring Plant è un'applicazione Android dedicata che consente di guidare l'utente nella gestione delle varie piante configurate e visualizzarne lo stato mediante le informazioni acquisite. Nella Fig. 17, viene mostrata l'interfaccia dell'app che visualizza tutte le piante inserite e configurate dall'utente, centralizzando sul proprio smartphone il control-

lo attraverso un'unica app. Come si può notare nell'esempio di Fig. 17, sono state configurate due tipologie di piante che vengono rilevate mediante la periferica bluetooth low energy (BLE), distinte attraverso il MAC del micro:bit e scambiano tutte le varie informazioni configurate precedentemente. Passiamo ora alla descrizione dell'applicazione ma senza scendere nel dettaglio implementativo perché non scopo di quest'articolo.

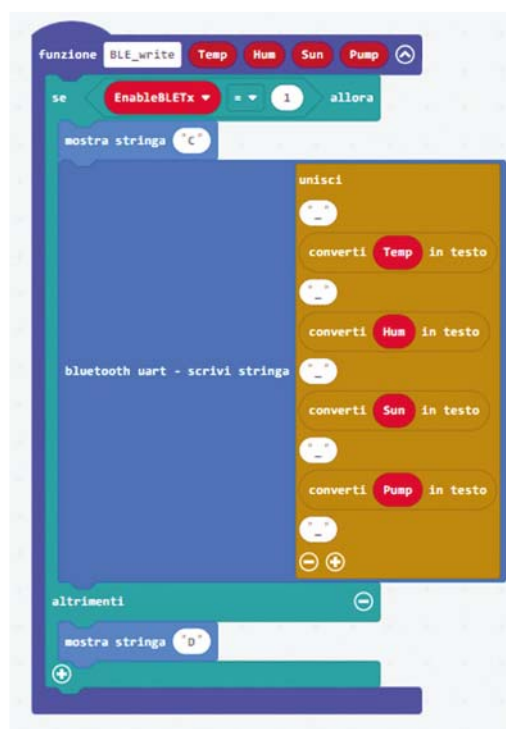


Fig. 15
Trasmissione
Dati via BLE.



Fig. 16
Verifica
Funzionalità.

Avviando l'app viene mostrata la **Fig. 18a**, ovvero una schermata iniziale dalla quale è possibile accedere alla seconda schermata (**Fig. 18b**) che presenta un'area (evidenziata in rosso) dedicata alle schede delle piante. In aggiunta è presente un tasto per rintracciare il dispositivo micro:bit (SMART MONITOR PLANT), un indicatore relativo allo stato della connessione ("Status") e un tasto, identificato come floating button, per aggiungere la scheda pianta ("+"). In questo caso non è presente ancora nessuna pianta configurata perciò si avrà una schermata vuota, ma passiamo all'inserimento di una pianta in particolare.

Per aggiungere una pianta remiamo sul tasto "+" che ci porta alla schermata **Fig. 18c**, in cui vengono richieste tutte una serie di informazioni relative alla propria pianta da inserire, come ad esempio il nome, la tipologia, il terreno (questo permette di fare un controllo più fine dello stato della pianta a partire anche dalla tipologia di terriccio) e l'indirizzo MAC del dispositivo associato. Dopo aver salvato i dati della pianta, questa verrà memorizzata e monitorata, reiterando la procedura per altre piante si arriva ad ottenere quanto mostrato come esempio in **Fig. 18d**. Per completare la procedura

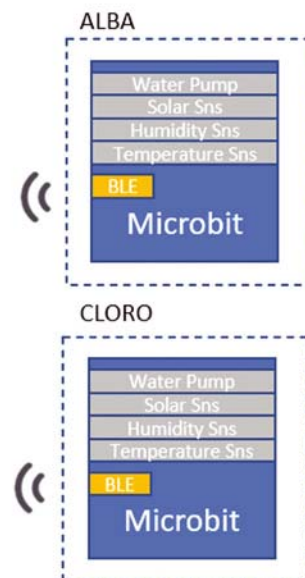


Fig. 17
Sistema Plant
Smart Monitor.

d'inserimento e collegare il particolare sistema PSM alla pianta appena configurata, mediante il tasto "SMART MONITOR PLANT" viene visualizzata un'altra schermata e mostrati tutti i dispositivi bluetooth attivi, tra questi dispositivi va individuato il "BBC Micro:bit" (come da **Fig. 19a**) della particolare pianta d'interesse che con il suo indirizzo MAC risulta allineato a quanto appena inserito nei dettagli della stessa. Quindi cliccando sul nome del dispositivo (BBC Micro:bit), si ritorna alla schermata di **Fig. 19b** e se il dispositivo risulta presente in termini di indirizzo MAC, allora sulla scheda della pianta comparirà il testo "FOUND" e lo stato della connessione aggiornato, altrimenti bisogna rifare la procedura del passo precedente. Fatto ciò, possiamo monitorare la pianta cliccando sulla scheda che riporterà le informazioni relative a temperatura, umidità, radiazione solare, stato

Cosa occorre?

I componenti utilizzati in questo progetto sono disponibili da Futura Elettronica. La board micro:bit V2 è disponibile con lo starterkit (cod. MICROBITQSKIT) a Euro 39,90. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

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

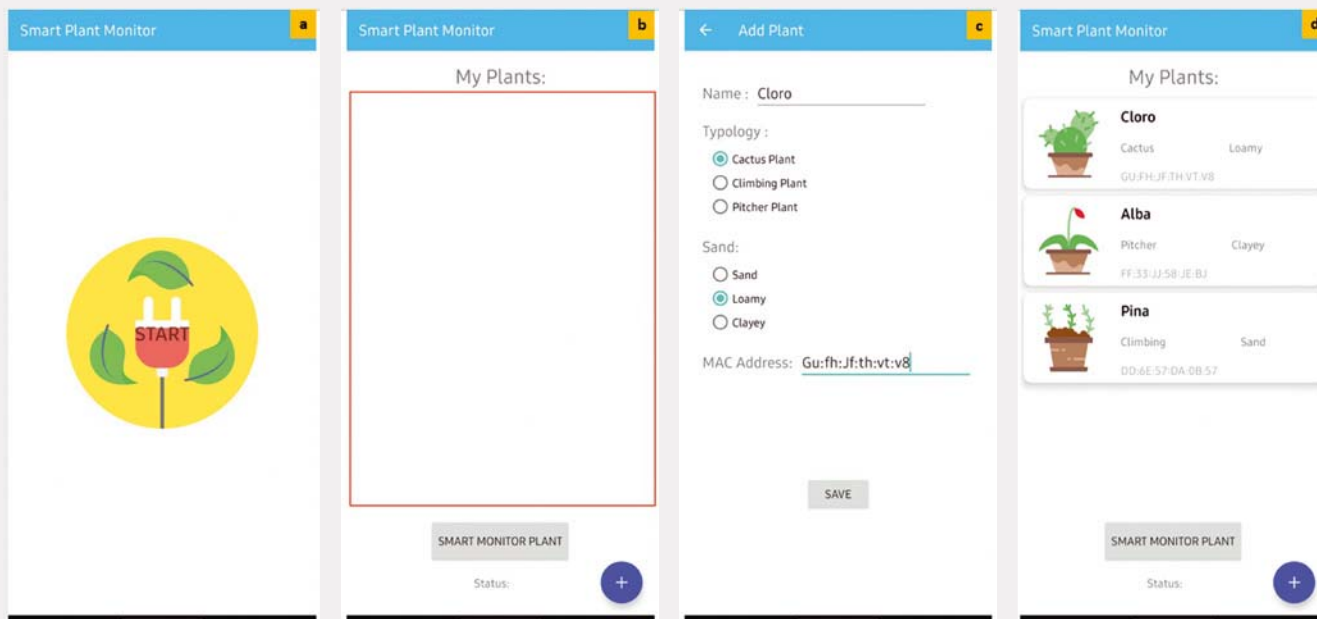


Fig. 18
Le schermate principali dell'APP.

della pompa dell'acqua ed eventuali suggerimenti per l'utente come mostrato nella **Fig. 19c**.

CONCLUSIONI

Arriviamo a termine del progetto "Plant Smart Monitor" illustrato a tutti i livelli, partendo dal punto di vista hardware e quindi nella configurazione del micro:bit, alla scelta dei sensori utili all'acquisizione di temperatura, umidità del suolo e radiazione solare, alla scelta del driver per il pilotaggio della pompa dell'acqua, all'implementazione del software per acquisire le informazioni, definire le logiche di controllo e trasmettere i dati via Bluetooth ed

infine alla configurazione dell'app Android dedicata, grazie alla quale si è in grado di monitorare lo stato della pianta e di informare prontamente l'utente. Questa può essere considerata come una prima implementazione di un sistema intelligente per il monitoraggio piante in completa ottica "green" ma ovviamente è espandibile a tutti i livelli, per esempio considerando di poter fornire all'utente la possibilità di condividere con altri utenti la propria scheda pianta; oppure arricchendo i database con più informazioni in base alle tipologie di piante e terreno utilizzato per rendere ancora più verticale il controllo dello stato del sistema. □

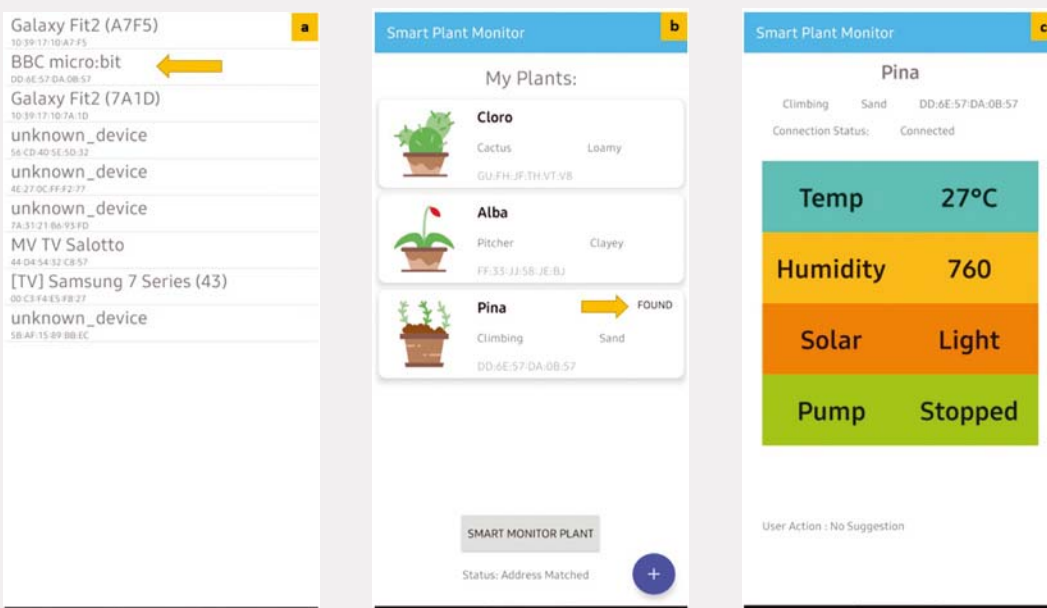


Fig. 19
App Sistema Monitoraggio Intelligente Piante.

NUOVI KIT PER VIDEOGIOCHI ARCADE con Raspberry Pi 4 !



KIT CABINATO ARCADE 1 GIOCATORE CON MONITOR



€ 299,00
Cod. **GAMECABINET**

CONSOLE 2 GIOCATORI PER VIDEOGIOCHI ARCADE



€ 239,00
Cod. **GAMECONSOLE**

Mini postazione per videogiochi Arcade come quelle degli anni '80 e '90! Compatibile con Raspberry Pi 4 tipo B (versione da 2-4-8 GB), utilizza come software RetroPie (aggiornato all'ultima versione), con il quale è possibile emulare vari tipi di console, tra cui: Nintendo, Game Boy, MAME, Sega Master System, Amiga, Commodore, ecc.... Raspberry Pi 4 tipo B e la scheda SD contenente il software RetroPie sono acquistabili separatamente. Grazie alle dimensioni contenute è possibile utilizzarlo ovunque!

Console per videogiochi Arcade con pannello in plexiglass serigrafato. Compatibile con la Raspberry Pi 4 tipo B (Versione da 2-4-8 GB), utilizza come software RetroPie (aggiornato all'ultima versione), con il quale è possibile emulare vari tipi di console, tra cui: Nintendo, Game Boy, MAME, Sega Master System, Amiga, Commodore, ecc.... Raspberry Pi 4 tipo B, la scheda SD contenente il software RetroPie e il monitor sono acquistabili separatamente.

€ 72,00

Cod. **RPI4-4GB**

**RASPBERRY PI 4
TIPO B CON 4GB
DI MEMORIA**



€ 14,90

Cod. **RETROPIESD32GB**

**SDCARD DA 32GB
CON SISTEMA
RETROPIE**



Prezzi IVA inclusa

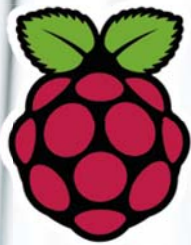
**FUTURA
ELETTRONICA®**

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche
e vendita on-line su:

www.futurashop.it

PARLIAMO CON RASPERRY Pi



dell'Ing. **DANIELE DENARO**

Interagiamo verbalmente con la popolare scheda Raspberry Pi attraverso un microfono e l'Intelligenza Artificiale.

T

ra le possibilità offerte dai dispositivi e sistemi basati sull'Intelligenza Artificiale, c'è sicuramente quella di fornire ai computer o ai robot sofisticate capacità di interazione con l'utente: per esempio, parlare con voce pressoché umana, riconoscere e tradurre in frasi scritte o in comandi le conversazioni con l'utente umano, oppure riconoscerlo dal viso. In questo articolo ci occuperemo della parola e dell'udito, mentre abbiamo già in fase di sviluppo un progetto che con l'Intelligenza Artificiale permetterà di agire con la visione artificiale. Le librerie e gli script Python che abbiamo implementato per l'applicazione qui descritta possono essere usati su qualunque hardware con un sistema operativo Linux o Windows e li potete trovare sul repository on-line https://github.com/open-electronics/Artificial_Intelligence nella sotto cartella TTS-STT.

Fig. 1
Schema generale
del sistema Amazon
Alexa.



DIALOGHIAMO CON RASPBERRY PI

Sicuramente la maggior parte dei lettori ha esperienza di interazione con “Alexa” di Amazon, perché questo prodotto on-line ha ormai preso piede e “vive” in molte case. Per chi non lo sapesse, consiste in alcuni tipi di “ascoltatori” (Amazon Echo) sempre attivi che possono riconoscere comandi e domande, rispondendo quasi sempre in modo corretto e secondo l’aspettativa dell’interlocutore umano. Per esempio, possono attivare dei carichi elettrici, come le lampadine, via WiFi, se sono stati inseriti degli attuatori specifici e compatibili con il sistema Alexa; oppure possono ricercare brani sul Web e riprodurli mediante il loro apparato audio. Naturalmente c’è, per così dire, un trucco: l’apparato intelligente non è nell’apparecchietto che avete a casa, ma sui server Amazon; questo significa che, per funzionare, il sistema ha bisogno del collegamento Internet. L’apparecchio casalingo ha il solo scopo di pulire l’audio del microfono, discriminare il rumore e le pause, e reagire alla parola d’ordine che attiva la sua attenzione. Il resto lo fa il server Amazon su Web. Ovvero: riconoscere il parlato (Speech To Text, STT), spedire la stringa ai server delegati ad esaurire le richieste con una risposta testuale (che può essere accompagnata da un comando codificato), tradurre la risposta testuale in un messaggio vocale (Text To Speech, TTS), comunicare la risposta sonora e l’eventuale comando all’apparecchio Amazon Echo che attiverà l’altoparlante. Lo schema concettuale di funzio-

namento è quello proposto dalla **Fig. 1**.

Abbiamo descritto il sistema Alexa, perché ormai molto comune, ma esistono altri tipi di “Assistenti Vocali”, come : Google Assistant, Cortana di Microsoft e Siri di Apple.

In questo articolo vogliamo spiegarvi come realizzare un’interfaccia vocale con Raspberry Pi utilizzando Python ed allo scopo utilizzeremo un approccio Web per quanto riguarda l’effettivo riconoscimento o la vocalizzazione, imitando Alexa e gli altri assistenti vocali. Il motivo di tale scelta è che purtroppo non sono ancora presenti librerie open-source per l’Italiano. Speriamo, in un prossimo articolo, di poterci svincolare dalla Rete nel momento in cui fosse disponibile un prodotto scaricabile per l’Italiano, disponibile gratuitamente e con buone prestazioni come quelli forniti dai succitati colossi del web e dell’IT.

Non si tratta di replicare Alexa, ma di dotare Raspberry Pi (o altri mini-computer o PC) di una capacità relazionale sofisticata per attivare comandi o partecipare a esperimenti di Intelligenza Artificiale. Quindi il trattamento del dialogo, tradotto in stringhe testuali, è lasciato all’inventiva dell’utente.

RICONOSCIMENTO DEL PARLATO (SPEECH TO TEXT)

Useremo la libreria Python SpeechRecognition <https://pypi.org/project/SpeechRecognition/>; questa libreria è in realtà un *framework* che può utilizzare alcuni motori per il riconoscimento del parlato.

RICONOSCIMENTO DEL PARLATO: UNA BREVE STORIA

Poter azionare comandi tramite la voce può essere molto utile in ambito industriale e addirittura fondamentale in ambito medico nel caso di pazienti paralizzati. Per questo motivo fino dagli inizi dell'era informatica (anni 50-60) si è cercato di affrontare il problema. Purtroppo la capacità umana nel riconoscere il parlato è una di quelle classiche attività per la quale il computer non ha molte chance. Come al solito il computer, inteso come paradigma di ragionamento, è ben diverso dal complesso neuronale cerebrale. Non stupisce, quindi, che gli approcci razionali, modellistici e di linguistica computazionale abbiano dato risultati veramente scarsi. L'approccio iniziale è stato quello di affrontare intanto il riconoscimento di singole parole, nelle due opzioni: parlatore dipendente (più facile) e parlatore indipendente (più difficile). Trasformazione del suono nel dominio della frequenza (Fast Fourier Transform; FFT), destrutturazione in fonemi (suoni elementari che formano la parola), utilizzo di modelli probabilistici (come catene di stati di Markov nascosti; Hidden Markov Model), dizionari della lingua e modelli linguistici, sono i mattoni utilizzati per costruire i primi soddisfacenti esempi di riconoscimento del parlato creati negli anni 80 e 90. IBM, Dragon System e Kurzweil, crearono prodotti commercialmente validi. In ogni caso prevedevano la possibilità di adattare il riconoscimento al parlatore mediante una piccola fase di training; erano quindi parzialmente "speaker dependent". In Italia, abbiamo avuto una importante attività di ricerca fino dagli anni 70, da parte dello CSELT (Centro Studi E Laboratori Telecomunicazioni) di Torino. Questo importante centro di ricerca a livello Europeo nacque negli anni 60 su imitazione dei Bell Labs statunitensi. Lo CSELT fornì alla SIP (poi Telecom Italia e ora TIM) un efficiente sistema di riconoscimento del parlato per il servizio telefonico 1412 (un prodotto che poi prese il nome commerciale di Loquendo). I progressi negli anni 80 e 90 sono da ascrivere sostanzialmente ai notevoli incrementi di capacità computazionali dell'hardware (CPU e DSP "Digital Signal Processor"). Ma i veri

sostanziali progressi degli anni 2000 sono derivati dal cambio di paradigma. Come in altri settori del comportamento umano, i migliori risultati ci sono stati quando si è deciso di imitarlo applicando i concetti di Machine Learning. Ovvero non più programmazione di modelli ma auto addestramento mediante esempi. In sostanza deep-learning nella versione Reti Ricorrenti (RNN). Più precisamente: "Long Short-term Memory Recurrent Neural Networks" (LSTM-RNN). Una rete neurale ricorrente è una rete il cui output è riportato all'ingresso. In questo modo può tenere conto degli eventi passati e quindi essere adatta alla simulazione di serie temporali. Il problema delle RNN è che se si procede troppo all'indietro nel tempo per la correzione dei pesi mediante la discesa del gradiente (Error Back Propagation), l'aggiustamento dei pesi tende a zero rendendo poco significativo il processo di apprendimento. Una rete LSTM tenta di ovviare a questo problema imparando anche a "dimenticare". Un nodo di una rete LSTM è quindi più complesso di quello di una normale rete "Feed Forward" perché accetta anche altri input e pesi relativi al mantenimento dell'informazione nel tempo. I precedenti software IBM, Dragon e Kurzweil sono stati aggiornati con questo approccio ed ora hanno performance di tutto rilievo come Google.

La libreria Google, presentata nel 2015, utilizza LSTM-RNN ed è stata addestrata con numerosissimi esempi sonori adeguatamente supportati da dizionari della lingua ovvero dai modelli linguistici. Con questi innovativi sistemi ora, anche il parlato continuo può essere adeguatamente riconosciuto. Anzi forse meglio, perché i modelli linguistici riescono a riconoscere il contesto e possono disambiguare le singole parole. Poiché il mercato del riconoscimento del parlato è molto importante e promettente, anche in virtù dei recenti progressi, le informazioni tecniche dei prodotti commerciali sono purtroppo fornite con parsimonia; come scarso, almeno per il momento è il software open-source in particolare per le lingue meno universali come l'Italiano.

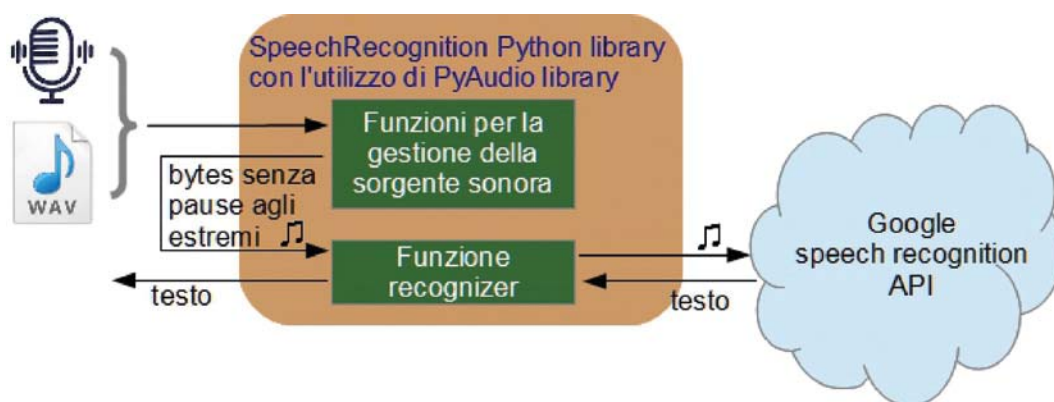
Un paio possono funzionare in locale ("offline"), ma gli altri prevedono di appoggiarsi su API (Application Program Interface) remote. Purtroppo quelli con funzionamento in locale non hanno la lingua Italiana come vocabolario. Per cui utilizzeremo le API di Google che presentano una buona efficienza e semplicità. Permettono utilizzo gratuito e senza registrazione. In pratica il framework SpeechRecognition rende assolutamente trasparente l'utilizzo delle API, per cui non ci si accorge di aver bisogno della rete. Le funzioni appaiono come se fossero totalmente

facenti parte di una libreria locale. Sono selezionabili diverse lingue, ma noi ci focalizzeremo solo su Italiano e Inglese.

In sostanza le funzioni audio, ovvero l'acquisizione dal microfono o da un file multimediale, sono gestite dalla libreria, mentre la funzione che riconosce il parlato maschera il trasferimento dell'audio alle API Google su rete ed il conseguente recupero del testo corrispondente (**Fig. 2**). Non si nota alcun ritardo dovuto al collegamento (a meno che la rete non sia mal funzionante) perché il tempo di decodifica (centesimi di secondo)



Fig. 2
Funzionamento
della libreria
SpeechRecognition.



è sicuramente superiore al tempo di trasmissione. Quindi la funzionalità appare come se fosse locale. Con Raspberry Pi possiamo usare anche il microfono di una comune webcam usb. Per utilizzare la libreria (su microcomputer o PC Windows):

- `pip install SpeechRecognition` (libreria Python);
- `pip install PyAudio` (usato dalla libreria per gestire il microfono);
- Per Raspberry Pi: `sudo apt-get install python-pyaudio python3-pyaudio`;
`sudo apt-get install flac` (coder necessario alla libreria).

Per semplificare l'utilizzo dello "Speech To Text" abbiamo creato un piccolo modulo libreria denominato **SpeechToTextLib.py**; quest'ultimo contiene le seguenti funzioni principali:

- **setLangIt()** e **setLangEn()** per decidere se

- parlare in lingua Italiana oppure in Inglese;
- **listen()**, che ascolta una frase e restituisce il testo ed un codice; l'ascoltatore aspetta un tempo infinito o limitato (parametro della funzione) fino a che non rileva suoni e registra fino ad una pausa maggiore di 0.5s (modificabile); la soglia del suono utile è posizionata a 1000 (modificabile) ed il codice restituito può essere 0 in caso di frase ricevuta, -1 se il tempo è scaduto, -2 se si verificano problemi di trascrizione e -3 se si sono verificati problemi con la rete;
- **startContinuousListen()** che fa partire un ascoltatore continuo che riempie una coda testuale da cui estrarre le frasi (separate dalle pause del parlatore) tramite la funzione **getText()** che svuota la coda dei testi riconosciuti.

La funzione `startContinuousListen()` utilizza due code per caricare i suoni e le frasi testuali, in modo da lavorare in modo asincrono. Le code possono essere normali o circolari. Nel caso di code normali, se si riempiono prima di venire svuotate, si perdono le ultime frasi. Invece se sono rese circolari si perdono le frasi più vecchie, in caso di riempimento. Sono presenti due semplici esempi di utilizzo del modulo libreria:

- `ExampleDialog.py`; ascolta e trascrive in Italiano singole frasi con gestione delle pause o dei silenzi (**Listato1**);
- `ExampleTranscribe.py`; trascrive tutto quello che sente pronunciare in Italiano o in Inglese, in modo asincrono (il codice corrispondente è riportato nel **Listato2**).

Nell'utilizzo di questa libreria, può sorgere il problema della gestione del microfono su Raspberry Pi. Per identificare la periferica ed eventualmente aggiustare il volume di registrazione si possono

Listato 1

```
import SpeechToTextLib as st

print("Ti ascolto")
timeout= 1
while True:
    print("Dimmi...")
    txt, ok = st.listen(timeout= timeout)
    if ok == -1:
        print("Parla, per favore!")
        timeout = 5
        continue
    if ok == -2:
        print("Scusa, non ho capito. Ripeti per favore")
        continue
    if ok == -3:
        print("Errore di sistema (manca connessione?) !")
        break
    print("Hai detto: ",txt)
    if txt.startswith("ascolto stop"):
        break
    timeout= 0.5
```

SINTESI VOCALE: UNA BREVE STORIA

Rispetto al riconoscimento del parlato la sintesi vocale è più semplice e soprattutto si presta molto di più ad un utilizzo classico del computer. In teoria si tratta di scomporre la parola in fonemi e di utilizzare i suoni catalogati per ogni fonema in output. Questo solo in teoria, perché i fonemi vanno anche legati l'uno all'altro e questa giuntura può variare da parola a parola ed anche in funzione della frase (frase propositiva, frase interrogativa, frase esclamativa e via di seguito). Comunque, a causa della teorica semplicità del problema, i tentativi di imitare la voce umana risalgono a prima dell'avvento del computer. Fin dal 1700 (o anche prima) apparati meccanici-sonori mutuati anche dagli strumenti musicali cercavano di modellare la voce umana. Negli anni 30 (del novecento) i Bell Labs crearono il "Vocoder" uno strumento vocalizzante comandato da tastiera. I primi utilizzi del computer risalgono agli anni 60 quando la IBM gestì con il computer dei vocoder. Sembra che Arthur C. Clarke (autore di "2001 Odissea nello spazio") sentì per caso queste prove e inserì la capacità di parola nel computer dell'astronave "HAL9000". Dono della parola che Kubrik inserì nel film omonimo. Va detto che un eloquio semplice e di base non è complicato da ottenere con un computer, a patto di contentarsi di un suono "robotico"; ma già il vocoder IBM aveva un suono molto umano. Questo si può ottenere rendendo fluido lo scorrere dei fonemi; allo scopo occorre analizzare meglio le parole ed il testo nella sua interezza tramite modelli linguistici propri di ogni lingua. In ogni caso esistono due differenti approcci alla sintesi dei fonemi: il campionamento o l'utilizzo dei formanti (detto anche parametrico), ovvero suoni base emessi da un modello acustico dell'apparato

umano. Della sintesi per formanti ricordiamo il famoso "Speak & Spell" (gioco della Texas Instrument) o realizzazioni presenti in alcuni videogiochi degli anni 70-80. Anche in questo settore lo CSELT è stato molto attivo. Già nel '75 creò il prodotto "MUSA" che leggeva qualunque testo Italiano con voce un po' metallica. Lo sviluppo è andato molto avanti fino a confluire nel prodotto "Loquendo" (prodotto industriale per STT e TTS). Anche nel settore sintesi vocale, l'arrivo dell'approccio machine-learning ha avuto un impatto importante anche se non paragonabile a quello prodotto nel riconoscimento vocale. Il risultato è stato un deciso miglioramento della fluidità della frase e del tono, con una qualità del suono in genere indistinguibile da quella del parlatore umano. Le API "gTTS" (2016) di Google sono basate su una rete deep-learning feed-forward di nome "WaveNet" (in realtà creata dalla società Inglese DeepMind). La rete WaveNet è una rete auto-generatrice allenata a produrre un suono corrispondente alla voce umana una volta che è stimolata con alcune caratteristiche del testo. Si tratta di una rete che produce il suono successivo sulla base dei suoni precedenti all'interno di una finestra temporale scorrevole (circa un centinaio di millisecondi). La rete però deve essere stimolata (o condizionata, come si dice nella documentazione) dalle caratteristiche del testo analizzato e scomposto precedentemente. In caso contrario si ottengono semplicemente suoni senza senso che assomigliano a fonemi casuali. La caratteristica di questo approccio alla sintesi vocale di tipo parametrico, è che la rete sa interpolare ed estrapolare tutte le inflessioni della voce dovute per esempio alla respirazione ed altro. Ne risulta un suono veramente naturale.

utilizzare i seguenti comandi, che fanno parte della gestione del suono ALSA su Linux:

- **amixer**, che restituisce l'elenco delle periferiche collegate con il suono;
- **amixer controls**, che restituisce identificatore,

nome ed elenco dei possibili settaggi sugli apparati;

- **amixer cget numid=identificatore**, che restituisce le caratteristiche dell'apparato;
- **amixer cset numid=identificatore valore**, che

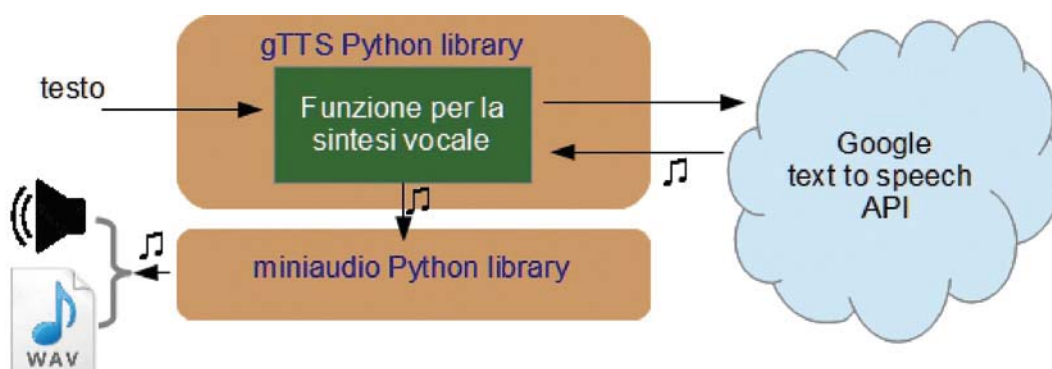


Fig. 3
Funzionamento
della libreria
TTS di Google.

↓ Listato 2

```
"""
Listen (def. in Italian) continuously and transcribe (print) everything
Say "ascolta stop" to stop
"""

import SpeechToTextLib as st
import time

st.setLangIt()
#st.setLangEn()

circque = False
#circque = True

print("Say stop to end")

st.startContinuousListen(circularqueue = circque)
if st.lang == 'it-IT':
    print("Sono in ascolto...")
else:
    print("I'm listening...")

fend = False

while not fend:
    txt = st.getText()
    if txt is None:
        time.sleep(0.2)
        continue
    print(txt)
    if txt.startswith("Ascolta stop"):
        print("Ending...")
        st.stopContinuousListen()
        break
```

setta il valore (esempio: amixer cset numid=2 50% regola al 50% il volume).

Un altro problema è la lunga lista di avvisi che il sistema ALSA produce su Raspberry Pi quando viene usato dal riconoscitore. Questo elenco in realtà non è formato da avvisi di errori, quindi il sistema funziona normalmente. Si tratta di avvisi di configurazione. Possono essere ridotti note-

↓ Listato 3

```
import TextToSpeechLib as ts

ts.setLangIt()
#ts.setLangEn()

print("Just return to end")
while True:
    txt = input("Testo: ")
    if txt == '':
        break
    ts.talk(txt)
```

volmente agendo sul file `/usr/share/alsa/alsa.conf` e provando ad eliminare la serie da `pcm.rear cards.pcm.rear` a `pcm.phoneline cards.pcm.phoneline`, ovviamente dopo aver copiato il file con un altro nome come backup. Infine se fossero visualizzati messaggi sul server "jack" potete farlo partire con il comando: `"jack-control start"`. Oppure semplicemente potete ignorare il fastidio.

FUNZIONALITÀ PARLATORE (TEXT TO SPEECH)

La funzionalità inversa corrisponde a far parlare il computer. Per questo scopo utilizzeremo la libreria Python gTTS. La libreria gTTS (Google Text-to-Speech) <https://pypi.org/project/gTTS/> funziona anch'essa tramite collegamento (trasparente) ad API collocate su rete. Anche questa volta sono selezionabili diverse lingue ma non il genere (maschile o femminile) o il tono (almeno per il momento).

Per l'Italiano è previsto un parlatore maschio, mentre per l'Inglese una voce femminile (chissà perché). Con Raspberry Pi non è possibile utilizzare il desktop remoto VNC perché non instrada l'audio del monitor. Ma si può comunque utilizzare l'uscita audio (abilitandola), oppure utilizzandolo con console-monitor locale.

Per utilizzare la libreria su qualunque computer:

- `pip install gTTS` (libreria Python);
- `pip install miniaudio` (libreria Python per eseguire brani in memoria).

Anche questa volta per semplificare l'utilizzo del "TTS" è stato preparato un modulo libreria: **TextToSpeechLib.py**, il quale contiene le seguenti funzioni principali:

- **setLangIt()** e **setLangEn()** per decidere se usare un parlatore Italiano o Inglese;
- **talk(text)**: la frase testuale viene letta immediatamente dal parlatore;
- **talktofile(text, nomefile)**: il testo anche molto lungo, viene letto e scaricato su un file mp3

Due esempi sono forniti per illustrare la modalità d'uso del modulo-libreria:

- **ExampleSpeech.py**: legge input da tastiera e lo esprime a voce (**Listato3**)
- **ExampleReadingDoc.py**: lo script può partire come programma Python e quindi ha un "help" e la possibilità di selezionare, tramite parametri di comando, il file testuale da declamare, la lingua o la possibilità di declamare mentre legge o di scaricare tutto su un file

TRADUZIONE AUTOMATICA: UNA BREVE STORIA

In seguito ai fenomeni di globalizzazione sia economica che civile, il problema della comprensione dei purtroppo numerosi idiomi parlati nel mondo si fa sempre più pressante. Mentre i linguisti e gli antropologi culturali adorano questa notevole varietà di espressività, lo stesso non si può dire per il comune cittadino che viaggia per affari o per turismo. Ma soprattutto i documenti soffrono di questa difficoltà di comprensione. Nella visione pragmatica dei problemi si è avviato automaticamente cercando di imparare le lingue più utilizzate come lingua franca di scambio internazionale. Nel tempo lo è stato il Latino, lo Spagnolo, il Francese ed ora l'Inglese. Forse nel futuro lo potrebbe diventare il Cinese mandarino. Già negli anni 30 del secolo scorso si pensò a lingue universali come il cosiddetto "Inglese Semplificato" oppure successivamente l'Esperanto. Ma le lingue artificiali pensate a tavolino non hanno mai avuto una chance. Non rimane che la traduzione da una lingua ad un'altra dove fosse necessario. È evidente che con l'avvento dei computer si è pensato di utilizzarli per questo scopo e si è cominciato a parlare di "Traduzione Automatica". Ma, come abbiamo già visto, ogni volta che si tenta di utilizzare il computer per questi scopi così umanamente semplici, fallisce miseramente. Il problema sono: il contesto, l'ambiguità semantica e la complessità sintattica. Non basta, ovviamente, tradurre parola per parola. Il primo approccio è stato quello basato sulle regole (anni 70-80). In questi sistemi il testo, ovvero la frase, viene analizzato e scomposto secondo le strutture proprie della lingua di partenza; poi viene creata una struttura comune (magari con una lingua fittizia intermedia) ed infine viene creata la frase tradotta. Purtroppo in questo processo il contesto rimane abbastanza fuori. Inoltre il corpo delle regole non può essere illimitato e alcune lingue possono essere molto complesse. Un successivo miglioramento è stato l'approccio basato sulla statistica (anni 80-90). Questo sistema utilizza enormi banche dati di testi in formato bilingue in modo da ricavare statisticamente la corrispondenza più opportuna

tra una sequenza di parole e la sua versione tradotta. Candida di IBM è un esempio di questo approccio. Con questa tecnica il contesto comincia ad entrare nel processo di traduzione. Infatti un sistema di questo tipo può prendere in considerazione anche le parole prima e dopo il testo dato, per farle partecipare alla determinazione probabilistica della frase tradotta. Inoltre le banche dati possono essere specializzate nel settore culturale di maggior utilizzo. Questi sistemi hanno costi più alti e si prestano per lo più all'uso in ambito burocratico. Se adesso scommettete che entra in campo, ancora una volta, il deep-learning, avete vinto la facile scommessa. Negli anni 2000, anche questo settore ha avuto un notevole incremento delle sue prestazioni per merito del machine learning. D'altronde l'approccio statistico era già un avvicinamento al mondo delle reti neurali che, per loro natura, estraggono implicitamente regole statistiche dall'addestramento. Le versioni precedenti dei traduttori Microsoft o Google basati sostanzialmente sulle regole, sono state aggiornate con i nuovi "motori" a reti neurali. Infatti Google ha annunciato la nuova versione del suo traduttore nel 2016. Il suo nuovo approccio è Google Neural Machine Translation system (GNMT). Questo sistema utilizza reti ricorrenti (RNN) e più precisamente reti LSTM-RNN (Long Short-term Memory Recurrent Neural Networks). Reti che abbiamo già visto nel caso del riconoscimento del parlato e che entrano in gioco quando si tratta di gestire serie di elementi, in questo caso serie di parole. Il sistema completo è formato da tre componenti. Un componente iniziale chiamato encoder che acquisisce i vocaboli di una frase e li proietta in uno spazio di punti. Un elemento chiamato attention lega questi punti, scorrendo la frase codificata e li passa all'ultimo elemento chiamato decoder che li trasforma negli elementi della frase tradotta. Questo sistema viene addestrato da coppie di frasi con il classico sistema della discesa del gradiente. Si è visto che il sistema sembra crearsi internamente una struttura corrispondente ad una lingua intermedia.

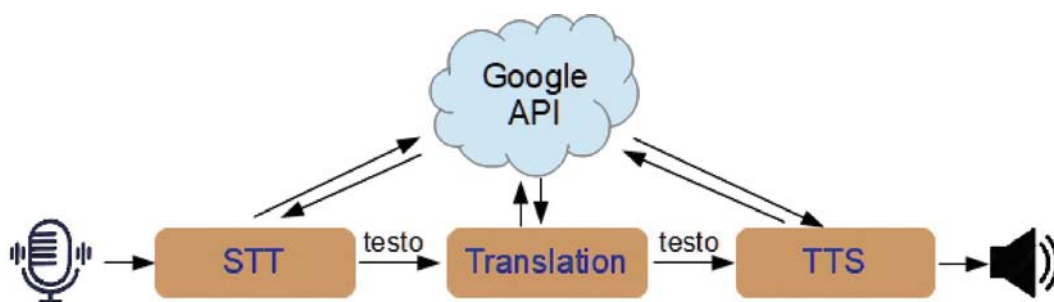
mp3. Se il file testuale non viene fornito come parametro, il programma lancia una finestra di selezione del file sul sistema. Inoltre si può avere la ripetizione su terminale della frase attualmente pronunciata. La frase si intende completata da un segno di punteggiatura di chiusura (. ? !).

usage: ExampleReadingDoc.py [-h] [--filetxt FILETXT] [--filemp3 FILEMP3] [--audio {True,False}]

```
--print {True,False} [--lang {IT,EN}]
Read text doc and speech (default in Italian)
optional arguments:
  -h, --help            show this help message and
exit
  --filetxt FILETXT      File doc (def: filechooser)
  --filemp3 FILEMP3      file output mp3 (def:
TextSpeech.mp3)
  --audio {True,False}  output audio no mp3 (def:
True; out audio)
```



Fig. 4
Traduzione
simultanea.



```
--print {True,False} print each line (def: True)
--lang {IT,EN}         Language (def: IT)
```

IMPLEMENTARE UN INTERPRETE SIMULTANEO

Abbiamo due lingue (ma volendo possiamo utilizzarne molte altre), abbiamo un riconoscitore del parlato ed un sintetizzatore vocale, se ci mettiamo in mezzo un traduttore (di testi) otteniamo un traduttore simultaneo Italiano-Inglese o Inglese-Italiano. Ancora una volta ricorriamo all'aiuto di Google e scarichiamo la libreria Python dal link web <https://pypi.org/project/py-googletrans/> che si installa con il comando:

```
pip install googletrans-3.1.0a0
```

(questa versione corregge un baco precedente).

↓ Listato 4

```
import SpeechToTextLib as st
import TextToSpeechLib as ts
import time
from googletrans import Translator

tr = Translator()

st.setLangIt()
ts.setLangEn()

st.startContinuousListen()

print("Avvio traduzione, parla...")

fend = False
while not fend:
    txt = st.getText()
    if txt is None:
        time.sleep(0.2)
        continue
    tdata = tr.translate(txt, src='it', dest='en')
    ts.talk(tdata.text)
    if txt.startswith("Ascolta stop"):
        print("Ending...")
        st.stopContinuousListen()
        break
```

Questa libreria è semplicissima da utilizzare come potete vedere dal **listato 4**, ed anch'essa lavora tramite la rete. Ma come potrete constatare voi stessi, i tempi di latenza di tutta la catena sono assimilabili a quelli di un traduttore umano. Probabilmente sono dovuti soprattutto al riconoscimento della pausa che chiude la frase pronunciata. Potete provare a ridurla fino a che non vi perdetevi qualche parola. Lo script di esempio ricalca quello che trascriveva tutto quello che sentiva in modo continuativo. La modifica è stata solo quella di inserire la traduzione del testo e di passarla alla funzione vocalizzante, invece di stamparla.

CONCLUSIONI

Bene, avete visto come sia relativamente facile e poco costoso farsi "ascoltare" da un computer appoggiandosi a piattaforme esistenti e utilizzando piccoli script Python. Fatene tesoro! □



Cosa occorre?

Raspberry Pi 4 Tipo B con 4GB di memoria (cod. RPI4-4GB) è disponibile da Futura Elettronica a Euro 69,00, la stessa versione con 8GB di memoria (cod. RPI4-8GB) è in vendita a Euro 99,00. Per Raspberry Pi 4 Tipo B sono disponibili anche diversi starter kit già pronti all'utilizzo disponibili alla pagina bit.ly/RPIStarterKit
I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

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

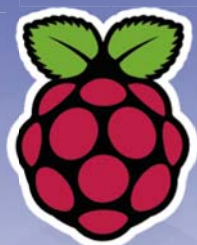
RASPBERRY PI 400!

La tastiera con il computer integrato

€ 146,00

Cod. PI400KIT-ITA

IVA inclusa



**Manuale
in italiano!**

Prezzo IVA inclusa.

Kit completo con computer Raspberry Pi 400 accessori e manuale in Italiano.



Raspberry Pi 400 è un personal computer completo che integra nella tastiera (italiana), una scheda Raspberry Pi 4 da 4Gb con processore quad-core a 64 bit, Wi-Fi, Bluetooth, doppia uscita video (micro HDMI) con una risoluzione fino a 4K, Gigabit Ethernet, USB 2.0 e USB 3.0.

Leggero, portatile, semplice da usare. È ideale per navigare sul web, creare e modificare documenti, guardare video e imparare a programmare utilizzando l'ambiente desktop del sistema operativo Raspberry Pi. I pin GPIO di Raspberry Pi rimangono comunque accessibili, permettendo di connettere componenti e prototipare progetti.

La confezione di Raspberry Pi 400 contiene tutto il necessario (ad eccezione di una TV o di un monitor): computer Raspberry Pi P400, alimentatore da rete, mouse, scheda microSD da 16GB pre-programmata con Raspberry Pi OS, cavo da Micro HDMI a HDMI A lungo 1 metro, un manuale illustrato a colori di 247 pagine completamente in Italiano.

Costruisci e programma un ROBOT!



cod. KSR21
€ 34,90

Robot RICCIO - in kit

Bellissimo robot riccio completo di tutte le parti meccaniche ed elettroniche. È ideale per avvicinare i ragazzi al mondo dell'elettronica e della meccanica. Il robot riccio è dotato di sensore sonoro che gli consente di interagire con il battito delle mani. In funzione del numero di battiti delle mani il robot riccio risponde con una sequenza di movimenti differente. Completo di manuale d'istruzioni in italiano per il montaggio organizzato in step e riccamente illustrato. Funziona con 4 batterie ministilo tipo AAA.



cod. KSR17
€ 29,90

Robot solare e idraulico 12 in 1 - in kit

Avvicinati al mondo dell'elettronica e della robotica imparando e divertendoti! Il Kit permette di realizzare fino a 12 differenti modelli con movimento idraulico e alimentati tramite energia solare. Completo di un manuale ricchissimo di illustrazioni per il montaggio passo passo di una scimmia, un cane e il suo padrone, uno pterosauro, un t-rex, un elefante, un coccodrillo, uno struzzo, uno scorpione, un carrello elevatore, una dune buggy, una scavatrice, un rover e per chi ha fantasia tantissimi altri modelli tutti da inventare.

Robot TOBBIE II con MICRO:BIT in kit

Robot a 6 gambe con corpo libero di ruotare, dotato di tutti i componenti elettronici necessari, tra cui un motore, un sensore a infrarossi e un cicalino 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 tuo PC portatile o smartphone. Funziona con 4 batterie ministilo tipo AAA.



cod. KSR20
€ 89,00



Robot TOBBIE II in kit

Robot in kit a 6 gambe. Può funzionare in due differenti modalità: "Seguimi" e "Esplora". Nella modalità "Seguimi" Tobbie è in grado di seguire qualsiasi cosa gli si avvicini. Quando funziona in modalità "Esplora", grazie al rilevamento automatico tramite sensori infrarossi, è in grado di evitare gli ostacoli che trova davanti a sé trovando sempre un nuovo percorso e continuare a muoversi. Adatto dai 14 anni in su. Funziona con 4 batterie ministilo tipo AAA.

cod. KSR18
€ 37,00



CONOSCERE E USARE

di FRANCESCO FICILI

3

Node-RED

Proseguiamo lo studio del tool di flow-based programming orientato all'IoT ed alla connettività. In questa terza puntata analizziamo la gestione delle periferiche della Raspberry Pi tramite Node-RED, approfittandone per l'implementazione di nuovi esempi pratici. Terza puntata.

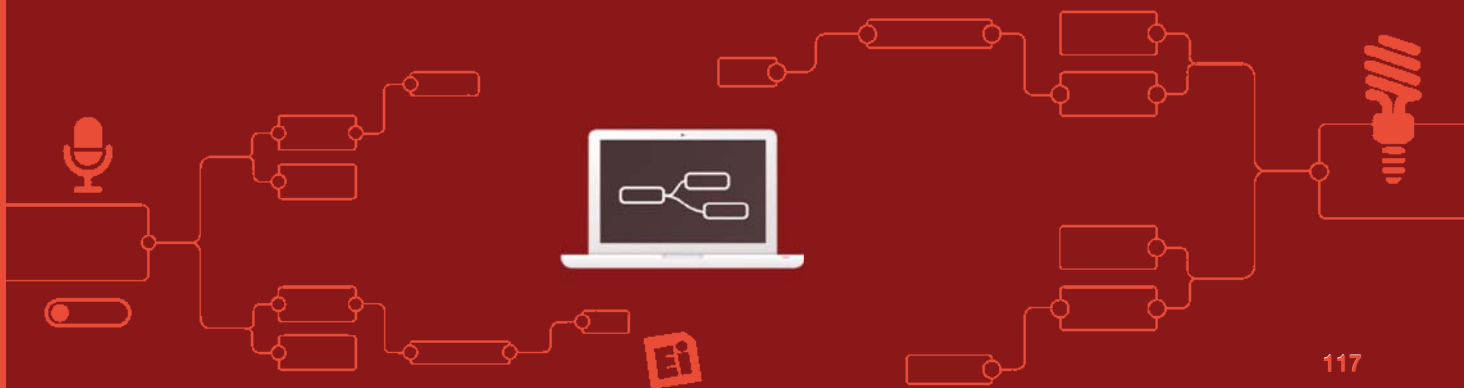
Nelle precedenti puntate abbiamo introdotto i concetti fondamentali di Node-RED, presentando l'ambiente grafico ed i principali nodi predefiniti, implementando anche i primi, semplici, esempi. In questa puntata muoviamo un passo avanti e iniziamo a scoprire come gestire le periferiche della Raspberry Pi tramite Node-RED. Come abbiamo già accennato in precedenza, Node-RED è un ambiente di programmazione che supporta una moltitudine di piattaforme, dal momento che le dipendenze hardware vengono completamente astratte. È chiaro, però, che tali dipendenze esistono e la gestione dell'hardware specifico varia a seconda del target utilizzato (ossia è diverso tra una Raspberry Pi e una BeagleBoard). Tali differenze sono indirizzate in maniera diversa a seconda del nodo utilizzato; in alcuni casi i nodi sono specifici, ossia sono legati alla piattaforma hardware (è il caso di alcuni nodi Raspberry Pi). In

altri casi invece i nodi sono strutturati per supportare diverse piattaforme, e fa parte della configurazione del nodo la selezione della piattaforma specifica. Nelle puntate future del corso, utilizzeremo sempre la Raspberry Pi come piattaforma hardware per i nostri test e quindi faremo spesso uso di nodi specifici.

Tenete comunque conto che tali specificità possono o meno essere portate su altre piattaforme, a seconda delle caratteristiche hardware (ad esempio una semplice interfaccia GPIO è spesso facilmente portabile, mentre interfacce più specifiche lo sono di meno o possono non essere supportate da alcune piattaforme).

PERIFERICHE RASPBERRY PI

La dotazione di periferiche della Raspberry Pi è straordinariamente ricca e questo SBC dispone sia delle periferiche evolute tipiche delle piattaforme ad alta potenza computazionale (Ethernet, USB,



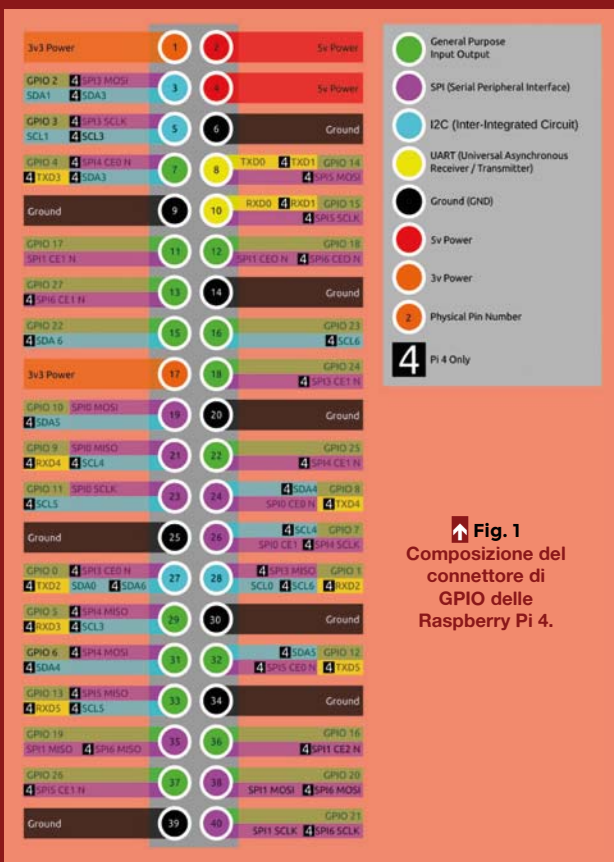


Fig. 1
Composizione del
connettore di
GPIO delle
Raspberry Pi 4.

HDMI, CSI, DSI ecc.), sia delle periferiche di base, più vicine al mondo dei microcontrollori (IO Digitale, PWM, periferiche di comunicazione inter-circuit, eccetera). In particolare, nella versione 4, con il passaggio dal BCM28xx al BCM2711, è stato notevolmente arricchito il set di funzioni alternative presenti sul connettore di GPIO. In particolare, la versione 4 dispone delle seguenti periferiche sul connettore GPIO (ovviamente multiplexate sul totale dei 28 pin disponibili):

- Fino a 28x GPIO Pin
- Fino a 6x UART
- Fino a 6x I2C
- Fino a 5x SPI
- Fino a 2x PWM
- Fino a 3x GPCLK outputs
- 1x SDIO
- 1x DPI (Display Parallel Port)
- 1x PCM
- 1x 1-Wire

In Fig. 1 è riportata la disposizione delle varie funzioni alternative sul connettore GPIO a 40-pin della Raspberry Pi 4 Model B. Come si può facilmente notare le possibilità offerte dalle interfacce presenti sul connettore di GPIO sono notevoli; è possibile infatti controllare I/O digitali, come

pulsanti o relè, interfacciare una moltitudine di sensori e attuatori usando i vari bus di comunicazione disponibili, interfacciare motori o LED tramite i canali PWM e molto altro ancora. Nei prossimi paragrafi vedremo come gestire in Node-RED due gruppi di interfacce, il GPIO standard (ossia IO Digitali e PWM) e alcune periferiche di comunicazione (UART, I²C, 1-wire).

GESTIONE GPIO RASPBERRY PI

La dotazione di base della Raspberry Pi prevede una serie di standard GPIO (28 in totale, corrispondenti al GPIO bank 0 del BCM2711), che possono essere utilizzati come digital input o digital output.

La dinamica del segnale di queste linee digitali è 0-3,3V e ogni linea è dotata di pull-up e pull-down selezionabili via software, quindi non è necessario avere resistenze di pull-up o pull-down esterne per interfacciare pulsanti ed interruttori. Inoltre alcune linee di GPIO possono essere utilizzate come canali PWM (Pulse-Width Modulation), in particolare il BCM2711 dispone di due canali PWM (PWM 0 e 1) multiplexati con i pin GPIO 12 e 13 (o in alternativa GPIO 18 e 19).

In ogni caso tutti i GPIO possono implementare il PWM software e vedremo che questa modalità è ampiamente supportata in Node-RED. Non ci sono canali analogici nativi all'interno del BCM2711 (come anche nei predecessori), di conseguenza per acquisire segnali analogici è necessario interfacciare un convertitore A/D esterno, tramite un apposito bus di comunicazione (tipicamente SPI o I²C). In Node-RED, per l'interazione con i GPIO della Raspberry Pi esistono diverse soluzioni; in particolare riportiamo tre moduli che possono essere installati dalla palette di Node-RED e sono tra i più diffusi.

- **node-red-node-pi-gpio**: questo è un set di nodi che troviamo installato di default su Node-RED quando viene utilizzato lo script di installazione suggerito sul



sito web di Node-RED. Se utilizziamo Raspberry Pi OS, il modulo è già preconfigurato per funzionare correttamente, quindi questa è la via più semplice per poter accedere al GPIO della Raspberry Pi usando Node-RED. Il modulo implementa tutte le funzionalità di base, ossia Digital Input (con gestione pull-up/down e debounce), Digital Output e PWM.

- **node-red-node-pi-gpiod**: questo modulo si basa sul demone PiGPIOd (<http://abyz.me.uk/rpi/pigpio/pigpiod.html>) per interfacciare il GPIO della Raspberry. Di conseguenza, per poter utilizzare i suoi nodi, è prima necessario installare correttamente il demone, seguendo le istruzioni riportate sulla sezione download della pagina web di PiGPIOd (<http://abyz.me.uk/rpi/pigpio/download.html>). Non esiste un vero e proprio installer, ma bisogna scaricare un pacchetto che poi deve essere compilato localmente. Inoltre il demone deve essere preventivamente avviato (tramite riga di comando o allo startup usando rc.local o systemd) affinché i nodi funzionino correttamente. Si tratta di un sistema più complesso rispetto ai nodi predefiniti, ma che offre migliori performance e funzionalità aggiuntive (ad esempio la possibilità di controllare servo tramite le linee di GPIO).
- **node-red-contrib-gpio**: questo modulo permette di accedere al GPIO su una serie di differenti piattaforme (incluse Raspberry Pi, Beagle Bone, Intel Galileo, eccetera).

Nei paragrafi che seguono utilizzeremo i nodi di default (node-red-node-pi-gpio), ma teniamo presente che ci sono anche altre strade che consentono di migliorare le performance ed accedere a funzionalità aggiuntive.

DIGITAL I/O

Come abbiamo già detto il set di nodi node-red-node-pi-gpio è preinstallato nella versione di Node-RED per Raspberry Pi. Questo set è presente nella sezione Raspberry Pi della palette (Fig. 2) e contiene i seguenti nodi:

- **rpi-gpio out**: permette di gestire i pin di GPIO in uscita; i pin possono essere gestiti in modalità digitale o PWM;
- **rpi-gpio in**: permette di gestire i pin di GPIO in ingresso; questo nodo permette anche di controllare i setting delle resistenze di pull-up e pull-down integrate, oltre che impostare il debounce.
- **rpi-mouse**: permette di gestire un mouse USB collegato alla Raspberry (solo gestione pulsanti);
- **rpi-keyboard**: permette di gestire una tastiera USB collegata alla Raspberry Pi.

Per i nostri scopi i nodi di maggiore interesse sono i primi due (GPIO out e in), mentre non ci occuperemo dei nodi mouse e keyboard. Abbiamo già avuto modo di utilizzare il nodo GPIO negli esempi precedenti, senza però entrare nel dettaglio della configurazione ed esaminare le funzionalità

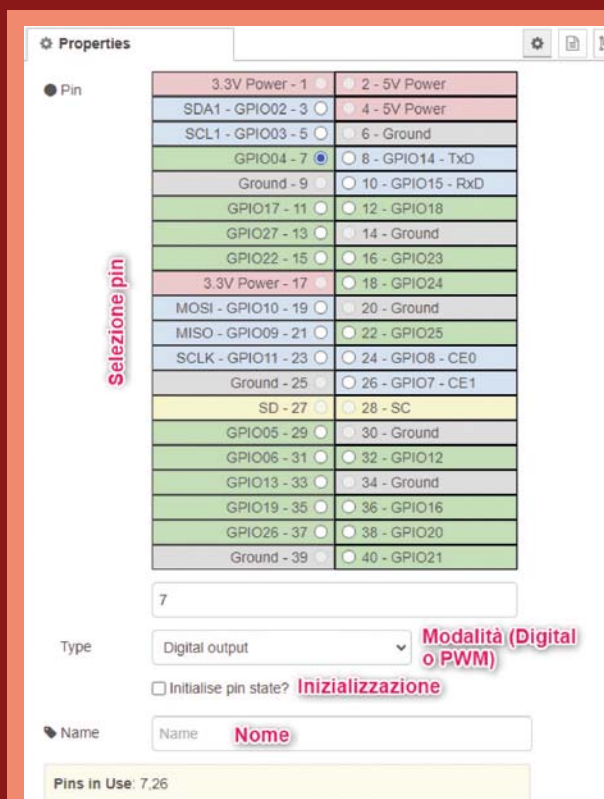


Fig. 3 - Configurazione nodo GPIO out.

aggiuntive. Partiamo quindi dall'analisi di questo nodo, per poi passare al nodo GPIO in.

GPIO out

Per posizionare il nodo sul workspace preleviamolo dalla palette e posizioniamolo su un flow, come di consueto. Come si può notare già dall'icona, il nodo ha un singolo ingresso e nessuna uscita, questo ci fa capire che riceve in input un messaggio (contenente lo stato logico o il valore del PWM duty-cycle con il quale vogliamo controllare l'uscita selezionata) e non restituisce nulla in uscita, in quanto attua direttamente la linea fisica. Cliccando due volte con il tasto sinistro del mouse sul nodo si aprirà la schermata di configurazione, riportata in Fig. 3.

Come si può vedere dalla predetta figura, la configurazione del noto GPIO out è estremamente intuitiva. Sostanzialmente abbiamo una schermata con un controllo principale (Pin) che riproduce la composizione del connettore di GPIO a 40 Pin della Raspberry Pi, dal quale possiamo selezionare il Pin che intendiamo utilizzare. Una volta eseguita questa operazione selezioniamo la modalità del Pin (Digital output o PWM), usando il drop-down menu Type. Infine abbiamo la possibilità di impostare il valore iniziale del Pin (selezionando la checkbox "Initialize pin state") e possiamo anche dare un nome più familiare al nodo (ad esempio

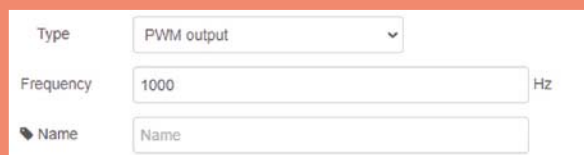


Fig. 4 - Configurazione PWM.



Fig. 5 - Test digital output e PWM.

LED_1). Nel caso in cui si selezioni PWM anziché Digital output come modalità, la schermata cambia leggermente e presenta un nuovo controllo di tipo textbox chiamato Frequency, che serve per impostare la frequenza del PWM in Hz, come si può vedere dalla Fig. 4.

Il controllo di questo nodo è estremamente semplice, in sostanza è sufficiente inviare un messaggio il cui msg.payload contenga lo stato dell'uscita digitale (come boolean o come duty-cycle PWM). Per testare il funzionamento del nodo si possono costruire dei semplici nodi di test, come quelli rappresentati in Fig. 5, che fanno uso del nodo Inject per iniettare i valori digitali e di duty-cycle.

GPIO in

Il nodo GPIO in funziona in maniera del tutto duale rispetto al nodo GPIO out. Prelevandolo dalla palette e posizionandolo sul workspace si nota subito che il nodo ha una singola uscita e nessun ingresso. In maniera simile al nodo GPIO out, l'input di questo nodo è dato dal cambio di stato sulla linea digitale configurata, e il suo output è un messaggio il cui payload contiene il valore digitale che rappresenta lo stato del pin (0 o 1). Come per il nodo GPIO out, anche questo nodo dispone di una finestra delle proprietà (Fig. 6) che può essere aperta tramite doppio click sinistro sull'icona del nodo.

Anche in questo caso abbiamo il selettore Pin che ci permette di selezionare quale GPIO utilizzare dalla lista dei GPIO presenti sul pinout del connettore a 40 pin. I controlli aggiuntivi Resistor e Debounce permettono di configurare

la resistenza di pull-up/down e l'intervallo di debounce (in ms). Questi due controlli risultano molto utili nel caso in cui si debbano interfacciare dei pulsanti. Infine la checkbox "Read initial state of pin on deploy/restart?" permette di effettuare una prima lettura all'avvio del flow, così che lo stato del pin non risulti indeterminato all'inizio. Il solito controllo di tipo textbox "Name" permette di dare un nome più appropriato al nodo (ad esempio BUTTON_1). Se vogliamo testare il funzionamento del nodo è sufficiente (dopo averlo configurato opportunamente) collegare un nodo di debug alla sua uscita, come rappresentato in Fig. 7 e collegare fisicamente un pulsante alla linea digitale selezionata. Con una singola pressione del pulsante il nodo dovrebbe ritornare due messaggi, corrispondenti ai due stati tra cui transita la linea digitale (ad esempio 0 e 1, pulsante premuto e poi rilasciato, se il pulsante si chiude verso massa e viene utilizzata una resistenza di pull-up sulla linea).

Per verificare il funzionamento del GPIO di base, può essere molto pratico utilizzare l'HAT I/O distribuito da Futura Elettronica (www.futurashop.it) con il codice FT1060M. Questo HAT (sostanzialmente uno shield, una cui immagine è visibile in Fig. 8) dispone inoltre di un convertitore AD/DA (PCF8591) I²C, in modo da poter implementare anche alcune funzioni analogiche. Ecco le caratteristiche e la dotazione dell'HAT FT160M:

- 5 LED, collegati ad altrettanti digital output;
- 2 Pulsanti e 1 interruttore, collegati ad altrettanti digital input;



Fig. 6 - Configurazione del nodo GPIO in.

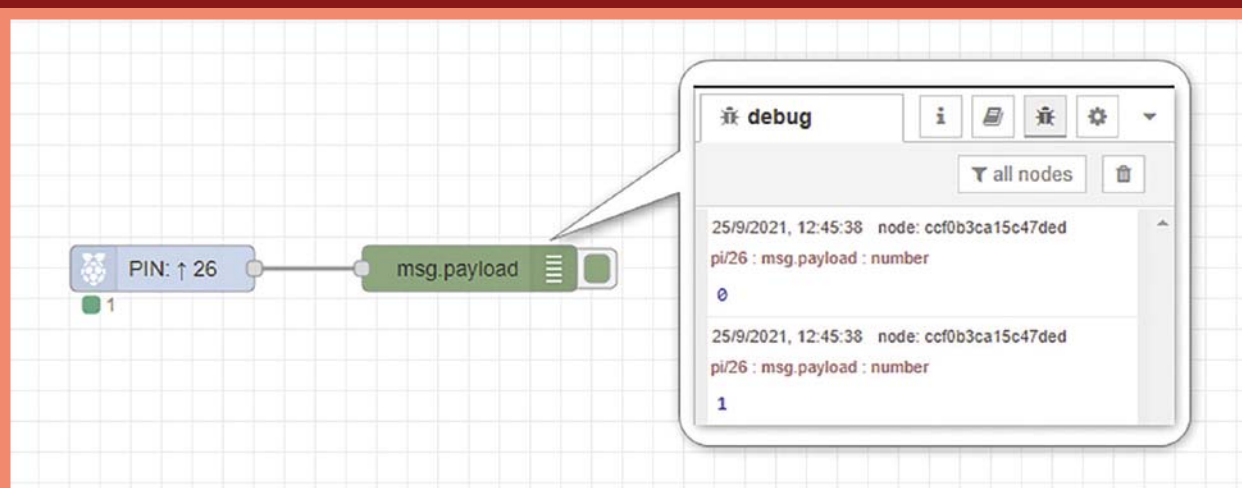


Fig. 7 - Test del nodo GPIO in con visualizzazione della finestra di debug.

- 1 LED, collegato al canale D/A del PCF8591;
- 1 Sensore di temperatura, collegato ad un canale A/D del PCF8591;
- 1 Fotoresistenza, collegata ad un canale A/D dell'integrato di data-acquisition PCF8591 on board.

PERIFERICHE DI COMUNICAZIONE

Dopo aver esaminato in dettaglio i vari nodi per la gestione del GPIO, passiamo adesso all'analisi delle più comuni periferiche della Raspberry Pi che possiamo accedere tramite Node-RED. Come abbiamo visto nella parte introduttiva dell'articolo, la Raspberry Pi dispone di diverse periferiche di comunicazione, che permettono di interfacciare diversi tipi di dispositivi, come ad esempio sensori, attuatori o anche PC o altri SBC.

Per ovvie ragioni di spazio non è possibile effettuare una trattazione esaustiva, quindi ci limiteremo a trattare alcune delle periferiche più comuni e che ci saranno utili per lo sviluppo dei nostri progetti pratici anche nel prosieguo del corso.

Seriale

Cominciamo dalla periferica di comunicazione per eccellenza, ossia la porta seriale o porta UART (Universal Asynchronous Receiver and Transmitter). La UART è una porta di comunicazione asincrona, di tipo full-duplex (ossia è possibile ricevere e trasmettere allo stesso tempo, dal momento che è dotata di due linee di trasmissione, RX e TX), ed opera nativamente a livello TTL (ossia con dinamica 0-5V). La porta seriale viene spesso utilizzata con diversi standard di livello fisico, come RS-232, RS-422 e RS-485, che permettono di ampliare le distanze di comunicazione (ad esempio tramite l'adozione di tecniche differenziali nella trasmissione del segnale) o migliorare l'indirizzamento dei dispositivi. La Raspberry Pi è nativamente dota-

ta di due UART: la mini UART (/dev/ttyS0) e la PL011 UART (/dev/ttyAMA0). Per impostazione predefinita, la mini UART è mappata sui pin GPIO 14 e 15 (rispettivamente TXD ed RXD) mentre la PL011 è dedicata al modulo Bluetooth on-board. Ci serviremo quindi della mini UART come



Fig. 8 - Raspberry Pi I/O HAT FT1060M.



Fig. 9 - Nodi del modulo serial port.

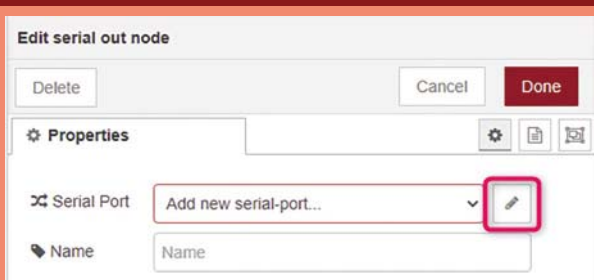


Fig. 10 - Finestra delle proprietà del nodo Serial Out.

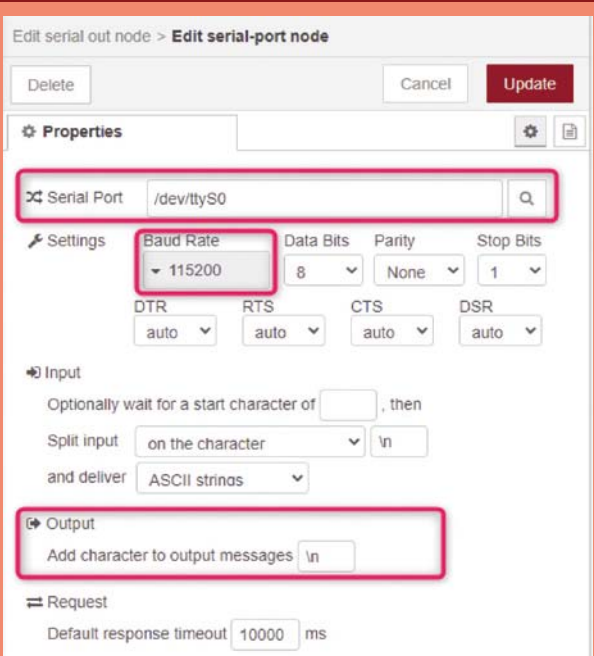


Fig. 11 - Configurazione del nodo serial-port.

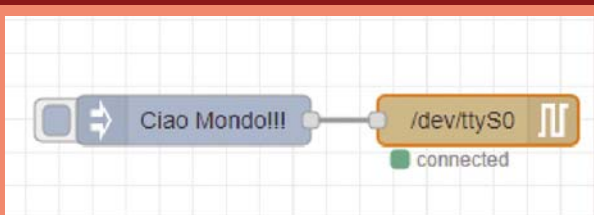


Fig. 12 - Test Seriale.

interfaccia seriale per i nostri test, ma prima di poterla utilizzare con Node-RED occorre disabilitare la console, che per impostazione predefinita è attiva su questa interfaccia di comunicazione e, di conseguenza, ci impedirebbe di utilizzare questo dispositivo da Node-RED. La console sulla mini UART può essere disabilitata utilizzando l'utility

raspi-config (sudo raspi-config) o, più semplicemente, editando il file cmdline.txt.

Per editare il file cmdline.txt, aprirete una shell (ad esempio tramite connessione SSH) e digitate il comando:

```
sudo nano /boot/cmdline.txt
```

Una volta aperto il file con nano editor, rimuovete la seguente sezione:

```
console=serial0,115200
```

ed effettuate un reboot.

```
sudo reboot
```

A questo punto la mini UART è scollegata dal terminale e possiamo utilizzarla. Node-RED offre il modulo serial port per la gestione della seriale (<https://flows.nodered.org/node/node-red-node-serialport>), composto da tre nodi (Fig. 9):

- **Serial In:** permette di ricevere dati dalla porta seriale.
- **Serial Out:** permette di inviare dati alla porta seriale.
- **Serial Request:** permette di inviare un dato e si mette in attesa della risposta (combinazione del nodo serial out – serial in).

A questi si aggiunge il nodo di configurazione serial-port. Il nodo di configurazione serial-port permette di configurare il comportamento della porta seriale, impostando valori come il baud rate, il numero di data bits e stop bits, la parità, ecc. Per testare il funzionamento della porta seriale dobbiamo dotarci di un convertitore USB-Seriale e collegarlo alla Raspberry Pi; Occorre collegare i pin 14 e 15 (con una connessione di tipo crossed), oltre ad un pin di GND. Una volta fatto il collegamento possiamo posizionare il nodo Serial out sul workspace e cliccare due volte sul tasto sinistro per aprirne la finestra delle proprietà (Fig. 10). A questo punto clicchiamo sull'icona a forma di matita a fianco del drop down menu "Serial Port".

Questo permetterà di aprire la schermata di Fig. 11, dalla quale è possibile configurare la porta seriale. Selezioniamo come porta /dev/ttyS0 (è possibile cliccare sul pulsante a forma di lente di ingrandimento per ottenere una lista delle interfacce seriali disponibili), impostiamo il baud-rate a 115200 baud e nella sezione output indichiamo '\n' come carattere da aggiungere ad ogni stringa (in modo da avere un ritorno a capo automatico).

Infine clicchiamo su Update e Done per confermare la configurazione e l'associazione del nodo Serial out con il nodo di configurazione serial-port appena creato.

Ora la nostra porta seriale è configurata, per utilizzarla in trasmissione è sufficiente collegare un nodo inject con la stringa che vogliamo inviare sulla porta seriale. Nell'esempio di Fig. 12 abbiamo inviato la stringa "Ciao Mondo!!!" sulla porta seriale, se apriamo una finestra di terminale su un PC collegato al convertitore USB-seriale che abbiamo

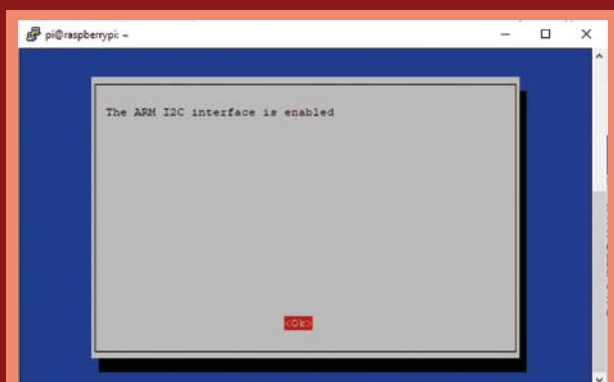


Fig. 13 - Attivazione della periferica I2C su Raspberry Pi.

usato per interfacciare la porta UART della Raspberry Pi dovremmo ricevere la stringa inviata.

I²C

La Raspberry Pi è dotata anche di una periferica I²C, assegnata per impostazione predefinita ai GPIO 2 e 3 (SDA e SCL, rispettivamente). L'I²C è una interfaccia molto utile, infatti si tratta di un protocollo seriale sincrono half-duplex, di tipo master-slave multi-drop. Questo significa che, a differenza della seriale asincrona, può indirizzare diversi dispositivi, agendo proprio come un piccolo bus di comunicazione. Tramite questa interfaccia possiamo collegare alla nostra Raspberry Pi una moltitudine di sensori ed attuatori, controllandoli facilmente tramite Node-RED. Per utilizzare l'I²C la prima operazione da eseguire è quella di abilitare l'interfaccia. Questo può essere fatto semplicemente dall'utility raspi-config (suo raspi-config), selezionando l'opzione "Interface Options" e successivamente l'opzione "I2C". Ci verrà chiesto se vogliamo abilitare la periferica, domanda alla quale dovremo rispondere affermativamente e ad operazione conclusa dovremo vedere un messaggio come quello di Fig. 13 che conferma l'attivazione della periferica I²C (al termine dell'operazione è necessario eseguire un reboot).

Una volta abilitata la periferica dovremo installare un modulo per la gestione dell'interfaccia I²C-Bus sotto Node-RED. Uno dei moduli più versatili, in quanto del tutto generico è il modulo node-red-contrib-i2c (<https://flows.nodered.org/node/node-red-contrib-i2c>), che come di consueto potremo installare tramite l'opzione "Manage palette → Install". Questo modulo contiene tre nodi, come evidenziato in Fig. 14, che sono:

- **I2c Scan:** permette di effettuare una scansione del bus I2C alla ricerca di dispositivi presenti, elencandoli per indirizzo in un array fornito come risposta.
- **I2c In:** permette di effettuare una richiesta di lettura sul bus I2C.
- **I2c Out:** permette di effettuare una richiesta di scrittura sul bus I²C.



Fig. 14 - Nodi per la gestione dell'interfaccia I2C tramite Node-RED.

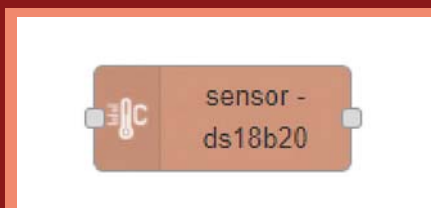


Fig. 15 - Nodo per la gestione dei sensori di temperatura DS18B20.

Oltre a questo set di nodi generico sono presenti diversi altri nodi per la gestione di dispositivi specifici, che semplificano ulteriormente la gestione, sollevando l'utilizzatore anche dall'onere di inviare la sequenza di comandi necessari all'interrogazione di un determinato dispositivo. Tuttavia, data la quantità di dispositivi I²C presenti sul mercato, risulta molto utile avere a disposizione anche questo set generico.

Avremo modo di utilizzare il bus I²C in seguito nel corso delle altre puntate, per adesso è sufficiente sapere come attivarlo e quali sono i nodi per gestirlo.

1-Wire

L'ultima interfaccia che analizziamo in questa puntata è l'interfaccia 1-wire. Come il nome suggerisce, la caratteristica principale di questa semplice interfaccia è quella di consentire la gestione di dispositivi esterni tramite un singolo collegamento (più ovviamente gli estremi della tensione di alimentazione). Sebbene non esistano moltissimi dispositivi compatibili con questo standard, il basso costo, la semplicità di gestione e le esigue risorse necessarie per gestirlo (un singolo pin da sacrificare per la connessione), lo rendono spesso un'ottima scelta implementativa.

Come nel caso precedente, per impostazione predefinita la Raspberry Pi ha un'interfaccia 1-wire disponibile, allocata sul GPIO 4, che va attivata, esattamente come



Fig. 16 - Finestra delle proprietà del nodo sensor-ds18b20.

abbiamo visto fare prima per l'I2C. Lanciamo quindi raspiconfig, e selezioniamo nuovamente "Interface Option", ma stavolta selezioniamo l'opzione "1-Wire" dal menu di gestione delle interfacce, confermando l'abilitazione. Anche in questo caso, al termine della procedura avremo un messaggio di conferma e sarà necessario eseguire un reboot del sistema.

Dal punto di vista di Node-RED, non esiste ad oggi un set di nodi generico per la gestione della 1-wire, ma piuttosto una serie di nodi specifici per i singoli dispositivi.

Ad esempio, uno dei dispositivi 1-wire più diffusi è il sensore di temperatura Dallas Semiconductor DS18B20. Questo dispositivo, incapsulato in un case plastico tipo TO-92 (a tre terminali) è facilmente reperibile, ha un bassissimo costo ed una elevata precisione di misura; inoltre se ne possono collegare diversi sullo stesso bus 1-wire, in quanto ogni dispositivo è identificato da uno specifico serial number univoco. Esistono svariati nodi per la gestione di questo dispositivo in Node-RED; uno molto efficace è quello fornito dal modulo node-red-contrib-sensor-ds18b20 (<https://flows.nodered.org/node/node-red-contrib-sensor-ds18b20>), illustrato in Fig. 15.

Il nodo è di semplicissimo utilizzo, una volta installato e portato sul workspace, facendo doppio click possiamo accedere alla finestra delle proprietà (Fig. 16). Qui verranno già elencati, ordinati per serial number, tutti i sensori

DS18B20 che la scheda Raspberry Pi è riuscita a rilevare (ovviamente dovremo averne collegato almeno uno per vedere qualcosa).

A questo punto il nodo può essere configurato per eseguire letture periodiche ad un certo rate, oppure naturalmente esegue una lettura ogni volta che viene ricevuto un messaggio sulla porta di ingresso. Il risultato della lettura, già convertito in Celsius, verrà inserito nel campo payload del messaggio in output, come di consueto.

ESEMPIO PRATICO: LETTURA DI UN SENSORE DS18B20 TRAMITE INTERFACCIA SERIALE

Arrivati a questo punto, possiamo sfruttare le conoscenze acquisite per implementare una semplice applicazione: ciò che vogliamo realizzare stavolta è un dispositivo che acquisisca una lettura da un sensore di temperatura, nel nostro caso il popolare integrato DS18B20, e la invii ad un terminale seriale in esecuzione su Personal Computer ogni volta che quest'ultima viene richiesta.

Allo scopo utilizzeremo la stringa "Temp?" per identificare una richiesta valida, mentre qualsiasi altra stringa verrà ritenuta non valida.

Per l'implementazione ci occorrono due blocchi serial: serial in (per ricevere la richiesta) e serial out (per fornire la risposta). Inoltre utilizzeremo un nodo function e un nodo switch per verificare che la richiesta ricevuta sia effettivamente formattata correttamente e altri 2 nodi function per formattare le due risposte (positiva con lettura di temperatura e negativa). Infine ci occorre il nodo sensor-ds18b20 visto in precedenza. In Fig. 17 è rappresentato il flow che implementa il nostro semplice programma: analizziamo nodo per nodo il flow al fine di comprenderlo meglio. Il Nodo Serial IN riceve la stringa in ingresso dalla linea seriale. Il configuration node di questo blocco è del tutto simile a quello visto nel paragrafo dedicato alla porta seriale. Il nodo function GetRequest ha il compito di verificare se la richiesta sia stata correttamente formattata: ci aspettiamo di ricevere "Temp?\n", in quanto il con-

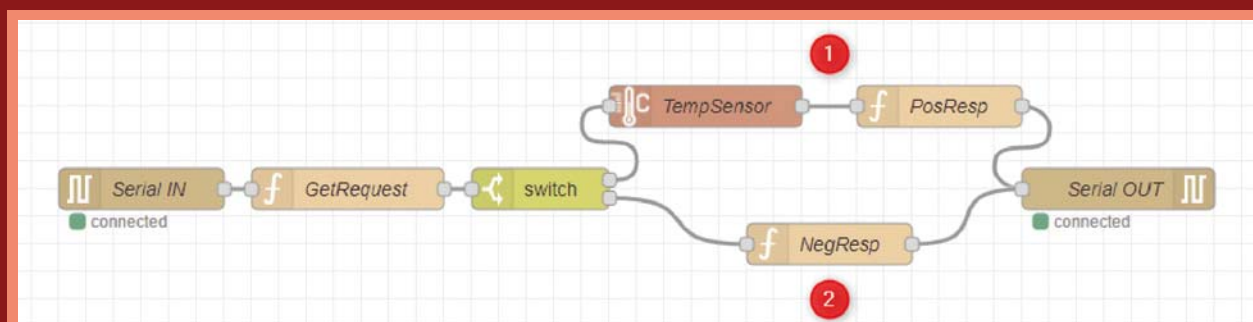


Fig. 17 - Flow dell'esempio pratico.

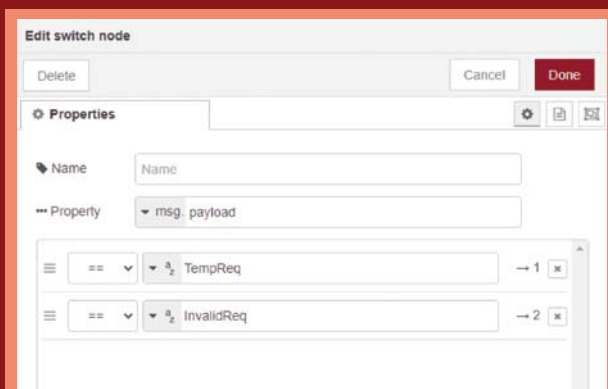


Fig. 18 - Regole del nodo switch.

figuration node della porta seriale si aspetta “\n” come carattere di terminazione di una stringa. Il controllo della formattazione è fatto con il codice javascript illustrato qui di seguito:

```
if (msg.payload == "Temp?\n")
{
    msg.payload = "TempReq";
}
else
{
    msg.payload = "InvalidReq";
}
return msg;
```

che riguarda il blocco GetRequest.

Quindi il nodo valuta la stringa e ritorna msg.payload contenente “TempReq” se la richiesta è corretta o “InvalidReq” altrimenti. A questo punto ci serviamo di un nodo Switch per creare due path alternativi nei due casi. Il nodo switch è stato configurato con le regole visibili in Fig. 18. A questo punto il flow ha una biforcazione: la parte marcata con ‘1’ è il flusso di risposta positivo, quindi qui viene letta la temperatura dal sensore DS18B20 e il nodo function PosResp formatta la risposta tramite il codice javascript:

```
msg.payload = "Temperature: " + msg.payload;
return msg;
```

che riguarda il blocco PosResp.

Siccome non è necessaria alcuna periodicità nella lettura (viene fornita la temperatura solo su richiesta), l'opzione periodic read è stata disabilitata nel nodo sensor-ds18b20. In questa configurazione il nodo fornisce la temperatura solo quando viene triggerato da un messag-

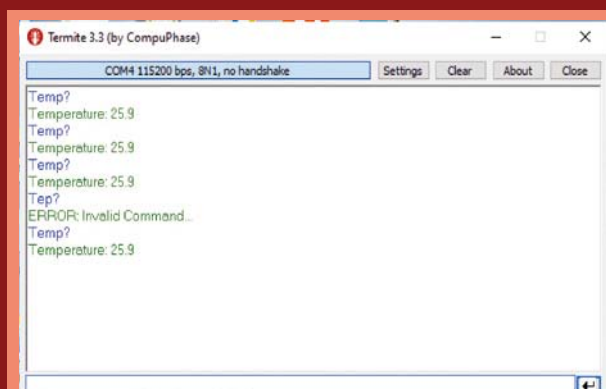


Fig. 19 - Letture di temperatura tramite terminale seriale.

gio in ingresso.

La parte marcata invece con ‘2’ è il flusso di risposta negativo, quindi non viene letto il sensore e il nodo function NegResp ritorna una stringa contenente un messaggio di errore, tramite il seguente codice javascript:

```
msg.payload = "ERROR: Invalid Command...";
return msg;
```

relativo al blocco NegResp.

I flussi si ricongiungono convergendo sul nodo Serial OUT, che invia la risposta sulla linea seriale (il configuration node referenziato è lo stesso del nodo Serial IN).

In Fig. 19 è possibile vedere le schermate relative ad alcune letture effettuate tramite il terminale seriale.

Per la realizzazione dell'esempio abbiamo utilizzato la sonda DS18B20 2846-SONDADS18B20 e il convertitore seriale 2846-CAVOUSBSEERIALE, entrambi distribuiti da Futura Elettronica. Nel collegare il convertitore USB-Seriale si tenga presente che è sufficiente collegare solo i pin TX, RX e GND, se la Raspberry Pi è già alimentata. La sonda DS18B20 va collegata al GPIO 4 ed alimentata tramite uno dei pin a 3,3V della Raspberry Pi. Inoltre sul GPIO 4 va attivata la resistenza di pull-up interna.

CONCLUSIONI

Bene, siamo così arrivati alla conclusione di questa terza puntata, nella quale abbiamo visto come controllare le periferiche della scheda Raspberry Pi tramite Node-RED. Siamo partiti dalla semplice gestione del singolo GPIO per arrivare fino alle periferiche di comunicazione, come UART, I²C e 1-wire, per le quali Node-RED offre diversi moduli di gestione.

Nella prossima puntata introdurremo un importantissimo elemento di Node-RED: si tratta della Dashboard, che permette di creare user interfacce (UI) accessibili via web per implementare i nostri progetti. □



Il film anisotropico dissipa il calore dei transistor pack

La realizzazione di chip ad alta densità di transistor implica il problema della dissipazione del calore. La continua miniaturizzazione nell'elettronica causa la produzione di maggiori quantità di calore a parità di spazio, che va smaltito, altrimenti i chip si guastano. I recenti progressi nella gestione del calore includono i "conduttori termici anisotropi" ossia materiali in cui il calore scorre più rapidamente in una direzione che nelle altre. Molte strutture cristalline naturali sono forti conduttori termici anisotropi: la grafite, ad esempio, dove il calore viene incanalato circa 340 volte più velocemente lungo un asse. Ma tali materiali sono spesso difficili da utilizzare nelle tecniche di produzione su larga scala e possono mancare delle proprietà elettriche o ottiche richieste. Ora gli scienziati hanno creato un materiale artificiale con un rapporto di flusso di calore velocissimo fino a circa 880 a temperatura ambiente; il segreto è nell'utilizzo di materiali costituiti da pellicole sovrapposte di strati atomicamente sottili, di bisolfuro di molibdeno. Gli strati sono tenuti insieme da deboli forze elettriche di van der Waals. Il bisolfuro di molibdeno accumula efficientemente il calore in due dimensioni, ma non nella terza. La chiave dietro l'effetto isolante è il modo in cui i reticoli dei film adiacenti vengono ruotati l'uno rispetto all'altro.

In questi stack, i principali vettori di calore sono i fononi, quasi particelle costituite da vibrazioni nelle strutture reticolari dei cristalli. Quando i film adiacenti di solfuro di molibdeno sono impilati in modo che i loro reticoli siano ruotati tra loro, i fononi fluiscono efficientemente all'interno degli strati. I ricercatori hanno usato queste pile per rivestire elettrodi d'oro alti 15 nm e larghi 100; gli elettrodi potevano trasportare più corrente senza surriscaldarsi e impedivano al calore di raggiungere la superficie del dispositivo.

<https://voices.uchicago.edu>

BMW avvia la produzione della i4

Nel suo storico stabilimento di Monaco di Baviera, BMW ha da poco iniziato la produzione della sua i4, versione full-electric della serie 4 a motore endotermico. La vettura, dall'indiscusso fascino BMW, sarà commercializzata con propulsore elettrico da 340 o da 544 CV, con autonomia, rispettivamente, di 590 e 510 chilometri.

Il produttore tedesco ha investito un totale di 200 milioni di euro nell'integrazione della i4, e quindi delle auto elettriche, nella struttura di produzione esistente.

Entro il 2023, la metà delle auto prodotte a Monaco di Baviera sarà ibrida o elettrica. La catena di montaggio della i4 è peraltro qualcosa di molto particolare, perché per ottimizzare l'impianto, sulla stessa linea potranno essere assemblate sia le versioni elettriche che quelle con motore tradizionale, dato che oltre il 90% degli impianti di produzione possono essere utilizzati

indistintamente; fanno eccezione la produzione del pianale e della parte posteriore delle auto elettriche, che richiedono passaggi aggiuntivi, soprattutto il pianale, che deve ospitare il pacco batteria. Il lancio della BMW i4 è una pietra miliare sulla strada della mobilità elettrica, una delle cui tappe sarà il 2023, quando più della metà di tutti i veicoli dello stabilimento di Monaco sarà a trazione elettrica. Entro il 2030, almeno la metà delle vendite globali del BMW Group dovrebbe riguardare veicoli full-electric. BMW ha fatto di tutto per preparare l'impianto quasi centenario per la produzione di un veicolo completamente elettrico. Soprattutto se si considera che anche lo stabilimento di Monaco deve continuare a costruire veicoli a combustione interna. Accanto alla i4, auto come la BMW Serie 3, la Serie 4 Gran Coupé e persino la BMW M3.

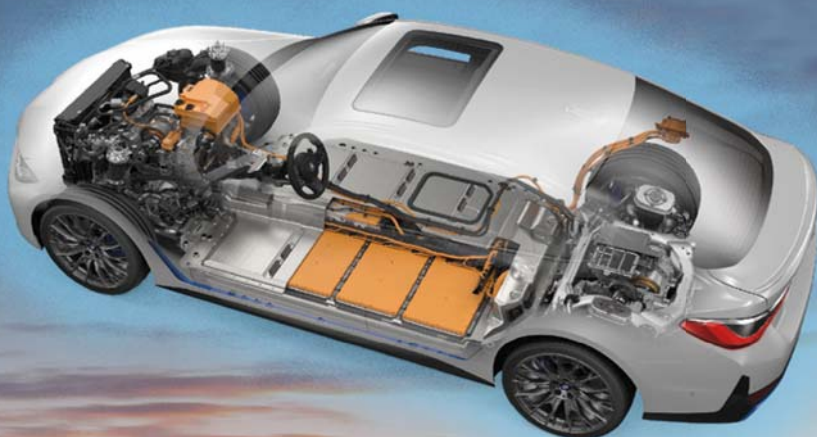
Lo stabilimento BMW di Monaco è an-



che uno degli impianti di produzione più high-tech del settore. L'intera struttura è attualmente sottoposta a scansione 3D (un processo che è stato completato sia per gli stabilimenti di Spartanburg che per quelli di Regensburg). Le scansioni 3D aiuteranno a creare un database digitale per l'impianto che può essere condiviso su cloud da qualsiasi parte del mondo per aiutare con la produzione aggiuntiva. Non è solo un impianto efficiente e ad alta tecnologia, è anche sostenibile. BMW sta ora utilizzando l'osmosi inversa per l'acqua utilizzata in alcuni processi di produzione, come l'immersione catodica, in cui viene applicato il rivestimento di

base ai veicoli, per riutilizzare l'acqua. Si dice che questo riduca il consumo di acqua dolce di oltre sei milioni di litri all'anno. BMW sta anche riducendo le sue emissioni di CO₂, aumentando la quantità di veicoli elettrici a batteria utilizzati nei suoi mezzi di trasporto e nei sistemi ferroviari. L'obiettivo finale è che tutti i veicoli di trasporto che lasciano lo stabilimento di Monaco siano veicoli a emissioni zero. È costato alla BMW 200 milioni di euro rinnovare lo stabilimento di Monaco per la produzione della BMW i4, ma è un investimento utile.

<https://www.press.bmwgroup.com/global>



Le criptovalute quantistiche nel System-on-a-Chip

Toshiba ha sviluppato chip fotonici capaci di comunicazioni quantistiche praticamente inviolabili, in grado di distribuire chiavi quantistiche full-stack. La crittografia dipende da un mittente e un destinatario che condividono una chiave che decifra il messaggio. Con la distribuzione delle chiavi quantistiche, il mittente e il destinatario condividono chiavi segrete casuali codificate in particelle, di solito coppie di fotoni create contemporaneamente. Ogni fotone è collegato con il suo partner, quindi qualsiasi hacker che cercasse di intercettare queste chiavi altererebbe i propri dati in modi che sia il mittente che il destinatario possono rilevare. Il sistema di comunicazione quantistica combina microchip trasmettitori e ricevitori quantistici con un generatore di numeri casuali quantistici basato su chip che funziona a una velocità di 4 miliardi di bit al secondo. A distanze di 10 chilometri, gli scienziati hanno raggiunto tassi di generazione delle chiavi di circa 470.000 bit al secondo in circa 5 giorni, sufficienti per proteggere le videochiamate. A distanze di 50 chilometri, hanno raggiunto velocità di circa 28.000 bit al secondo in circa 12 ore. I chip fotonici sono stati prodotti con processi a semiconduttore convenzionali, il che faciliterebbe la diffusione delle comunicazioni quantistiche; sono anche più piccoli e leggeri delle controparti in fibra ottica tipicamente utilizzate nella distribuzione delle chiavi quantistiche e consumano meno. I chip sono in moduli CFP2, formato diffuso nelle telecomunicazioni ad alta velocità. Il sistema può essere interfacciato con un codificatore commerciale di livello industriale, consentendo il trasferimento sicuro dei dati con un massimo di 100 miliardi di bit al secondo. I fotoni nel sistema vengono rilevati utilizzando rilevatori a fotone singolo, soluzione che ridurrà ulteriormente il costo e l'ingombro complessivo dei sistemi.

<https://www.nature.com>

L'interruttore ottico da 1 trilione di ops

Ricercatori in Russia e all'IBM hanno sviluppato un interruttore ottico capace di 1 trilione di operazioni al secondo. I computer gestiscono i dati sotto forma di livelli di tensione 1 e 0, commutando

dei transistor; nei computer ottici i transistor sono sostituiti da interruttori ottici come quello descritto, basato su un film polimerico semiconduttore organico largo 35 nm inserito tra due specchi altamente riflettenti a formare una cavità microscopica che mantiene la luce all'interno il più a lungo possibile affinché si accoppi con il materiale della cavità. Il dispositivo utilizza due laser: uno a pompa luminosa ed un laser Seed. Quando il primo illumina la cavità, i suoi fotoni possono accoppiarsi con gli eccitoni all'interno del materiale della cavità,



originando quasiparticelle di breve durata note come eccitoni-polaritoni. L'ammasso di eccitoni-polaritoni può formare i condensati di Bose-Einstein, che il laser Seed può commutare fra due stati:

0 e 1. La ricerca potrebbe portare a dispositivi di calcolo ottici in grado di eseguire operazioni mille volte più veloci dei computer a transistor. Questi potrebbero aggirare i colli di bottiglia computazionali nei supercomputer che si basano su un'elaborazione parallela massiva. Oltre agli acceleratori ottici e al calcolo ottico, il nuovo interruttore ottico potrebbe essere applicato come fotorelevatore per scanner LIDAR di droni e veicoli autonomi.

<https://crei.skoltech.ru>

L'Arabia trasformerà la CO2 in fonte energetica

Conosciamo l'Arabia Saudita come produttore di petrolio, ma la vedremo presto impegnata a generare il 50% della sua elettricità da gas naturale e fonti rinnovabili; inoltre, con la Saudi Green Initiative ridurrà le emissioni CO2 tramite progetti di cattura, utilizzo e stoccaggio del carbonio (CCUS) che trasformano l'anidride carbonica in prodotti utili e vendibili.

Ad esempio la creazione di polioli, per la quale, Aramco ha acquisito una tecnologia petrolchimica Converge, in grado di trasformare la CO2 in polioli di qualità industriale. Questi composti chimici possono essere utilizzati per realizzare rivestimenti per elettrodomestici, imballaggi e adesivi, oltre che materie per l'industria automobilistica e medica.

Aramco ha anche lavorato su soluzioni di cattura della CO2 dallo scarico degli autoveicoli, che viene immagazzinata e portata agli impianti di riciclaggio; questo processo è noto come cattura del carbonio mobile (MCC). Anche i settori petrolchimico, della raffinazione, del cemento, dei fertilizzanti e dell'energia potrebbero utilizzare i gas serra riciclati per produrre combustibili sintetici per l'esportazione. Parte di questo processo potrebbe comportare la costruzione di hub in tutto il Regno, in particolare nell'est del paese, ricco di industrie, dove le emissioni possono essere immagazzinate e trasformate in prodotti.

www.saudigreeninitiative.org



Estrazione efficiente e sostenibile del litio

L'incremento nella produzione di auto elettriche aumenterà la richiesta di litio, necessario per le batterie. Per promuovere un'estrazione del litio rispettosa dell'ambiente e delle risorse, la startup statunitense Lilac Solutions ha sviluppato e brevettato una tecnologia di scambio ionico per l'estrazione del litio dalla salamoia geotermica, ovvero depositi naturali di acqua salata. Questa consentirà l'abbattimento dei costi del litio perché ne aumenterà la disponibilità. La ricerca e lo sviluppo della tecnologia Lilac Solutions sono finanziati dal gruppo BMW, SK Materials, Presidio Ventures (una società del gruppo Sumitomo Corporation), MCJ Collective e Earthshot Ventures, che nel complesso portano 150 milioni di dollari. Lilac Solutions ha sviluppato una nuova tecnologia di scambio ionico per rendere l'estrazione del litio dalla salamoia geotermica più efficiente e conveniente. L'azienda cerca di ridurre significativamente l'impatto ambientale rispetto ai metodi convenzionali e, allo stesso tempo, di proteggere le comunità e gli ecosistemi locali. La tecnologia Lilac Solutions è già stata provata nei primi progetti pilota sul campo e deve ora dimostrare la scalabilità e l'industrializzazione a medio termine. Il processo potrebbe poi essere utilizzato in tutto il mondo, anche in salamoie con un basso contenuto di litio.

L'azienda utilizzerà i fondi a disposizione per aumentare la produzione delle esclusive sfere a scambio ionico, espandere i suoi team di ingegneri e operatori sul campo e distribuire la tecnologia a livello globale.

<http://lilacsolutions.com>

MAKER UNO

La scheda **Cytron**
Technologies compatibile
con il mondo Arduino®

Starter kit con Maker Uno
e tutto quello che serve
per iniziare
a programmare
con il linguaggio
Arduino!

+

50
video
tutorial



€ 58,00

cod. MAKERKITBASE

**FUTURA
ELETTRONICA®**

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Prezzo IVA inclusa

Caratteristiche tecniche
e vendita on-line su:

www.futurashop.it

La TV cambia? Non cambiarla!

Arriva la nuova TV digitale, acquista il decoder senza cambiare la TV!

Decoder TV digitale terrestre DVBT, DVBT2, H.265/HEVC



Cod. GS240T2
€ 29,00

**NUOVO
MODELLO**

DVBT/T2
TERRESTRIAL

**H.265
HEVC**

**FULL HD
1080**

USB

HDMI **mpeg** **XVID**
VIDEO VIDEO

GoSat GS240T2 è un decoder digitale terrestre di nuova generazione compatibile con il nuovo standard di trasmissione DVBT, DVBT2 H.265/HEVC. L'apparecchio può ricevere comunque trasmissioni nei vecchi standard DVB-T H.264 e MPEG-4. Il dispositivo offre una perfetta qualità delle immagini in Full HD 1080p. Grazie alla porta USB, disponibile sulla parte frontale, e alla funzione PVR (Personal Video Recorder), permette la registrazione di programmi TV e radio su chiavetta USB o su hard disk esterno. Può essere inoltre connesso a internet tramite il connettore RJ45 per utilizzare servizi come YouTube e guardare canali IPTV (liste con estensioni M3U). Il telecomando in dotazione, oltre ad avere tutte le funzioni per il controllo del ricevitore, ha la funzione di apprendimento e può essere utilizzato anche per altri dispositivi (TV, ecc.). Il pannello frontale ha un display che mostra l'ora o il numero del programma selezionato.



Prezzi IVA inclusa.

Il tuo telecomando TV è rotto?

Sostituiscilo con i telecomandi universali a infrarossi compatibili con le TV delle maggiori marche prodotte dal 2000 ad oggi. Dispongono di tutte le funzioni del telecomando originale. Subito utilizzabili!

€ 5,90
OGNI PEZZO

**PRONTI
ALL'USO!**



Cod.
TVIR-SAMSUNG



Cod.
TVIR-SONY



Cod.
TVIR-PHILIPS



Cod.
TVIR-LG



Cod.
TVIR-PANASONIC

DA PROGRAMMARE

Per comandare fino a due dispositivi audio/video.

€ 9,00
Cod. TVFREEDOM2



DISPONIBILE ANCHE 1 in 1
Cod. TVFREEDOM1

€ 8,50

**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