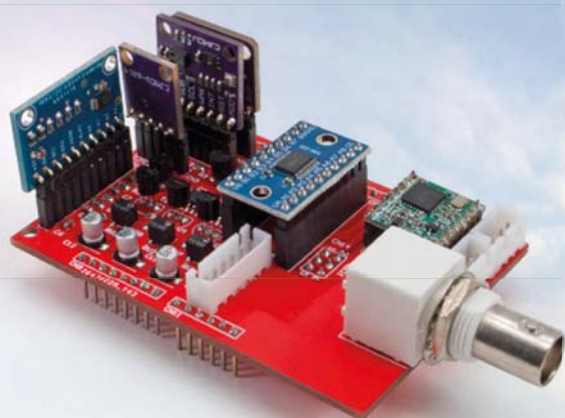


Elettronica In

www.elettronica.in.it

€ 6,00 - Anno XXVIII - n. 261 - FEBBRAIO 2022

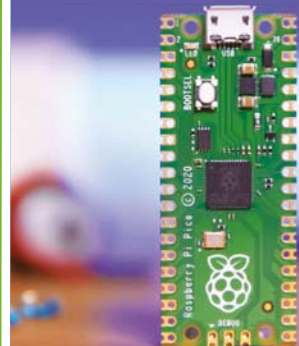
Stazione di monitoraggio qualità dell'aria



- **Gioco a tris con micro:bit**
- **Double Smart Switch**
- **Riconoscimento volti con Raspberry Pi**
- **La stampante 3D sotto controllo**
- **Tracciamento meteo IoT su mappa**



Raspberry Pi PICO



Servizi cloud IoT per makers



CONOSCERE
E USARE



Node-RED

TEST CLIP E ADATTATORI



UTENSILI PER L'ELETTRONICA




minitools

**VERSATILITÀ A SERVIZIO
DELL'ELETTRONICA**

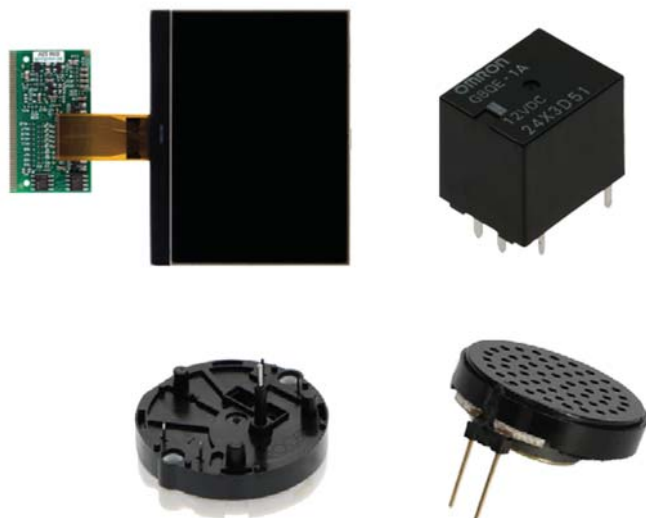
www.minitools.com



BOX MODULARI E CONNETTORI PER DIAGNOSI



COMPONENTI PER L'ELETTRONICA DI BORDO





Cosa ci aspetta nel 2022

Con il primo editoriale dell'anno verifichiamo con voi quelli che saranno i trend tecnologici del 2022. Fa parte del nostro lavoro analizzare il passato e soprattutto guardare al futuro per capire dove l'evoluzione tecnologica ci porterà, così da potervi proporre progetti in linea con le tendenze.

Iniziamo quindi con il *lavoro ibrido* che la pandemia ci ha portato a conoscere molto bene e che in un timido tentativo di tornare alla

normalità coniuga il lavoro in presenza con la possibilità di compiere il proprio dovere da casa.

Questo è possibile anche grazie all'evoluzione della *cybersecurity* che in questo momento di connettività remota diventa fondamentale per garantire la sicurezza dei nostri dati.

Un altro trend in ascesa riguarda le tecnologie smart, dove per smart non si intende semplicemente "connesse", ma bensì controllate da *Intelligenza Artificiale* che sarà sempre più presente e performante anche grazie all'evoluzione della *rete ad alta velocità*, anche mobile, che ha permesso di vedere internet come un sistema al nostro servizio, sempre fruibile e disponibile.

Continua poi ad essere al centro dell'attenzione la ricerca di soluzioni per risolvere il problema del *riscaldamento globale*; siccome una fase principale di questa ricerca passa dal controllo della situazione, in questo numero proponiamo sia una stazione di monitoraggio completa, sia delle stazioni più semplici che possono essere sparse per il territorio per avere una visione più ampia.

Come è intuibile molti trend sono il frutto di un perfezionamento di tendenze che abbiamo conosciuto già negli anni passati, mentre altri assumeranno delle forme totalmente nuove. Non ci rimane altro che seguire attentamente questa evoluzione per proporvi in queste pagine quello che la tecnologia ci suggerirà.

Molti trend sono il frutto di un perfezionamento di tendenze che abbiamo conosciuto già negli anni passati, mentre altri assumeranno delle forme totalmente nuove.

ON LINE

elettronica.in.it

SI RINGRAZIA

Futura Elettronica
Microchip
Mouser
Savix



In copertina:
La stazione di monitoraggio della qualità dell'aria è il nostro progetto di copertina: un insieme di sensori ambientali che ci permette di capire se la zona in cui viviamo è salubre.

Boris Landoni
 boris.landoni@elettronica.in.it



Elettronica In

Rivista mensile, anno XXVIII n. 261

FEBBRAIO 2022

EDITORIALE

Direttore Responsabile

Boris Landoni

boris.landoni@elettronica.in.it

Redazione

Stefano Garavaglia, Paolo Gaspari,
Davide Scullino, Gabriele Daghetta

redazione@elettronica.in.it

Grafica

Alessia Sfulcini

alessia.sfulcini@elettronica.in.it

Ufficio Pubblicità

Monica Premoli (0331-752668)

monica.premoli@elettronica.in.it

Ufficio Abbonamenti

Tel. 0331-752668

abbonati@elettronica.in.it

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.it/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.it/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.

© 2022 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

Wireless

La stampante 3D
sotto controllo

31



19

ARTICOLI

NETWORKING

19 Servizi Cloud IoT per makers

Arduino è uno degli strumenti più adatti per iniziare a sperimentare con l'IoT. Vediamo quali sono i principali servizi cloud per makers per inviare e collezionare i dati acquisiti.

WIRELESS

31 La stampante 3D sotto controllo

Introduciamo un progetto che ci permetterà di controllare la stampante 3D tramite ESP32. Prima puntata.

DIDATTICA

45 Meteo Net Maps

Permette di geolocalizzare in un'unica mappa più dispositivi IoT e di visualizzare i parametri ambientali rilevati da ciascuno di essi.

Networking

Servizi Cloud IoT
per makers



19

Progetto di Copertina

STAZIONE DI MONITORAGGIO QUALITÀ DELL'ARIA

54





45



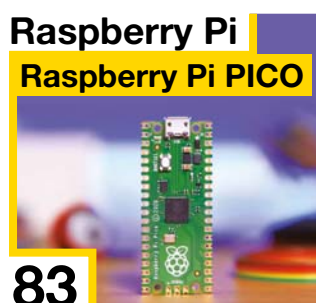
107



93



71



83



117

TOP PROJECT

54 Stazione di monitoraggio qualità dell'aria

Realizzata con l'ausilio di sensori dedicati alla rilevazione di elementi inquinanti e di grandezze meteorologiche, permette un'analisi approfondita dell'inquinamento atmosferico. I valori rilevati vengono trasmessi a distanza permettendone un'accurata valutazione.

NETWORKING

71 Double Smart Switch

Doppio interruttore IoT controllato da smartphone o mediante assistente vocale Amazon Alexa o Google Assistant. Configurabile da browser, è proposto sia con uscita a relé, sia con uscita a SSR.

RASPBERRY PI

83 Raspberry Pi PICO

Pico è il primo microcontrollore prodotto da Raspberry Pi che monta il chip proprietario RP2040, economico ma estremamente efficiente: scopriamo come utilizzarlo.

GADGET

93 Gioco a tris con micro:bit

Riedizione di un classico gioco elettronico realizzato in chiave moderna con la piccola scheda che rende facile imparare a programmare.

RASPBERRY PI

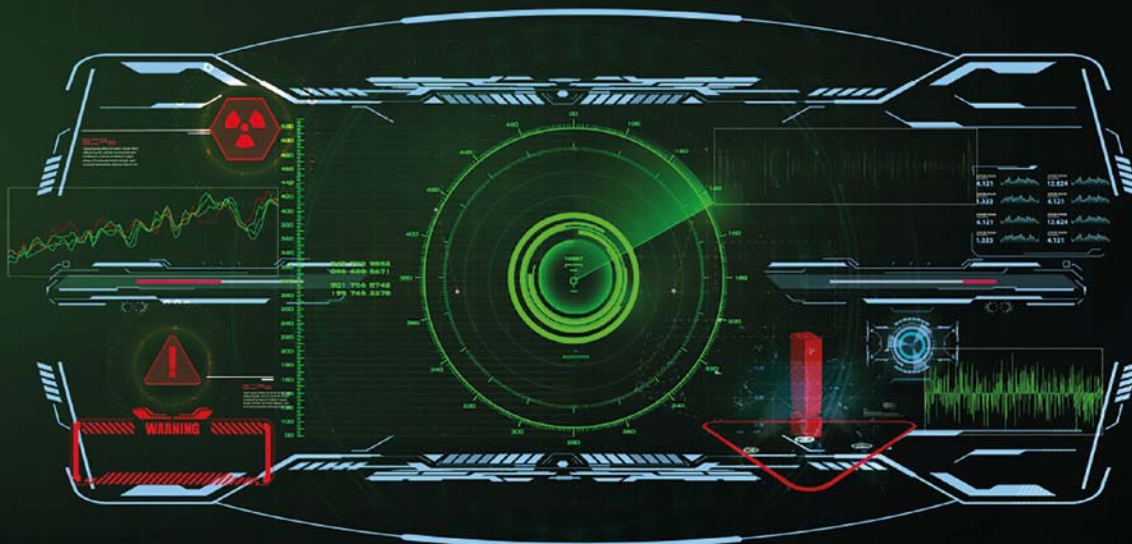
107 Raspberry Pi ci riconosce

Riconosciamo i volti umani utilizzando la nostra scheda Raspberry Pi e una libreria di riconoscimento d'immagine.

DIDATTICA

117 Conoscere e usare Node-RED

Iniziamo a scoprire Node-RED, un tool di flow-based programming, orientato all'IoT ed alla connettività, originariamente sviluppato dall'IBM Emerging Technology Services team e adesso parte della JS Foundation. In questa puntata ci occupiamo dell'implementazione della parte grafica delle nostre applicazioni, sfruttando la Node-RED dashboard, uno dei pacchetti di nodi aggiuntivi maggiormente utilizzati. Quarta Puntata.



Rileva, Avvisa e Rendiconta

Acquisisci e Reagisci agli Eventi di Alimentazione con il Power Monitor Multicanale di Precisione

Sei alle prese con la gestione di un sistema di potenza?

Le famiglie PAC194x e PAC195x, le nostre famiglie di Power Monitor di precisione multicanale di prossima generazione offrono più scelte di architettura e rilevamento avanzato degli eventi di alimentazione e opzioni per requisiti di potenza ridotti.

Rispetto ai monitor singolo canale, queste famiglie di monitoraggio multicanale di corrente/potenza consentono di risparmiare il 31% o più di energia. Sono inoltre disponibili in due gamme di misurazione della tensione: il PAC195x misura fino a 32V mentre il PAC194x apre la porta ad applicazioni a bassa potenza a 9V con 16 bit di risoluzione della precisione sul rilevamento di eventi e monitoraggio.

Vantaggi di rilievo

- Fa meglio dei consumi energetici, più elevati, che si verificano con i monitor di corrente/potenza
- Utilizza avvisi multipli per facilitare il rilevamento flessibile di eventi di alimentazione
- Impiega una bassa impedenza di ingresso per consentire il posizionamento flessibile del resistore di rilevamento
- Aggiungi soluzioni specifiche mirate alle batterie agli ioni di litio a cella singola e doppia



[microchip.com](https://www.microchip.com)



Il nome e logo Microchip e il logo Microchip sono marchi registrati di Microchip Technology Incorporated negli U.S.A. e in altri Stati. Tutti gli altri marchi menzionati appartengono ai rispettivi titolari.
© 2022 Microchip Technology Inc. Tutti i diritti riservati.
DS00004331A, MEC2410A-ITA-01-22

Modem PowerLine 1Mbps

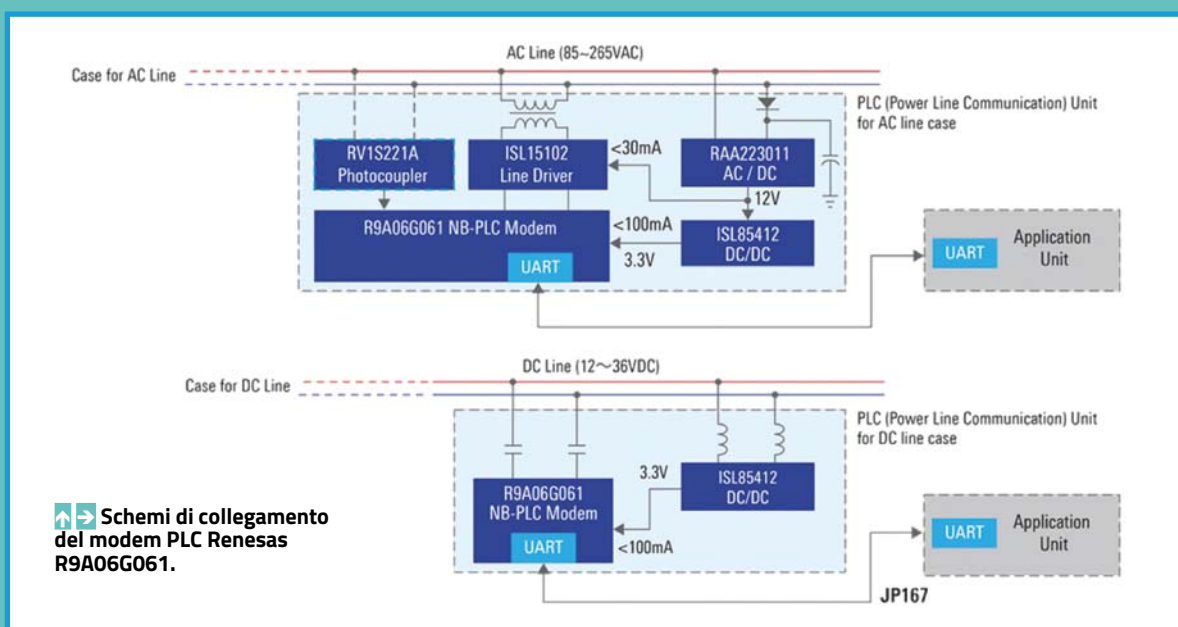
Altrimenti nota come comunicazione dati a onde convogliate, la Power Line Communication (PLC) è una tecnica di comunicazione seriale che utilizza le linee di alimentazione, siano esse in corrente continua o in alternata, come veicolo di propagazione dei segnali (unidirezionali o bidirezionali). Esistono varie soluzioni e protocolli adottati per questo genere di comunicazione, che sfrutta linee esistenti ed ha quindi il pregio di non richiedere cablaggi dedicati, a tutto vantaggio del costo di impianto e della possibilità di sfruttare installazioni esistenti quali, ad esempio, la linea elettrica di distribuzione civile a 230 Vca degli edifici o le linee di alimentazione in c.c. (ad esempio per apparati di automazione, allarme e domotica).

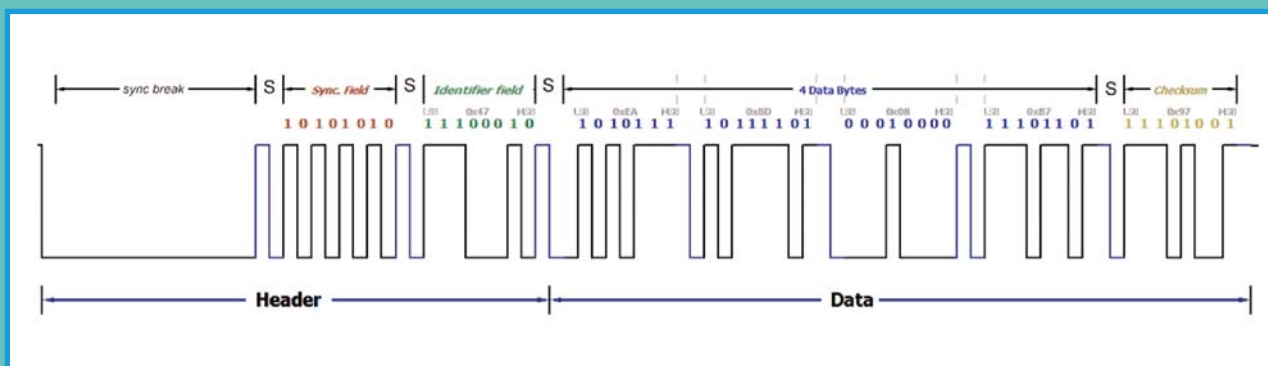
La comunicazione su powerline prevede l'accoppiamento (in trasmissione) e il disaccoppiamento (in ricezione) dei segnali alla linea

elettrica, implementati tipicamente mediante un trasformatore elettrico idoneo a trasferire segnali della frequenza richiesta dal data-rate dei dati, abbinato a un filtro per la reiezione (nel caso delle linee di distribuzione elettrica in alternata) della frequenza di rete. Negli anni i vari produttori di semiconduttori hanno sviluppato

circuiti integrati per implementare vari tipi di comunicazione dati su powerline: da semplici codici di telecomando a vere e proprie comunicazioni secondo protocolli seriali anche standardizzati. Un componente recentemente introdotto è il modem PLC a banda stretta (narrow-band) R9A06G061, prodotto da Renesas, che consente comunicazioni ad alta velocità (data-rate fino a 1 Mbps) stabili

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.





↑ Struttura della stringa dati caratteristica secondo il protocollo LIN.

e a distanza fino ad un chilometro. Le funzioni delle periferiche analogiche integrate nel componente riducono il numero di componenti esterni richiesti, consentendo di realizzare un modem powerline in pochi cm quadrati di PCB. Grazie a queste caratteristiche, l'R9A06G061 è ideale per il controllo di riscaldamento, ventilazione e condizionamento dell'aria (HVAC) e per il controllo dei sistemi di illuminazione negli edifici, oppure per il monitoraggio delle stringhe di pannelli, degli inverter e degli altri apparati degli impianti fotovoltaici, sfruttando, per la comunicazione, la linea in continua delle stringhe o quella in alternata dell'inverter.

La modalità di comunicazione peer-to-peer (P2P) su reti con topologia a bus o a stella, consente di raddoppiare il numero di dispositivi che possono essere collegati, arrivando quindi a più di 200 unità senza l'adozione di ulteriori amplificatori di linea. Inoltre l'elevata tolleranza al rumore rende l'R9A06G061 adatto ad applicazioni industriali.

Il Bus LIN

Acronimo di Local Interconnection Network, il LIN è un bus esterno a bassa velocità utilizzato per connessioni locali di sensori e attuatori ai sottosistemi di bordo dei moderni autoveicoli, ma anche alla ECU motore o al body computer; per esempio il LIN si utilizza per interfacciare la ECU con il modulo regolatore di carica dell'alternatore, ma anche i dispositivi di controllo degli automatismi contenuti all'interno di una portiera, come alzacristalli, chiusura centralizzata, regolazione specchietto laterale, ma anche i sensori di livelli dei fluidi e delle temperature del motore. Il bus LIN è gestito da un controller LIN, che da un lato è interfacciato alla centralina gerarchicamente superio-

re (con un bus più veloce, come ad esempio il CAN) e dall'altro ai sensori/attuatori, che rappresentano le parti più periferiche dell'elettronica di bordo. La comunicazione avviene su un solo filo rispetto a massa, come nell'onewire; un dispositivo Master gestisce la comunicazione sul bus, e deve avere una temporizzazione precisa, mentre i dispositivi Slave si sincronizzano mediante l'intestazione (header) dei messaggi inviati dal Master. La comunicazione è quindi asincrona. Il Master deve avere una temporizzazione precisa, mentre per gli Slave è ammessa una tolleranza del clock del 14%. La velocità di comunicazione (data-rate o bit-rate che dir si voglia) sul bus LIN non supera i 19,2 kbps, il che limita il massimo numero di dispositivi gestibili, almeno se si desidera una risposta in tempo breve, tuttavia il LIN nasce per l'interrogazione di sensori e il comando di attuatori che non richiedono tempi di reazione particolarmente brevi né, peraltro, ridondanza e tolleranza dei guasti. Nel LIN, ogni dispositivo (Master o Slave) connesso al bus è chiamato nodo e il complesso dei nodi su un bus prende il nome di Cluster; ogni cluster può essere formato da un massimo di 16 nodi e la lunghezza del bus non deve superare i 40 metri.

Per quanto riguarda il link layer, il bus LIN ha molte caratteristiche in comune con il CAN, in particolare la modalità di comunicazione che è di tipo broadcast, quindi ogni messaggio inviato viene ricevuto da tutti i nodi della rete; questo mette al riparo da collisioni e non richiede la sincronizzazione dei nodi nel Cluster. Ogni messaggio si definisce frame ed ha un proprio identificativo (PID); sono previsti 64 PID (indirizzi da 0x00 a 0x3F) di cui gli ultimi quattro riservati a comandi specifici del protocollo. La tipica connessione elettrica del LIN è un

connettore a due poli, dei quali spesso è collegato solo quello di segnale, perché come massa viene utilizzata quella dell'involucro del sensore o attuatore da leggere; almeno quando il tipo di segnale lo permette perché non soffre del cammino di corrente sulle masse dell'automobile nel punto dove si va a effettuare la connessione LIN. Il connettore può anche avere tre poli. Il segnale è tipicamente un'onda rettangolare unidirezionale dell'ampiezza di 12V e a vuoto, ossia senza collegamento alle unità Master, sulle Slave si rileva una frequenza dell'ordine di 4,8 kHz (assenza di dati). Il bus LIN è basato su un'interfaccia UART a 8 bit e l'architettura è definita Single Master/Multi Slave; l'aggiunta o la rimozione di un nodo interessa soltanto il Master, mentre è trasparente agli altri Slave.

Sample & Hold

Un elemento fondamentale dei convertitori di tipo analogico/digitale è lo stadio di ingresso, che integra il circuito campionatore, anche noto come Sample & Hold; l'utilità di quest'ultimo si può comprendere considerando che ogni A/D Converter (ADC) può rappresentare in forma digitale, espresso su un certo numero di bit di uscita, il valore della tensione all'ingresso, ma deve farlo per un determinato campione temporale, che sia il più breve possibile, quantomeno se si tratta di convertire segnali variabili. Se durante la conversione da analogico a digitale il segnale varia, il risultato fornito non è preciso, non a caso per ogni ADC è stabilita una massima frequenza del segnale applicabile all'ingresso.

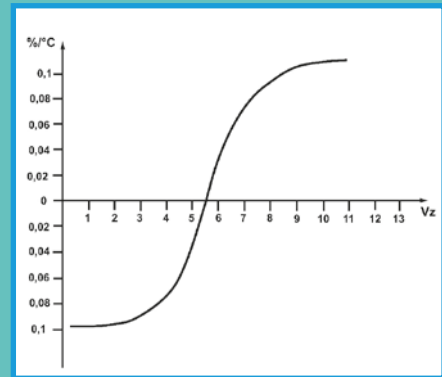
Il campionamento del segnale, ossia l'estrazione di un campione entro un intervallo di tempo tanto piccolo che la sua ampiezza non cambi apprez-

zabilmente del segnale analogico da convertire, avviene tramite un circuito campionatore che è appunto il Sample & Hold (spesso abbreviato in S&H) o campionatore a memoria, che abbiamo schematizzato nella figura in questa pagina, cui ci riferiamo per spiegarne il funzionamento; il circuito è formato da due coppie di switch CMOS e funziona in due tempi: nella prima fase gli interruttori verso l'ingresso sono chiusi e fanno caricare il condensatore con il segnale d'ingresso (gli interruttori dal lato dell'A/D Converter sono aperti). Nella seconda fase, si aprono gli interruttori dal lato dell'ingresso e la tensione sul condensatore giunge al convertitore A/D mediante gli interruttori di destra. In realtà questo circuito viene usato quando c'è la necessità di separare elettricamente la tensione da campionare dall'ingresso dell'ADC; per campionare il segnale come è richiesto dal tipico convertitore A/D a retroazione, basterebbe una coppia di interruttori per applicare a tratti il segnale d'ingresso. In quest'ultimo caso gli interruttori dovrebbero essere sincronizzati con il reset del contatore, in modo da chiudersi sul segnale dopo che il contatore è stato resettato ed è pronto a iniziare un nuovo ciclo di conversione analogico/digitale.

Il diodo Zener

Quando parliamo di diodo Zener intendiamo un componente elettronico a giunzione di semiconduttore, nello specifico un diodo a giunzione PN, la cui particolarità è di condurre in entrambe le direzioni ma a condizioni differenti: se polarizzato direttamente conduce oltre la tensione di soglia (quindi superando i classici 0,6 volt nelle giunzioni in silicio),

mentre se viene polarizzato inversamente, rimane idealmente interdetto (vi scorre solo la modesta corrente di saturazione inversa, dell'ordine dei nanoampere, almeno nei diodi al silicio...) e inizia a condurre come fosse in polarizzazione diretta (facendosi quindi attraversare da una corrente comparabile con quella diretta) quando viene superata la sua "tensione di Zener". La tensione ai suoi capi rimane praticamente costante per qualsiasi valore di corrente, almeno in teoria. Questa caratteristica rende lo Zener ideale per realizzare circuiti stabilizzatori di tensione, alimentatori stabilizzati e comunque dove serva un riferimento costante di tensione. Il tipico circuito di polarizzazione usato per ottenere una tensione stabilizzata è composto dal diodo Zener con in serie all'alimentazione di partenza una resistenza di limitazione (zavorra). La regolazione della tensione deriva dal fatto che, entro certi margini, qualsiasi variazione della corrente nel diodo derivante da un cambiamento del valore della tensione di alimentazione del bipolo Zener/resistenza zavorra o da una variazione di assorbimento del carico, lascia praticamente inalterata la V_z (tensione di Zener). Questo parametro può essere considerato una tensione di breakdown della giunzione PN, anche se tale termine è improprio e ne spieghiamo il perché: nei diodi con tensione di Zener fino a 5,6 volt, il funzionamento sopra descritto nasce proprio dall'Effetto Zener, che consiste nella rottura dei legami tra gli elettroni di drogante e gli atomi associati, sotto l'effetto del campo elettrico operato dalla polarizzazione inversa; oltre i 5,6V la conduzione inversa del diodo si deve prevalentemente all'effe-

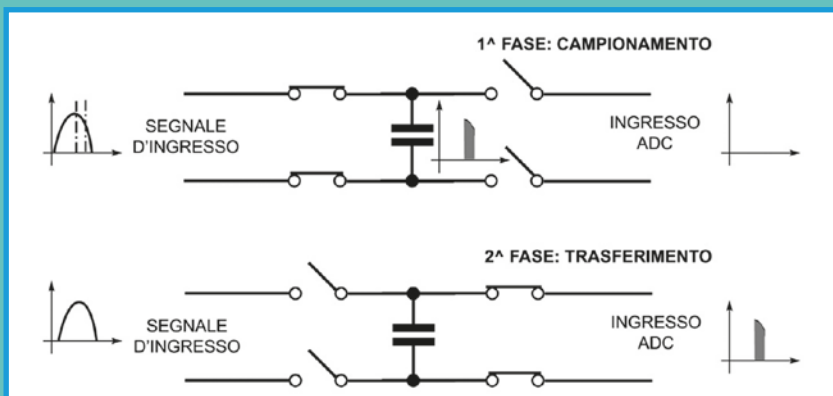


↑ Coefficiente di temperatura dei diodi Zener in funzione della tensione di Zener nominale.

to valanga innescato dagli elettroni che, strappati agli atomi, si spostano con un'energia tale da investire altri atomi di drogante e liberarne gli elettroni. Quindi, sebbene il comportamento sia praticamente lo stesso e tutti sono diodi Zener, in realtà si può parlare di effetto Zener solamente nei componenti con una V_z fino a 5,6 volt.

Questo è peraltro dimostrato dalla curva, mostrata nel grafico in questa pagina, dove è proposta la variazione della tensione del diodo Zener in funzione della temperatura: vediamo che sopra i 5,6V il coefficiente di temperatura è positivo, ossia occorre maggiore tensione di quella nominale per portare in diodo in conduzione inversa, e questo si spiega con il fatto che l'effetto valanga è ostacolato dall'agitazione termica nel semiconduttore, mentre sotto i 5,6V il coefficiente è negativo, perché l'energia termica si somma a quella del campo elettrico e fa liberare gli elettroni prima che sia raggiunta la tensione di Zener.

↓ Schematizzazione del Sample & Hold e del suo funzionamento: nella figura, il segnale campionato varia nell'intervallo di campionamento, ma il disegno ha il solo scopo di evidenziare la porzione di segnale d'ingresso campionata.



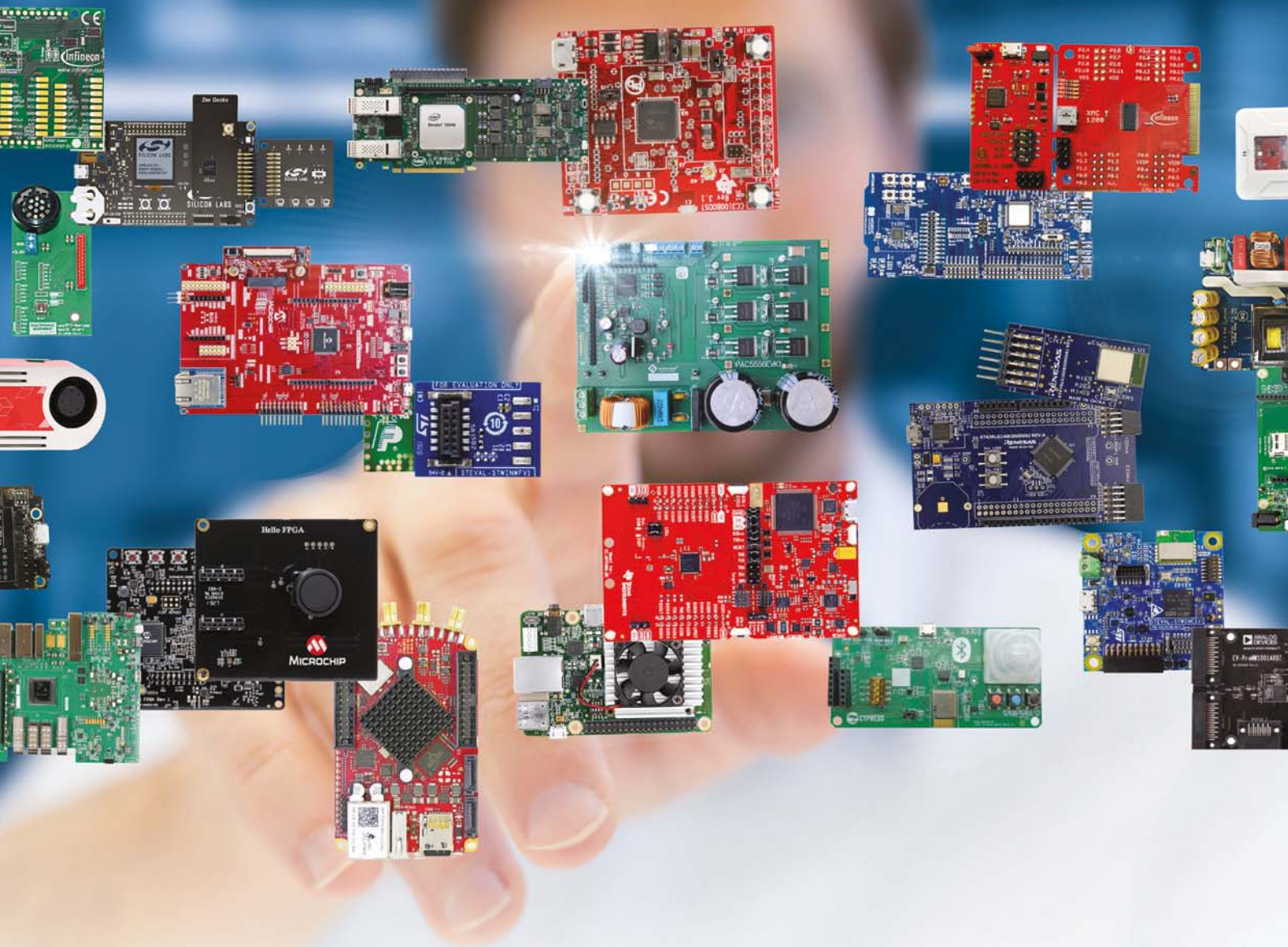
Web Forum e supporto tecnico

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.elettronicaain.it/webforum

Gli strumenti di sviluppo in un unico posto

Migliaia di strumenti da centinaia di produttori fidati



mouser.it/dev-tools



MOUSER
ELECTRONICS

Mentre le reti 5G continuano la loro diffusione in tutto il mondo, già si guarda a una nuova generazione di reti mobili, il 6G, le cui frequenze radio andranno nello spettro dei terahertz (THz). Questo offre più larghezza di banda, ma implica il superamento di alcuni ostacoli, dato che le frequenze THz producono condizioni di propagazione ancora più difficili rispetto alle mmWave. Inoltre a frequenze THz la lunghezza d'onda è comparabile con le dimensioni dei transistor: le tecnologie CMOS sembrano poter scalare solo fino a 140 GHz a causa della non linearità dei componenti.

Per affrontare questi problemi servono banchi di prova con capacità in tempo reale ad alte prestazioni. Poiché i banchi di prova THz avranno

➤ NI: strumenti di test per reti 6G



limitazioni di portata dovute alla perdita di percorso, inizialmente saranno costituite da semplici componenti a corto raggio come antenne a tromba. NI sta affrontando questi problemi con il suo sistema sub-THz e mmWave Transceiver System (MTS), che fornisce una piattaforma flessibile e ad alte prestazioni per dimostrare risultati

reali per la ricerca e la prototipazione ad alta frequenza.

L'architettura del sistema modulare può essere configurata per soddisfare una varietà di casi d'uso, costruita su un insieme comune di componenti. Gli esempi di riferimento di LabVIEW forniscono un punto di partenza per esperimenti di channel sounding e Physical Layer IP, consentendo all'utente di modificare l'IP per eseguire ricerche in nuove aree. Un'architettura di elaborazione multi-FPGA consente un sistema realmente in tempo reale senza necessità di elaborazione offline e con 2 GHz di larghezza di banda in tempo reale, consentendo prototipi over-the-air (OTA) di collegamenti di comunicazione bidirezionali.

www.ni.com

➤ LoRaWAN per lo smart metering del gas

Semtech Corporation collabora con il fornitore spagnolo di sistemi di misurazione delle utenze Arson Metering, il quale utilizza i dispositivi di comunicazione wireless Long-Range LoRa di Semtech e lo standard LoRaWAN per la sua nuova soluzione di sensori di misurazione del gas intelligente. Attraverso l'implementazione della connettività LoRaWAN, i sensori possono rilevare livelli pericolosi di metano nel gas, inviare una notifica in modalità wireless a un dispositivo mobile e quindi interrompere immediatamente la fornitura di gas. La soluzione del sensore è universale in cui può leggere tutti i tipi di contatori di acqua e gas di tutti i produttori. Secondo Arson Metering, il sensore può essere integrato per funzionare con tutti i tipi di tecnologie di misurazione, comprese quelle attuali e future.

Come fruitore dei prodotti Arson Metering, il fornitore di gas Nortegas ha convalidato l'uso di LoRaWAN per fornire connettività affidabile ai clienti



finali e, attraverso un'app mobile può fornire alert ai clienti se si verifica eccessiva concentrazione di metano e quindi una perdita, rilevata mediante sensori di gas metano e monossido di carbonio.

Si tratta di un'ulteriore conferma della validità della piattaforma da dispositivo a cloud. LoRa di Semtech è una

soluzione a lungo raggio e a bassa potenza adottata a livello globale per applicazioni IoT, che consente lo sviluppo e l'implementazione rapidi di reti IoT, gateway, sensori, moduli e prodotti a lungo raggio a bassissima potenza, convenienti e Servizi IoT in tutto il mondo.

www.semtech.com

› DataStorm: la piattaforma DAQ per gestire dati

Arrow Electronics propone DataStorm DAQ, una piattaforma di valutazione e sviluppo che collega i sensori e le tecnologie signal chain analogico/digitali di Analog Devices (ADI) a un FPGA con CPU integrata Intel. Questa piattaforma consente di rispondere alle esigenze di progettazione che sorgono dall'aumento dei requisiti di sistema in termini di velocità e precisione, oltre ai principali problemi relativi alla temporizzazione e all'integrità dei dati. DataStorm DAQ è una piattaforma di sviluppo basata sul SoC Cyclone V di Intel, progettata per una prototipazione rapida o per uno sviluppo proof-of-concept. Il SoC FPGA Cyclone V Intel integrato ha 85 k LE e contiene un

ARM Cortex-A9 MPCore duale, a supporto di un'ampia gamma di applicazioni. La piattaforma è particolarmente adatta per applicazioni di precisione nei settori di mercato della strumentazione e della sanità, dell'energia e in campo industriale, aerospaziale, automobilistico e delle comunicazioni. La scheda supporta connettori FMC e PMOD a basso numero di pin (LPC) per una facile estensibilità. Questi connettori consentono all'FPGA di interfacciarsi con componenti analogici e digitali della catena di segnali, come sensori, amplificatori, convertitori dati, attuatori e altri blocchi della signal chain. Per le applicazioni che richiedono misure di precisione, i test

› AF porta in auto l'Intelligenza Artificiale

L'elaborazione dati dei veicoli di nuova generazione sta migrando sempre più da sistemi di bordo a pochi grandi computer centralizzati ad alte prestazioni; per questo, il software in esecuzione su di essi dovrebbe essere facile da aggiornare ed espandere con nuove funzioni, come uno smartphone.

E tutto questo con i più alti standard di sicurezza funzionale per i sistemi di mobilità. Per raggiungere questo obiettivo, ZF ha rafforzato la propria attività nell'ambito del software automotive acquisendo una quota della software house software Apex.AI, che le permetterà di sviluppare soluzioni basa-

te sul metasistema operativo Apex.OS. Tra le soluzioni ZF per il veicolo software-defined, computer e software ad alte prestazioni svolgono un ruolo centrale. Nelle architetture elettriche ed elettroniche dei veicoli di prossima generazione, controlleranno tutte le funzioni importanti e le collegheranno sia con i sistemi del veicolo che con il mondo esterno. ZF Middleware è una piattaforma software aperta che funge da "mediatore" tra il sistema operativo del computer del veicolo e le sue applicazioni software. Le funzioni chiave del middleware sono l'astrazione dell'hardware del computer dalle applicazioni

software e la comunicazione tra queste applicazioni. Le funzioni software effettive, come quelle di guida autonoma, possono a loro volta essere costruite su questo middleware.

Apex.OS è un meta-sistema operativo derivato dal Robot Operation System (ROS), originariamente sviluppato per applicazioni di robotica; Apex.AI lo ha evoluto per il settore automobilistico e ha ottenuto la certificazione ASIL D secondo lo standard di sicurezza funzionale ISO 26262 per i veicoli; pertanto tale software di base per computer automobilistici ad alte prestazioni, inclusi i kit di sviluppo software associati, è pronto all'uso, offrendo alle case automobilistiche la possibilità di impiegare risorse umane in altre aree che offrono una differenziazione significativa per il cliente finale. Apex.OS può essere utilizzato in tutti i veicoli: autovetture, robotaxi, navette e veicoli commerciali, indipendentemente dal loro livello di automazione.

Apex.OS è scritto nei nuovi standard del linguaggio di programmazione C++, il che consente un'elevata riutilizzabilità dei singoli elementi costitutivi; questo approccio è un framework di comunicazione alternativo agli standard Adaptive Autosar ampiamente utilizzati nell'industria automobilistica.

www.zf.com



fra l'FPGA e i convertitori di dati di precisione possono essere eseguiti utilizzando schede mezzanine con connettori FMC LPC. Analogamente, la valutazione proof of concept può essere eseguita utilizzando data converter, sensori, componenti RF e altri. La piattaforma DataStorm DAQ include progetti e tool dimostrativi, come codice HDL, device driver ed esempi di progetti di riferimento per una prototipazione rapida e tempi di sviluppo ridotti. Supporta inoltre varie schede mezzanine attraverso una suite completa di strumenti e applicazioni demo; per esempio la scheda evaluation EVAL-AD4020FMCZ con 1,8 MSPS a 20 bit easy drive e ADC SAR differenziale e la scheda di valutazione multicanale EVAL-AD7768FMCZ dispone di ADC a campionamento simultaneo a 8 canali, 24 bit, power scaling e larghezza di banda 110,8 kHz. DataStorm DAQ integra i device driver I/O dei sottosistemi I/O industriali (IIo) Linux personalizzati con librerie



LibIIo e IIo Oscilloscope (un'applicazione dello spazio utente Linux).

www.arrow.com

› Ameca: il robot animato dal Machine Learning

Tra le novità del CES di Las Vegas spicca Ameca, il robot umanoide più avanzato al mondo, che rappresenta l'avanguardia della tecnologia robotica, non solo per le sue abilità motorie e la mimica facciale incredibilmente realistica di cui è capace, ma anche per le sue implicazioni a livello di Intelligenza Artificiale e Machine Learning. Infatti questo avanzatissimo robot, dalle sembianze umane, è stato concepito come piattaforma su cui sperimentare algoritmi di Machine Learning: ha un corpo quasi umano e un cervello elettronico in cui si eseguono algoritmi di AI e Machine Learning. Il viso ha una sorprendente somiglianza con quella di un essere umano e le espressioni di cui è capace -comprese quelle di gioia e di rabbia- sorprendono e quasi spaventano per il loro realismo o meglio, per la loro umanità.

Progettato specificamente come piattaforma per lo sviluppo nelle future tecnologie di robotica, Ameca è la piattaforma robotica umanoide perfetta per sperimentare e implementare l'interazione uomo-robot: è modulare, aggiornabile e facile da sviluppare.

Se vogliamo, Ameca è il corpo umano artificiale dove l'Intelligenza Artificiale può esprimersi al meglio e i due formano quel connubio noto come AI

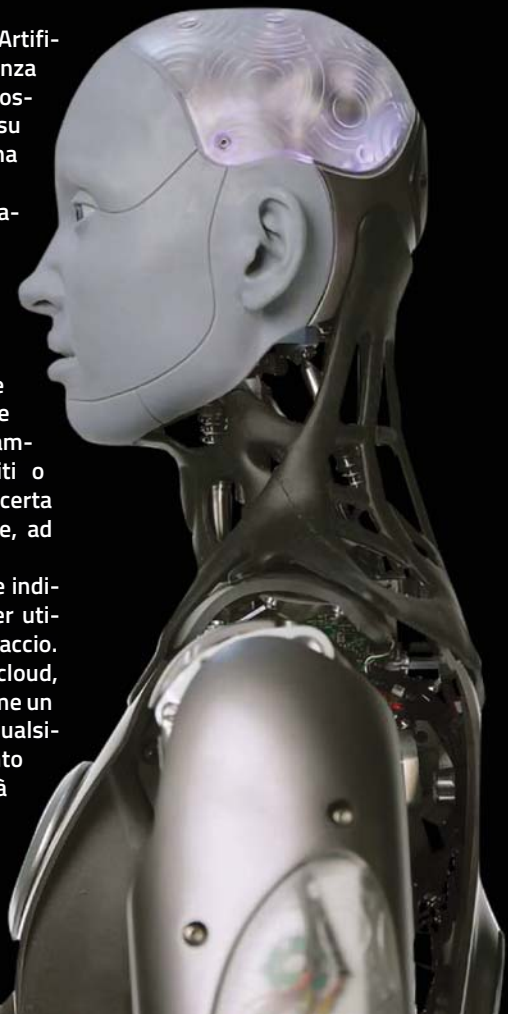
x AB (Artificial Intelligence for Artificial Body); i sistemi di intelligenza artificiale e machine learning possono essere testati e sviluppati su Ameca insieme al potente sistema operativo per robot Tritium.

L'hardware di Ameca nasce dall'avanzata tecnologia Mesmer.

L'architettura modulare consente futuri aggiornamenti, sia fisici che a livello software, per migliorare le capacità di Ameca; quindi lo sviluppatore può, mantenendo l'umanoide tale e quale, personalizzare il programma affinché svolga vari compiti o semplicemente per ottenere una certa interazione con noi umani. Come, ad esempio, stringerci la mano.

Tutti i moduli possono funzionare indipendentemente in modo da poter utilizzare solo la testa o solo un braccio. Pensato per essere connesso al cloud, il robot può essere controllato come un avatar personale disponibile da qualsiasi parte del mondo. Il movimento fluido e realistico e le capacità avanzate di espressione facciale consentono ad Ameca di stabilire un rapporto e comunicare con chiunque.

<https://www.engineeredarts.co.uk/robot/ameca/>



La Presa su misura per te!

PRESA MULTIPLA SCHUKO 6 POSIZIONI CON PROTEZIONE DA SOVRATENSIONE PROTECTION

€ 21,⁵⁰

Cod. **WTPRESAM6P**



Presa multipla universale con 6 prese Schuko (angolate di 45°) con protezione da sovratensione, interruttore ON/OFF luminoso, struttura robusta con alloggiamento antiurto, LED di sicurezza luminoso. Lunghezza del cavo 1,4 metri con spina Schuko, corrente massima 16 A, potenza massima 3500 W.

SET ESPANDIBILE PRESA DA RETE RADIOCOMANDATA

Set composto da una presa passante radiocomandata e da un radiocomando portatile in grado di attivare (ON/OFF) un carico massimo di 1100 W. Il set è espandibile, con la possibilità di aggiungere ulteriori trasmettitori e abbinare fino a quattro prese, ciascuna associata ai quattro canali operativi del trasmettitore. Frequenza di lavoro 433 Mhz con codifica ad elevato grado di sicurezza, basso consumo energetico. Presa di tipo tedesco Schuko con protezione per bambini. Il radiocomando copre distanze fino a 30 metri e risulta particolarmente adatto per controllare le prese posizionate in punti difficili da raggiungere. Il trasmettitore è alimentato con batteria di tipo 23A inclusa nella confezione.

€ 19,⁹⁰
cod. **WTWRS**



Prodotti aggiuntivi al set WTWRS:

PRESA RADIOCOMANDATA ADDIZIONALE AL SET WTWRS



€ 12,⁹⁰
Cod. **RXWTWRS**

RADIOCOMANDO 4 CANALI ADDIZIONALE AL SET WTWRS



€ 9,⁰⁰
Cod. **TXWTWRS**

PRESA TIMER DIGITALE SETTIMANALE DA ESTERNO 3600W - IP44

Consente di accendere e spegnere apparecchiature elettriche nei giorni e nelle ore della settimana prestabilita. Dispone di pulsante On/Off, 10 programmi di funzionamento per ciascun giorno della settimana con step di regolazione di 1 minuto. Ha inoltre 16 diverse combinazioni giornaliere, batteria di backup, formato dell'orario 12/24 ore, funzione di regolazione automatica dell'ora legale, presa di tipo SCHUKO. Adatto per impieghi all'esterno.



€ 16,⁵⁰
Cod. **WT51301**

PRESA DI CORRENTE SCHUKO DA 16A CON 2 USCITE USB DA 5 V - 3,15 A



€ 11,⁹⁰
Cod. **WT40885**

Presse di corrente passante con due prese USB da 5 V - 3,15 A (totali). Ideale per ricaricare smartphone, lettore MP3, fotocamera digitale o tablet tramite cavo USB.

PRESA DI RETE SCHUKO CON PROTEZIONE DA SOVRATENSIONE



€ 6,⁶⁰
Cod. **WT51270**

Presse in grado di proteggere i dispositivi elettrici dalle sovratensioni come quelle causate dai fulmini. Dispone inoltre di LED di stato che si accende appena viene alimentata.

Melexis presenta DVK per sensori di corrente

Melexis ha introdotto due nuovi kit di sviluppo che consentono ai progettisti di studiare le capacità funzionali dei circuiti integrati per sensori di corrente Melexis, riducendo così i tempi di sviluppo e accelerando il completamento del progetto. Un primo Development Kit (DVK) consente di sperimentare dispositivi con tecnologia IMC-Hall (con o senza schermatura), mentre il secondo svolge le stesse funzioni con la tecnologia Hall basata su core convenzionale. Mediante il loro impiego, gli ingegneri saranno in grado di convalidare il proprio hardware con diversi tipi di sensore e valutare diversi intervalli di rilevamento della corrente. I nuovi DVK sono dotati di numerosi sensori di corrente Melexis. In questo modo gli ingegneri possono testare diverse configurazioni magnetiche e vari circuiti integrati, risparmiando tempo e fatica nel processo di selezione. Il DVK per IMC-Hall include i circuiti in-



tegrati MLX91208 e MLX91216, oltre al MLX91218 lanciato di recente. Il DVK per effetto Hall convenzionale viene fornito con MLX91209, MLX91211, MLX91217 e MLX91219. Questi dispositivi coprono efficacemente la seconda generazione di sensori di corrente Melexis. I DVK sono dotati di una struttura stampata in 3D che consente di mantenere in posizione

gli schermi e i nuclei. In questo modo gli ingegneri non hanno bisogno di costruire la struttura di prova, risparmiando tempo e fatica. I materiali utilizzati sono in grado di affrontare temperature molto elevate (170 °C). I DVK includono anche bus bar, che semplificano il processo di installazione.

www.melexis.com

Terza generazione di MOSFET SiC STPOWER di STM

Con l'accelerazione del mercato dei veicoli elettrici, molte case automobilistiche e fornitori del settore automotivo stanno adottando sistemi di azionamento a 800 V per velocizzare sensibilmente le operazioni di ricarica e ridurre il peso dei veicoli. I nuovi MOSFET SiC (Carburo di Silicio) di STM sono ottimizzati in modo specifico per soddisfare tali requisiti in applicazioni automotive che includono inverter di trazione per EV, caricabatterie di bordo e convertitori DC/DC, oltre a compressori per climatizzatori a pompa di calore. La nuova generazione è

adatta anche alle applicazioni industriali in quanto aumenta considerevolmente l'efficienza degli azionamenti dei motori industriali, dei convertitori di energia rinnovabile e dei sistemi di accumulo, così come dei gruppi di alimentazione per telecomunicazioni e data center. STM ha completato la qualificazione della piattaforma tecnologica SiC di terza generazione e prevede di portare alla maturità commerciale la maggior parte dei prodotti derivati entro la fine del 2021. Saranno disponibili MOSFET con tensioni nominali da 650 V e 750 V fino a 1200 V, che offriranno ai progettisti flessibilità in applicazioni operanti dalle tensioni di rete fino a quelle delle batterie e dei caricabatterie ad alta tensione dei veicoli elettrici.

www.st.com

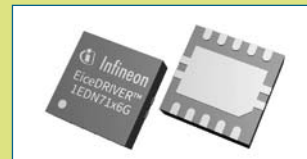


Gate driver single-channel Infineon da 200V

Infineon presenta la famiglia di gate driver GaN a singolo canale EiceDRIVER 1EDN71x6G HS 200V, progettata per elevare le prestazioni degli HEMT CoolGaN Schottky Gate (SG) e compatibile con HEMT GaN e MOSFET al silicio. I gate driver sono orientati ad applicazioni come convertitori DC-DC, azionamenti motore, robot, droni, utensili elettrici e amplificatori audio in classe D. I chip sono dotati di controllo pull-up e pull-down selezionabile, che consentono l'ottimizzazione della forma d'onda e della velocità di commutazione senza bisogno di resistori di gate. La versione più veloce (1EDN7116G) è adatta per configurazioni half-bridge con parallelismo significativo. La versione con minore velocità e meno

potente (1EDN7146G) può essere impiegata dove basti un dv/dt limitato, come azionamenti di motori o HEMT con alta RDSon e basso Qg. La funzione di ingresso logico veramente differenziale (TDI) elimina il rischio di falsi trigger dovuti al rimbalzo di massa nelle applicazioni low-side e consente ai dispositivi 1EDN71x6G di indirizzare anche le applicazioni high-side. Tutte le versioni sono dotate di un contatto Miller attivo per evitare l'accensione indotta. I gate driver sono incapsulati in package PG-VSON-10.

www.infineon.com



Da ROHM, driver per converter flyback AC/DC da 45W con MOSFET a super giunzione



Rohm propone i nuovi IC serie BM2P06xMF-Z per convertitori flyback AC/DC contenenti un MOSFET SJ (a super giunzione) con tensione di breakdown di 730 V, ideali per Switch Mode Power Supply (SMPS) in azionamenti industriali ed elettrodomestici, condiziona-

tori d'aria e apparecchiature di automazione industriale. Non richiedono dissipatori di calore né condensatori e resistori di scarica e rispondono alle esigenze degli alimentatori AC/DC, che devono accettare tensioni comprese tra 85 e 264 Vca, ma anche ottemperare

a standard come Energy Star per il risparmio energetico e IEC 62368 per la sicurezza. Gli integrati, in SMD, possono erogare 45 W di potenza e una tensione di 60Vcc, senza bisogno di alimentatore step-down esterno.

www.rohm.com

Nuove MCU Renesas automotive

Con l'avvento delle nuove architetture E/E, che supportano anche le applicazioni di controllo di zona e dominio, i meccanismi di controllo stanno evolvendo per integrare ulteriori funzionalità necessarie in campo automotive; tra queste, ad esempio, il controllo delle luci, dei finestrini e degli specchietti, il controllo motore per le pompe e ventole del sistema di raffreddamento del motore, la gestione di diversi tipi di sensori. Nel prossimo futuro, l'alta velocità e la connettività sicura con i controlli di zona e dominio sarà fondamentale per le unità elettroniche di controllo (ECU). La nuova generazione di MCU RL78/F24 e RL78/F23 di Renesas soddisfano le attuali richieste delle applicazioni di controllo di attuatori e sensori, offrendo security avanzata, numerose opzioni di connettività e sicurezza funzionale. I nuovi dispositivi supportano il protocollo di comunicazione ad alta velocità CAN FD (RL78/F24) e le funzionalità di security EVITA-Light e sono ottimizzati per sistemi che devono essere certificati ASIL-B secondo lo standard di safe-



ty funzionale ISO 26262. I nuovi micro hanno frequenze operative fino a circa 70% più alte rispetto alla generazione precedente, permettendo di incrementare le prestazioni più del doppio nelle applicazioni di controllo di motori brushless (BLDC). Renesas ha inoltre migliorato l'acceleratore hardware e i timer per il controllo motore ed ha ag-

giunto un convertitore A/D a 12 bit, con l'obiettivo di offrire ai propri clienti le funzionalità e i livelli di prestazioni da loro richiesti. Le caratteristiche principali dei microcontrollori RL78/F24 e RL78/F23 sono la frequenza operativa di 40 MHz e interfacce di connettività come CAN FD (RL78/F24), LIN, SPI, e I²C-Bus.

www.renesas.com

SoC RadioVerse per ottimizzare le prestazioni delle Radio Unit 5G

Analog Devices propone nuovi System-on-Chip (SoC) RadioVerse, che forniscono agli sviluppatori di unità radio (RU) una piattaforma SDR (Software Defined Radio) agile e conveniente. I nuovi SoC offrono un'avanzata elaborazione del segnale RF con funzionalità digitali e capacità RF che migliorano le presta-

zioni e l'efficienza energetica dei sistemi 5G; combinano l'architettura Zero IF (ZiF) con significativi progressi nell'integrazione funzionale e nella linearizzazione. L'ADRV9040 è il primo dei nuovi SoC RadioVerse: dotato di otto canali di trasmissione e ricezione con larghezza di banda di 400 MHz, integra funzioni

avanzate di elaborazione del segnale digitale, tra cui up-converter (CDUC), down-converter (CDDC), riduzione del fattore di cresta (CFR) e pre-distorsione digitale (DPD). Gli algoritmi DPD convalidati 4G e 5G, incluse varie tipologie di tecnologia PA come il nitruro di gallio (GaN).

www.analog.com



Interruttori invisibili da Schurter

L'integrazione degli switch e pulsanti nelle interfacce utente e nei relativi pannelli, di solito richiede lavorazioni di fresatura o foratura, ma in alcune applicazioni, specialmente quelle che sono molto sensibili in termini di igiene o particolarmente esigenti in termini di design, questo è poco accettabile e si cercano soluzioni alternative, tra cui gli schermi tattili; purtroppo questi vanno



bene per realizzare controlli virtuali, mentre per quelli fisici servono componenti elettromeccanici o capacitivi. Per venire incontro alle esigenze dei progettisti che vogliono pulsanti e interruttori a minimo impatto visivo e incorporati nei pannelli, Schurter ha creato una nuova linea di interruttori chiamata CHS (Capacitive Hidden Switch, ossia

interruttore capacitivo nascosto) che può essere applicato tramite una pellicola adesiva in modo invisibile dietro a quasi tutti i pannelli di interfaccia utente in materiale anche non conduttivo.

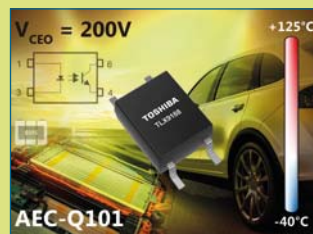
Gli interruttori capacitivi Schurter si basano su un alimentatore che genera un campo elettrico costante sulla superficie di il sensore (attuatore). Toccando il sensore con un dito,

che è a sua volta circondato da un campo elettrico, si provoca una reazione sul lato sensore. L'interruttore è quindi innescato dall'interazione di due campi elettrici. I touchscreen degli smartphone, dei tablet e dei computer funzionano in base a questo principio. La sensibilità dell'interruttore può essere regolata.

www.schurter.com

Opto automotive Toshiba da 200V

Toshiba ha sviluppato il fotoaccoppiatore automotive con uscita a transistor ad alta tensione TLX9188, per applicazioni che includono la comunicazione di segnali isolata in sistemi automotive quali i veicoli ibridi e full-electric. L'opto è in grado di funzionare con una tensione collettore-emettitore (V_{ce}) di 200V. Il dispositivo offre un transfer-rate di corrente (IC/IF) fino al 600% e una tensione di isolamento ingresso-uscita di 3750 Vrms (min.). Il TLX9188 è alloggiato in un package SO6 a 4 pin di dimensioni e spessore molto contenuti che misura appena 3,7 mm x 7,0 mm x 2,1 mm, adatto per i formati più recenti ad alta densità. È in grado di operare in un ampio intervallo di temperature da -40°C a



+125°C ed è pienamente qualificato secondo lo standard AEC-Q101 per applicazioni automotive. La capacità di operare ad alte tensioni rende il fotoaccoppiatore TLX9188 adatto per l'utilizzo nei sistemi di gestione delle batterie dei veicoli ibridi ed elettrici, in cui può essere utilizzato, tra l'altro, per monitorare le tensioni e la resistenza dei relè meccanici e rilevare guasti a terra. Il dispositivo è già in commercio.

www.toshiba.com

MOSFET SUPERFET V da ON, per server e telecomunicazioni

Onsemi annuncia la nuova famiglia di MOSFET 600V SUPERFET V ad alte prestazioni, che consentono agli alimentatori di soddisfare le più stringenti normative di efficienza energetica come 80 PLUS Titanium, soprattutto nelle condizioni di carico più impegnative. La famiglia è suddivisa in tre categorie: FAST, Easy Drive e FRET, tutte ottimizzate per fornire le migliori prestazioni in una varietà di applicazioni e topologie.

La famiglia 600V SUPERFET V offre eccellenti caratteristiche di commutazione e un rumore di gate inferiore con conseguente miglioramento delle prestazioni elettromagnetiche (EMI), un vantaggio significativo per server e sistemi di telecomunicazioni. Inoltre, l'affidabilità del sistema è migliorata dalla presenza di un robusto diodo di recovery e da una elevata V_{gss} .

Le versioni FAST offrono la massima efficienza nelle topologie di commutazione hardware (come PFC di fascia alta) e sono ottimizzate per fornire una ridotta carica di gate (Q_g) e basse perdite EOSS per consentire una commutazione rapida. I dispositivi includono l'NTNLO41N60S5H ($R_{DSon}=41$ mohm) e l'NTHL185N60S5H ($R_{DSon}=185$ milliohm), entrambi in package TO-247. L'NTP185N60S5H è in package TO-220 mentre l'NTMT185N60S5H è in package Power88 da 8x8x1mm, che garantisce un livello MSL1 (Moisture Sensitivity Levels) e ha una configurazione Kelvin source che migliora il rumore di gate e la perdita di potenza

di commutazione.

Le versioni Easy Drive sono adatte a topologie di commutazione sia hard che soft e contengono un resistore di gate interno (R_g) e una capacità ottimizzata. Sono adatti all'uso generico in molte applicazioni, tra cui PFC e LLC. In questi dispositivi, il diodo Zener integrato tra l'elettrodo di gate e l'elettrodo di source da 120 milliohm (R_{DSon}) fornisce meno sollecitazioni sull'ossido di gate e una maggiore robustezza ESD che porta a una migliore resa dell'insieme con un numero di guasti inferiore. Attualmente sono disponibili due dispositivi con RDS (on), da 99 milliohm e 120 milliohm: NTHL099N60S5 e NTHL120N60S5Z, entrambi con package TO-247.

Le versioni a ripristino rapido (FRFET) sono adatte a topologie di commutazione soft come il full-bridge a sfasamento (PSFB) e le soluzioni LLC. Beneficiano di un diodo body veloce e offrono Q_{rr} e T_{rr} ridotti. L'NTP125N60S5FZ con diodo Zener integrato, presenta una R_{ds} di 125 milliohm ed è disponibile in package TO-220 mentre l'NTMT061N60S5F presenta 61mohm di R_{ds} in package Power88. Il dispositivo con il valore R_{DSon} più basso è l'NTHL019N60S5F, in package TO-247, con R_{ds} 19 milliohm.

www.onsemi.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.it/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"

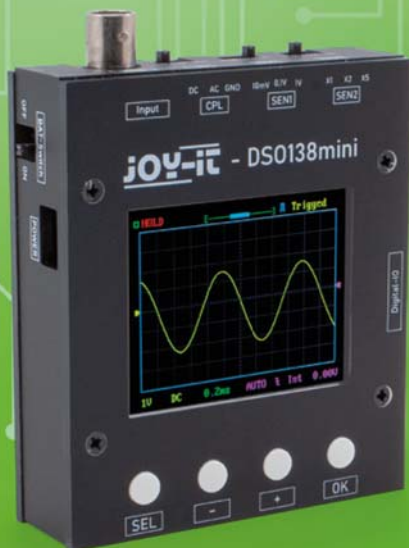
La possibilità di **LEGGERE GRATUITAMENTE**
la versione digitale di ELETTRONICA IN



Ora i componenti non hanno più segreti!

DISPOSITIVI PROFESSIONALI PER MISURE E ANALISI PRECISE

MINI OSCILLOSCOPIO A 1 CANALE CON DISPLAY TFT LCD DA 2,4 POLLICI



COD. DSO138MINI
€ 59,90

Dispone di display da 2,4" TFT LCD - 320x240, larghezza di banda da 200 kHz e frequenza di campionamento 1 MSa/s con risoluzione 12 bit. I circuiti di alimentazione sono stati semplificati rispetto al modello precedente, così come la portabilità grazie alle dimensioni del case notevolmente ridotte. L'ingresso di trigger esterno accetta segnali TTL e LVTTTL come sorgente e permette infine di esportare (in formato testo) la forma d'onda in seriale. Alimentazione attraverso connettore micro USB. Questo oscilloscopio è particolarmente indicato per gli hobbisti interessati a misurare e visualizzare la maggior parte dei segnali elettronici.

TESTER PER COMPONENTI ELETTRONICI



COD. JT-LCRT7
€ 29,00

Dotato di display LCD da 3.5" a colori, ricevitore a infrarossi e batteria ricaricabile integrata da 3,7 volt 350 mAh. Permette di misurare diodi, diodi zener, resistori, condensatori, l'ESR (Resistenza Equivalente Serie) di un condensatore, induttori, tiristori, triac, FET, transistor bipolari (indicando se è NPN o PNP) e batterie (max. 4,5 V). Funziona anche con condensatori elettrolitici polarizzati e non polarizzati in entrambi gli orientamenti. Grazie al ricevitore a infrarossi integrato possono essere decodificati anche i segnali a infrarossi e visualizzati sul display, in forma d'onda e codice esadecimale. Le misurazioni vengono avviate automaticamente premendo semplicemente un pulsante.

Prezzi IVA inclusa.

Analizzatore di componenti DCA55



€ 73,00
cod. DCA55

Consente di analizzare una vasta gamma di componenti attivi, identificando la pinout e fornendo informazioni relative ai principali parametri.

Analizzatore di semiconduttori DCA75



€ 148,00
cod. DCA75

Analizza svariati tipi di semiconduttori. Può essere interfacciato al PC e, grazie al software in dotazione, fornisce un'analisi dettagliata del componente.

Misuratore di induttanza, capacità, resistenza e impedenza LCR45



€ 129,00
cod. LCR45

Strumento multifunzione. Misura induttanza, resistenza e impedenza arbitraria. Riconosce automaticamente il tipo di componente, e fornisce informazioni relative ai principali parametri.

SERVIZI CLOUD IoT PER MAKERS



dell' Ing. **VINCENZO MENDOLA**

Arduino è uno degli strumenti più adatti per iniziare a sperimentare con l'IoT. Vediamo quali sono i principali servizi cloud per makers per inviare e collezionare i dati acquisiti.

A

ad aver permesso una rapida diffusione dell'Internet of Things (IoT, internet delle cose) tra i makers, che, per essere facilmente utilizzato anche da questi ultimi, necessita di servizi di accesso al cloud economici e facilmente configurabili. Vediamo quali sono le principali soluzioni sviluppate appositamente per i makers.

UNA VOLTA ERA PACHUBE

Chi di voi ha iniziato ad utilizzare Arduino e le altre board di prototipazione veloce già da un po' di anni si è sicuramente imbattuto in PACHUBE, uno dei primi servizi cloud per makers,

e probabilmente ne ha anche utilizzato il servizio. Negli anni ha subito varie acquisizioni e trasformazioni, diventando prima Cosm una volta acquisito da Logmein ed infine Xively, ora proprietà di Google che ne ha un po' snaturato l'essenza "makers", rendendolo un prodotto rivolto principalmente ad un'utenza business anche se esiste ancora un piano gratuito rivolto a chi vuole sperimentare.

GLI ATTUALI SERVIZI IOT PER MAKERS: INIZIAMO CON ARDUINO

Oggi i servizi IoT per makers più famosi, diffusi ed utilizzati sono tre: Arduino IoT cloud, Adafruit IO e ThingsSpeak di Mathworks (i creatori di matlab). Tutti e tre presentano almeno un piano gratuito per testare il servizio. Vediamone le principali caratteristiche, le peculiarità e le differenze.

Cominciamo con Arduino IoT cloud: ha 4 piani di cui uno base, gratuito, che permette di creare infinite dashboard ma mette a disposizione soltanto due grandezze (things) che possono essere monitorate; inoltre i dati inviati nel cloud possono essere archiviati soltanto per un giorno. Tenendo conto di queste sue caratteristiche, possiamo considerarlo utile essenzialmente per sperimentare e prendere confidenza con il servizio offerto. Per poter utilizzare in maniera più estesa le potenzialità di questo cloud, è necessario passare ad uno dei pacchetti superiori,

chiamati rispettivamente "entry", "maker" e "maker plus". Già con il piano "entry" si possono fare delle cose interessanti, con 10 grandezze indipendenti monitorabili, con un'archiviazione dei dati trasmessi di 15 giorni; il piano maker porta a 25 gli oggetti con una data retention di 3 mesi mentre con "maker plus" si arriva a 100 "things", archiviati per un massimo di un anno, sicuramente utile per chi ha esigenze professionali o semi professionali. A partire dal piano "entry" è possibile utilizzare le API di arduino IoT cloud (<https://www.arduino.cc/reference/en/iot/api>). Ovviamente passando dal piano "free" via via al piano "maker plus", di pari passo al miglioramento delle caratteristiche del servizio offerto, aumentano di conseguenza anche i prezzi. In **Fig. 1** è possibile vedere una comparativa delle caratteristiche e dei prezzi delle versioni disponibili del cloud offerto da Arduino. Nel **Listato 1** vediamo ora un semplice sketch per MKR1010 WIFI per inviare la temperatura misurata con un comunissimo sensore DS18B20 su questo cloud. Il codice è molto semplice e le modifiche rispetto allo sketch utilizzato senza la connessione al cloud sono minime. Si vuole far notare che queste aggiunte al codice vengono effettuate automaticamente dall'IDE di Arduino nel cloud. Quindi se volete utilizzare il servizio Arduino dovete obbligatoriamente utilizzare la versione nel cloud

Fig. 1
Confronto dei piani offerti da Arduino IoT cloud.

MONTHLY ☒ YEARLY			
FREE	ENTRY	MAKER	MAKER PLUS
Everything you need to learn Arduino, build your first IoT project and control it from your phone.	Get unlimited storage, scale up your IoT projects and get access to advanced features.	For makers that are getting serious and need a reliable and sophisticated IoT platform to run their projects.	The option for makers with ambitions, that need to manage a small fleet of connected devices.
✓ 2 Things ✓ Unlimited dashboards ✓ 100 Mb to store sketches ✓ 1 day data retention ✓ 200s/day of compilation time	✓ 10 Things ✓ Unlimited dashboards ✓ Unlimited storage for sketches ✓ 15 days data retention ✓ Unlimited compilation time ✓ APIs ✓ Over the Air Updates	✓ 25 Things ✓ Unlimited dashboards ✓ Unlimited storage for sketches ✓ 3 months data retention ✓ Unlimited compilation time ✓ APIs ✓ Over the Air Updates ✓ Dashboard sharing	✓ 100 Things ✓ Unlimited dashboards ✓ Unlimited storage for sketches ✓ 1 year data retention ✓ Unlimited compilation time ✓ APIs ✓ Over the Air Updates ✓ Dashboard sharing
Free	\$ 2.99/month Plus applicable taxes. Not available in Brazil.	\$ 6.99/month Plus applicable taxes. Not available in Brazil.	\$ 23.99/month Plus applicable taxes. Not available in Brazil.
GET STARTED	GET STARTED	GET STARTED	GET STARTED

↓ Listato 1

```
// First we include the libraries
/*****
#include <OneWire.h>
#include <DallasTemperature.h>
#include "thingProperties.h" //automatically added by arduino IoT cloud
*****/
// Data wire is plugged into pin 2 on the Arduino
#define ONE_WIRE_BUS 2
/*****
// Setup a oneWire instance to communicate with any OneWire devices
// (not just Maxim/Dallas temperature ICs)
OneWire oneWire(ONE_WIRE_BUS);
*****/
// Pass our oneWire reference to Dallas Temperature.
DallasTemperature sensors(&oneWire);
/*****
void setup(void)
{
  // start serial port
  Serial.begin(9600);
  sensors.begin();
  sensors.setResolution(11); //DALLAS RESOLUTION SETTING
  // Start up the library
  // Defined in thingProperties.h
  initProperties();

  // Connect to Arduino IoT Cloud
  ArduinoCloud.begin(ArduinoIoTPreferredConnection);
  setDebugMessageLevel(2);
  ArduinoCloud.printDebugInfo();
}

void loop(void)
{ ArduinoCloud.update();

  sensors.requestTemperatures();
  // call sensors.requestTemperatures() to issue a global temperature
  // request to all devices on the bus
  /*****
  Serial.print(" Requesting temperatures...");
  //temperature=sensors.requestTemperatures(); // Send the command to get temperature readings
  Serial.println("DONE");
  *****/
  Serial.print("Temperature is: ");
  Serial.println(sensors.getTempCByIndex(0)); // Why "byIndex"?
  // You can have more than one DS18B20 on the same bus.
  // 0 refers to the first IC on the wire
  delay(100);
  temperature=sensors.getTempCByIndex(0);
  delay(1000);
}
```

che inserirà queste modifiche al vostro codice:

- verrà inclusa la libreria `#include "thingProperties.h"` necessaria per utilizzare il servizio;
- nel `setup` verrà aggiunto `initProperties();` e `ArduinoCloud.begin(ArduinoIoTPreferredConnection);` per inizializzare la comunicazione con il cloud;
- le istruzioni `setDebugMessageLevel(2);` e `ArduinoCloud.printDebugInfo();` dovranno essere utilizzate per abilitare i messaggi di debug;
- nel loop troveremo la funzione `ArduinoCloud.update();` che permette l'aggiornamento della variabile

inviata nel cloud, che nel nostro caso abbiamo chiamato `"temperature"` (`temperature=sensors.getTempCByIndex(0);`).

Notare che `"temperature"` viene definita solamente durante la configurazione della variabile direttamente nel cloud, nella sezione `"things"` e non nel nostro codice.

ADAFRUIT IO

Adafruit offre un servizio con una scelta molto



↓ Listato 2

```
/*  
  
https://github.com/avilmaru/iot_arduino_adafruit/blob/master/readsen-  
sors_sendtoadafruit_iotcloud.ino  
  
*/  
  
#include <ArduinoJson.h>  
#include <avr/dtostrf.h>  
#include <SPI.h>  
#include <WiFiNINA.h>  
#include "arduino_secrets.h"  
#include <OneWire.h>  
#include <DallasTemperature.h>  
  
#define ONE_WIRE_BUS 2  
  
char ssid[] = SECRET_SSID;        // your network SSID (name)  
char pass[] = SECRET_PASS;        // your network password (use for WPA, or use as key for WEP)  
int status = WL_IDLE_STATUS;  
char server[] = "io.adafruit.com"; // name address for Adafruit IoT Cloud  
  
////////////////////////////////////  
  
unsigned long lastConnectionTime = 0; // last time you connected to the server, in milliseconds  
const unsigned long postingInterval = 7000; // delay between updates, in milliseconds  
// Initialize the client library  
WiFiClient client;  
  
// Setup a oneWire instance to communicate with any OneWire devices  
// (not just Maxim/Dallas temperature ICs)  
OneWire oneWire(ONE_WIRE_BUS);  
/*****  
// Pass our oneWire reference to Dallas Temperature.  
DallasTemperature sensors(&oneWire);  
  
void setup() {  
  Serial.begin(9600);  
  sensors.begin();  
  sensors.setResolution(11); //DALLAS RESOLUTION SETTING  
  connectToWiFi(); // start the WiFi OTA library with internal (flash) based storage  
}  
  
void loop() {  
  
  // if 7 seconds have passed since your last connection,  
  // then connect again and send data:  
  if (millis() - lastConnectionTime > postingInterval)  
  {  
  
    /*****  
  
    displayValuesOnSerial();  
  
    if (WiFi.status() != WL_CONNECTED){WiFi.begin(ssid, pass);  
    // wait 10 seconds for connection:  
    delay(10000);} //if WIFI NOT CONNECTED RECONNECT  
  
    httpRequest(); // send data to Cloud  
  
  }  
  
  delay(10000); // wait for a second  
}  
  
/*  
 * This method makes a HTTP connection to the server and post deread sensor values  
 * to the Adafruit IoT Cloud  
 */  
  
void httpRequest()  
{  
  
  const size_t capacity = JSON_ARRAY_SIZE(3) + 3*JSON_OBJECT_SIZE(2) + 2*JSON_OBJECT_SIZE(3) + 130;
```



```

StaticJsonDocument<capacity> doc;

// Add the "location" object
JsonObject location = doc.createNestedObject("location");
location["lat"] = 0;
location["lon"] = 0;
location["ele"] = 0;

// Add the "feeds" array
JsonArray feeds = doc.createNestedArray("feeds");
JsonObject feed1 = feeds.createNestedObject();
feed1["key"] = "";
feed1["value"] = sensors.getTempCByIndex(0);

delay(1000);
// close any connection before send a new request.
// This will free the socket on the Nina module
client.stop();
// WiFi.end();
delay(5000); //send data every 5s

Serial.println("\nStarting connection to server...");
if (client.connect(server, 80))
{
    Serial.println("connected to server");
    // Make a HTTP request:
    client.println("POST /api/v2/" IO_USERNAME "/" IO_GROUP "/data HTTP/1.1");
    //Serial.println("POST /api/v2/" IO_USERNAME "/" IO_GROUP "/data HTTP/1.1");
    client.println("Host: io.adafruit.com"); //Serial.println("Host: io.adafruit.com");
    client.println("Connection: close"); //Serial.println("Connection: close");
    client.print("Content-Length: "); //Serial.print("Content-Length: ");
    client.println(measureJson(doc)); // Serial.println(measureJson(doc));
    client.println("Content-Type: application/json"); //Serial.println("Content-Type: application/json");
    client.println("X-AIO-Key: " IO_KEY); //Serial.println("X-AIO-Key: " IO_KEY);

    // Terminate headers with a blank line
    client.println();
    // Send JSON document in body
    serializeJson(doc, client);

    // note the time that the connection was made:
    lastConnectionTime = millis();

    Serial.println("data sent!");
} else {
    // if you couldn't make a connection:
    Serial.println("connection failed!");
}
}

void connectToWiFi()
{
    // check for the WiFi module:
    if (WiFi.status() == WL_NO_MODULE) {
        Serial.println("Communication with WiFi module failed!");
        // don't continue
        while (true);
    }

    String fv = WiFi.firmwareVersion();
    if (fv < "1.0.0") {
        Serial.println("Please upgrade the firmware");
    }

    // attempt to connect to Wifi network:
    while (status != WL_CONNECTED) {

        Serial.print("Attempting to connect to SSID: ");
        Serial.println(ssid);
        // Connect to WPA/WPA2 network. Change this line if using open or WEP network:
        status = WiFi.begin(ssid, pass);
        WiFi.lowPowerMode(); //USE LOW POWER WIFI MODE
        // wait 10 seconds for connection:
        delay(10000);
    }
}

```

```

Serial.println("Connected to wifi");
delay(2000);
printWifiStatus();
}

void printWifiStatus() {
  // print the SSID of the network you're attached to:
  Serial.print("SSID: ");
  Serial.println(WiFi.SSID());

  // print your board's IP address:
  IPAddress ip = WiFi.localIP();
  Serial.print("IP Address: ");
  Serial.println(ip);

  // print the received signal strength:
  long rssi = WiFi.RSSI();
  Serial.print("signal strength (RSSI):");
  Serial.print(rssi);
  Serial.println(" dBm");
}

// Display values on Serial Port
void displayValuesOnSerial()
{
  Serial.print(" Requesting temperatures...");
  sensors.requestTemperatures(); // Send the command to get temperature readings
  delay(1000);
  Serial.println("DONE");
  Serial.print("Temperature is: ");
  Serial.print(sensors.getTempCByIndex(0)); // Why "byIndex"?
  // You can have more than one DS18B20 on the same bus.
  // 0 refers to the first IC on the wire
  delay(1000);
}

```

più limitata, possibile solamente tra un pacchetto gratuito ed uno a pagamento; per contro il piano gratuito è già di per sé molto valido: si possono inviare fino a 10 feed, cioè grandezze monitorabili, archiviati per 30 giorni, con una frequenza di aggiornamento complessiva massima di 30 dati al minuto. Il piano a pagamento, oltre a raddoppiare sia il tempo di archiviazione, ora pari a 60 giorni, sia il numero di data points inviati al minuto (in questo

caso 60), permette di creare un numero infinito di feed, per cui possiamo considerare giustificato e proporzionato il prezzo di 99 dollari all'anno (**Fig. 2**). Sarebbe stato molto apprezzabile per i maker anche avere a disposizione dei piani intermedi per quanto riteniamo il piano base ottimo per fare dei test e per gestire in maniera gratuita i propri progetti nel cloud. Anche in questo caso vediamo un esempio (**Listato 2**) per connettere nel cloud un arduino MKR1010 WIFI, connesso ad un sensore DS18B20. Il codice si presenta piuttosto semplice, anche se più articolato; la maggiore lunghezza è essenzialmente legata al fatto che in questo caso si sono inserite le funzioni per collegarsi al WIFI e per visualizzare lo stato della connessione con i relativi dettagli, cosa che in precedenza era "nascosta" dalle funzionalità del cloud che inseriva automaticamente le funzioni necessarie per la connessione ad esso. Vediamo che a differenza del cloud di Arduino, viene inclusa la libreria "ArduinoJson", utilizzata per la corretta formattazione dei dati inviati ad Adafruit IO nella funzione "void httpRequest()", invio che avviene in maniera sicura, utilizzando la chiave "IO_KEY" generata durante la fase di creazione del proprio

Our pricing is as simple as our API.

Try Adafruit IO for free. Unlock its full potential for \$10 per month.

Get Started	Power Up
FREE forever	\$10 or \$99 per month or per year
30 data points per minute 30 days of data storage Triggers every 15 minutes 5 dashboard limit 2 WipperSnapper device limit 5 group limit 10 feed limit	60 data points per minute 60 days of data storage Triggers every 5 seconds Unlimited dashboards Unlimited WipperSnapper devices Unlimited groups Unlimited feeds Learn more about IO+
Sign Up Now	Sign Up Now

Fig. 2
Piani offerti da
Adafruit per il suo
servizio cloud.

License Options

ThingSpeak is available as a free service for non-commercial small projects (<3 million messages/year or ~8,200 messages/day). For larger projects or commercial applications, four different annual license types are offered: Standard, Academic, Student and Home. ThingSpeak is bought in units, where one unit allows 33 million messages to be processed and stored in a one-year period (~90,000 messages/day). One unit also provides the ability to create a fixed number of channels on ThingSpeak. Choose your license type below to learn more.

Standard

For use at a commercial, government, or other organization.

[Learn More](#)

Academic

For use in teaching and academic research at a degree-granting institution.

[Learn More](#)

Student

For use in conjunction with courses offered at a degree-granting institution.

[Learn More](#)

Home

For personal use only. Not for government, academic, commercial, or other organizational use.

[Learn More](#)

Fig. 3
L'offerta di ThingSpeak è stata pensata per un vasto pubblico, comprendente, oltre ai makers, anche le istituzioni accademiche e le utenze professionali.

account, da custodire con cura e da non divulgare, presente all'interno di "arduino_secrets.h", che contiene SSID, password della propria connessione

WIFI, l'"IO_USERNAME" con cui vi siete registrati e l'"IO_GROUP" di cui fanno parte le grandezze che volete monitorare da remoto.

	FREE For time-limited commercial evaluation of the service	STANDARD For all commercial, government and revenue generating activities
Scalable for larger projects	✗ No. Annual usage is capped.	✓
Number of messages	3 million/year (~8,200/day) ⁽²⁾	33 million/year per unit (~90,000/day per unit) ⁽²⁾
Message update interval limit	Every 15 seconds	Every second
Number of channels	4	250 per unit
MATLAB Compute Timeout	20 seconds	60 seconds
Private channel sharing	Limited to 3 shares	Unlimited
Technical Support	Community Support	Standard MathWorks support
	FREE For small non-commercial projects	HOME For personal use only ⁽¹⁾
Scalable for larger projects	✗ No. Annual usage is capped.	✓
Number of messages	3 million/year (~8,200/day) ⁽²⁾	33 million/year per unit (~90,000/day per unit) ⁽²⁾
Message update interval limit	Every 15 seconds	Every second
Number of channels	4	10 per unit
MATLAB Compute Timeout	20 seconds	20 seconds
Private channel sharing	Limited to 3 shares	Unlimited
Technical Support	Community Support	Community Support

Fig. 4
Confronto fra le versioni "free", "home" e "standard" di ThingSpeak.

↓ Listato 3

```
/*
  WriteVoltage

  Reads an analog voltage from pin 0, and writes it to a channel on ThingSpeak every 20 seconds.

  ThingSpeak ( https://www.thingspeak.com ) is an analytic IoT platform service that allows you
  to aggregate, visualize and
  analyze live data streams in the cloud.

  Copyright 2017, The MathWorks, Inc.

  Documentation for the ThingSpeak Communication Library for Arduino is in the extras/documentation
  folder where the library was installed.
  See the accompanying licence file for licensing information.
*/

#include "FishinoThingSpeak.h"

// *****
// This example selects the correct library to use based on the board selected under
// the Tools menu in the IDE.
// Yun, Ethernet shield, WiFi101 shield, esp8266, and MXR1000 are all supported.
// With Yun, the default is that you're using the Ethernet connection.
// If you're using a wi-fi 101 or ethernet shield (http://www.arduino.cc/en/Main/ArduinoWiFiShield),
// uncomment the corresponding line below
// *****

#define USE_WIFI101_SHIELD
#define USE_ETHERNET_SHIELD

#if !defined(USE_WIFI101_SHIELD) && !defined(USE_ETHERNET_SHIELD) && !defined(ARDUINO_SAMD_MKR1000) && \
    !defined(ARDUINO_AVR_YUN) && !defined(ARDUINO_ARCH_ESP8266) && !defined(FISHINO_WIFI)
#error "Uncomment the #define for either USE_WIFI101_SHIELD or USE_ETHERNET_SHIELD"
#endif

#if defined(ARDUINO_AVR_YUN)
#include "YunClient.h"
YunClient client;
#elif defined(FISHINO_WIFI)
#include <Fishino.h>
char ssid[] = "<YOURNETWORK>"; // your network SSID (name)
char pass[] = "<YOURPASSWORD>"; // your network password
int status = STATION_IDLE;
FishinoClient client;
#elif defined(USE_WIFI101_SHIELD) || defined(ARDUINO_SAMD_MKR1000) || defined(ARDUINO_ARCH_ESP8266)
// Use WiFi
#ifdef ARDUINO_ARCH_ESP8266
#include <ESP8266WiFi.h>
#else
#include <SPI.h>
#include <WiFi101.h>
#endif
char ssid[] = "<YOURNETWORK>"; // your network SSID (name)
char pass[] = "<YOURPASSWORD>"; // your network password
int status = WL_IDLE_STATUS;
WiFiClient client;
#elif defined(USE_ETHERNET_SHIELD)
// Use wired ethernet shield
#include <SPI.h>
#include <Ethernet.h>
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
EthernetClient client;
#endif

#ifdef ARDUINO_ARCH_AVR
// On Arduino: 0 - 1023 maps to 0 - 5 volts
#define VOLTAGE_MAX 5.0
#define VOLTAGE_MAXCOUNTS 1023.0
#elif defined(FISHINO_PIC32_)
// On FISHINO32: 0 - 1023 maps to 0 - 3.3 volts
#define VOLTAGE_MAX 3.3
#define VOLTAGE_MAXCOUNTS 1023.0
#elif ARDUINO_SAMD_MKR1000
// On MKR1000: 0 - 1023 maps to 0 - 3.3 volts
#define VOLTAGE_MAX 3.3
#define VOLTAGE_MAXCOUNTS 1023.0
#elif ARDUINO_SAM_DUE
// On Due: 0 - 1023 maps to 0 - 3.3 volts
#define VOLTAGE_MAX 3.3
```

```

#define VOLTAGE_MAXCOUNTS 1023.0
#elif ARDUINO_ARCH_ESP8266
// On ESP8266: 0 - 1023 maps to 0 - 1 volts
#define VOLTAGE_MAX 1.0
#define VOLTAGE_MAXCOUNTS 1023.0
#endif

/*
***** Visit https://www.thingspeak.com to sign up for a free account and create
***** a channel. The video tutorial http://community.thingspeak.com/tutorials/thingspeak-channels/
***** has more information. You need to change this to your channel, and your write API key
***** IF YOU SHARE YOUR CODE WITH OTHERS, MAKE SURE YOU REMOVE YOUR WRITE API KEY!!
*****
*/

unsigned long myChannelNumber = 31461;
const char * myWriteAPIKey = "LD79EOAAWRVYF04Y";

void setup()
{
#ifdef ARDUINO_AVR_YUN
  Bridge.begin();
#elif defined(FISHINO_WIFI)
  Fishino.begin(ssid, pass);
#elif defined(ARDUINO_ARCH_ESP8266) || defined(USE_WIFI101_SHIELD) || defined(ARDUINO_SAMD_MKR1000)
  WiFi.begin(ssid, pass);
#else
  Ethernet.begin(mac);
#endif

  ThingSpeak.begin(client);
}

void loop()
{
  // read the input on analog pin 0:
  int sensorValue = analogRead(A0);
  // Convert the analog reading
  // On Uno,Mega,YunArduino: 0 - 1023 maps to 0 - 5 volts
  // On ESP8266: 0 - 1023 maps to 0 - 1 volts
  // On MKR1000, Due: 0 - 4095 maps to 0 - 3.3 volts
  float voltage = sensorValue * (VOLTAGE_MAX / VOLTAGE_MAXCOUNTS);

  // Write to ThingSpeak. There are up to 8 fields in a channel, allowing you to store up to 8 different
  // pieces of information in a channel. Here, we write to field 1.
  ThingSpeak.writeField(myChannelNumber, 1, voltage, myWriteAPIKey);
  delay(20000); // ThingSpeak will only accept updates every 15 seconds.
}

```

THINGSPEAK

Vediamo infine l'offerta di ThingSpeak, un servizio cloud IoT che permette un'analisi approfondita (e facoltativa) dei dati, utilizzando i potenti strumenti disponibili mediante l'uso del software matlab, il cui possesso e utilizzo non è comunque necessario per un normale uso. Anche in questo caso, come per l'offerta di Arduino, è possibile scegliere fra vari profili di utenza: si spazia dalla versione "home" che include la possibilità di un utilizzo gratuito, passando per una versione per studenti, una per le istituzioni accademiche ed infine una versione "standard" per utilizzo commerciale (**Fig. 3**).

La versione gratuita permette di scambiare un numero di messaggi 10 volte inferiore rispetto a quella "standard" e "home" (3 milioni all'anno, rispetto a 33 milioni), limitandone l'invio ogni 15 secondi, con-

tro l'aggiornamento possibile ogni secondo per le versioni a pagamento, "home" compresa. Per quanto riguarda invece il numero delle grandezze che è possibile monitorare, la differenza è più marcata: si hanno a disposizione 4 "canali" per la versione "free", 10 per la versione "home" e 250 per quella "standard" (**Fig. 4**).

A differenza delle offerte viste in precedenza, ThingSpeak risulta essere quella con i costi più "complicati" da definire, tanto che per il calcolo viene messo a disposizione un apposito tool in cui si inseriscono tutte le caratteristiche desiderate per il servizio scelto (**Fig. 5**).

Anche in questo caso vediamo un semplice esempio che mostra come sia possibile interfacciarsi con questo cloud. Utilizziamo l'esempio "WriteVoltage" (**Listato 3**), disponibile nella cartella "ThingSpeak"



Fig. 5
Tool per definire il
costo del servizio
ThingSpeak.

Pricing calculator

How many channels?

Currently: To be added:

Calculated number of channels needed: 10

How often will they collect data? Every Seconds

Calculated Message Use**

31,536,000/year
2,592,000/month
86,400/day
3,600/hour
60/minute

Our Recommendation:

Thingspeak units: 1
License type: Home
Maximum number of channels allowed: 10

* Typically one channel per device
** Assumes constant message rate

Purchase

License type: Home

ThingSpeak units:

x USD 95.00
price/unit/year

Total: USD 95.00/year

[Purchase](#)

You will be taken to the MathWorks store to complete your purchase.

Questions about ThingSpeak purchasing?

[Licensing FAQ](#) [Contact us](#)

della libreria di fishino, scaricabile direttamente dal sito www.fishino.it
In questo sketch viene letto il valore analogico presente all'ingresso del canale 0 dell'ADC. Le righe di codice che permettono la "customizzazione" al cloud sono le seguenti:

- `"#include "FishinoThingSpeak.h"` per includere la libreria scritta specificatamente per far comunicare le schede della famiglia fishino con ThingSpeak;
- `"unsigned long myChannelNumber = 31461;"` `"const char * myWriteAPIKey = "LD79EOAAWRVYF04Y";"` che sono i valori inseriti nell'esempio ma che dovranno necessariamente essere sostituiti dal vostro numero di canale dalla vostra chiave, generati durante la registrazione al servizio;
- `"ThingSpeak.begin(client);"` posto all'interno del setup, inizializza la comunicazione con il cloud
- `"ThingSpeak.writeField(myChannelNumber, 1, voltage, myWriteAPIKey);"` è invece all'interno del loop e permette l'invio vero e proprio del valore misurato dall'ADC ed associato alla variabile "voltage".

CONCLUSIONE

La disponibilità di un gran numero di schede di prototipazione facilmente collegabili in rete, ha semplificato notevolmente l'accesso anche ai maker ai servizi IoT presenti nel cloud. Di questi, non tutti hanno caratteristiche e costi facilmente accessibili a chi vuole sperimentare l'invio dei dati acquisiti dai propri sensori in internet. Ne abbiamo analizzati tre, che consideriamo i più diffusi e adatti per entrare in questo affascinante mondo: Arduino IoT cloud, Adafruit IO e ThingSpeak, ognuno con i propri punti di forza e le proprie "debolezze". Tutti e tre offrono un servizio gratuito, senza limiti di tempo, per chi vuole iniziare a muovere i primi passi e non ha grandi esigenze pre-

stazionali. Tra questi, Adafruit ci è sembrato offrire il servizio migliore: senza alcun costo di gestione, è possibile monitorare fino a 10 grandezze e archiviare i dati fino a 30 giorni, contro le due grandezze di Arduino che ne memorizza i dati solamente per un giorno. In posizione intermedia si pone ThingSpeak che gratuitamente permette di gestire 4 "canali". Per contro Arduino è l'unico a fornire un'applicazione gratuita per android e iOS per monitorare e gestire il proprio cloud anche da smartphone. Per quanto riguarda la complessità e la personalizzazione, Adafruit ci è sembrata quella che offre il servizio più flessibile, sia perché non pone limite alla tipologia di schede di sviluppo utilizzate, sia perché si utilizza con il normale IDE di Arduino, senza rendere obbligatorio l'utilizzo dell'IDE presente nel cloud. Anche Arduino permette l'utilizzo di altre schede di prototipazione diverse dalle proprie ma obbliga comunque all'uso del proprio IDE nel cloud, cosa che all'inizio può essere poco intuitiva, perché si lavora in maniera interconnessa tra l'IDE e le altre sezioni di definizione della scheda e delle grandezze da inviare nel cloud. □

Cosa occorre?

La scheda MKR1010 (cod. MKRWIFI1010) è in vendita a Euro 39,00. Le board Fishino sono disponibili in diverse versioni nello store online di Futura Elettronica all'indirizzo www.futurashop.it

Il materiale va richiesto a:

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

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

HDMI
↓
VGA

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.

€ 11,50

Cod. **BOXHDMITOVGA**

Convertitori di segnale

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

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.

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.

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.

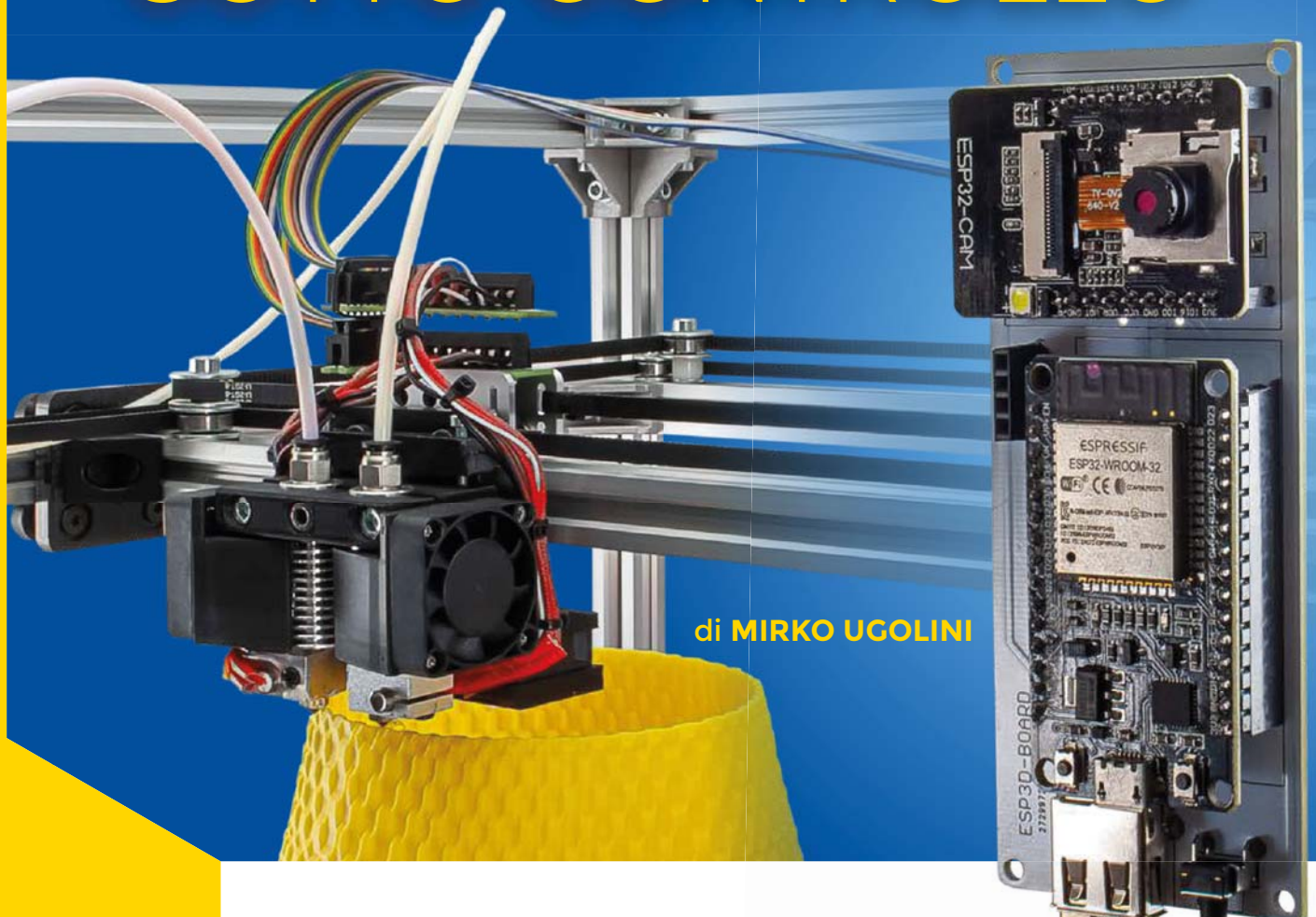
HDMI
↓
AV

€ 11,00

Cod. **BOXHDMITOAV**

WIRELESS

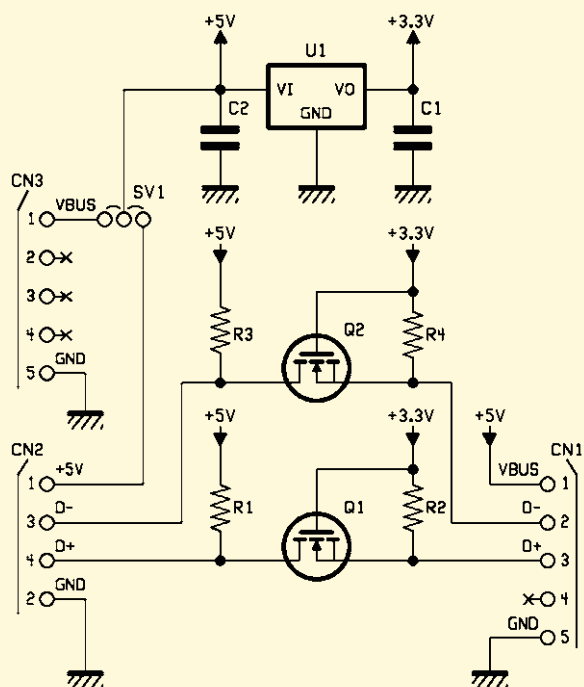
LA STAMPANTE 3D SOTTO CONTROLLO



di MIRKO UGOLINI

Introduciamo un progetto che ci permetterà di controllare la stampante 3D tramite ESP32. Prima puntata.

La vostra amata stampante 3D è installata nel garage di casa e siete stanchi di fare continuamente le scale per controllare il progresso della stampa lanciata, magari, la sera tardi? Desiderate poter osservare il progresso della stampa in tempo reale ed abortirla immediatamente nel caso vi rendiate conto che qualcosa stia andando per il verso sbagliato? Volete poter controllare da remoto i parametri di stampa o siete preoccupati nel mantenere la stampante sempre sotto tensione tutta la notte?



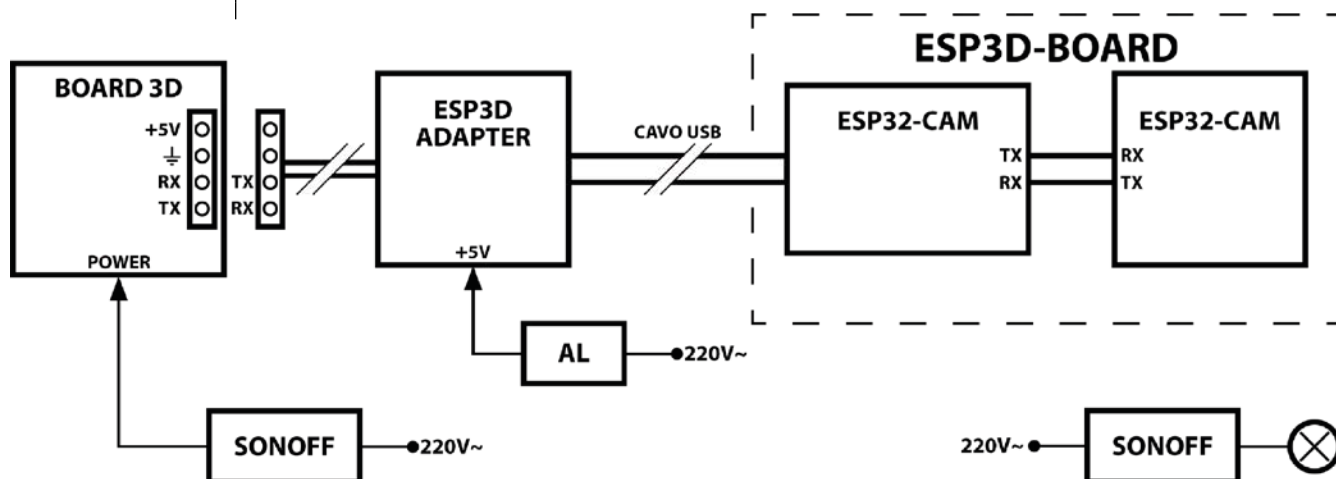
di caricamento del software e l'estrema fragilità del file-system del Raspberry Pi che in seguito ad accidentali perdite di tensione provoca la corruzione della micro-SD e richiede la reinstallazione dello stesso. Per non parlare del costo della stessa Raspberry Pi. Inoltre, alzi la mano chi si è avventurato con successo nella personalizzazione del software di Octoprint? Certo esistono i Plug-in, ma volete mettere la soddisfazione di avere il pieno controllo di quello che avete realizzato ed oltretutto impiegando il "linguaggio di programmazione" di Arduino?

IL MODULO ESP32

Siamo partiti da un progetto creato da **Luc Lebosse** (<https://github.com/luc-github/ESP3D>), ma poi ci siamo lasciati prendere la mano ed abbiamo realizzato un fork del suo ottimo lavoro, lo abbiamo espanso e personalizzato e l'entusiasmo per quello che vedevamo crescere sotto le nostre mani ci ha portati alla stesura dell'articolo seguente e di quelli che seguiranno. Questa prima puntata sarà principalmente introduttiva e vi metterà in condizioni di capire a grandi linee come funziona e nel contempo vi permetterà di utilizzarlo immediatamente nel caso in cui acquistiate i componenti necessari e vogliate aggiungerlo subito alla vostra stampante 3D. Nella seconda puntata andremo più a fondo nella descrizione delle varie componenti hardware e software del progetto e vi guideremo nella personalizzazione di eventuali modifiche che potreste ambire a realizzare. Uno schema a blocchi descrittivo si può vedere in **Fig. 1**. Il progetto comprende due PCB: una board principale denominata ESP3D-BOARD ed una scheda adattatore ESP3D-ADAPTER.

Vi piacerebbe ricevere una notifica che la stampa è terminata? Oppure che è terminato il filamento? Allora abbiamo il progetto che fa al caso vostro: non si tratta di qualcosa di rivoluzionario sotto l'aspetto del software, perché già esiste Octoprint, che fa tutto quello che abbiamo appena elencato; tuttavia Octoprint, che consente la gestione della stampante 3D da remoto attraverso la connessione WiFi, viene eseguito su una Raspberry Pi e quanti di voi hanno provato ad installarlo si sono accorti che l'utilizzo quotidiano non è molto pratico. Il problema principale è rappresentato dal tempo

Fig. 1
Schema a blocchi del sistema.



La ESP3D-BOARD, di cui vedete a pagina 4 lo schema elettrico, ospita un modulo ESP32-DEV, su cui gira la logica per gestire l'interfaccia web e la comunicazione con la stampante 3D ed un modulo ESP32-CAM che utilizzeremo per acquisire il video della stampa in tempo reale e che mostreremo nella interfaccia web. Il modulo ESP3D-ADAPTER viene utilizzato per consentire il collegamento della scheda ESP3D-BOARD con la scheda controller della stampante 3D. L'abbiamo progettata espressamente per inserirla nel connettore CON5 a 12 poli disposti su due file della nostra controller 3D. In **Fig. 2** è possibile vedere il connettore e la

relativa pin-out della scheda controller presentata nei numeri 255 e 256 della rivista. I segnali che ci interessano sono TXD, RXD, +5V e GND. Teniamo a mente che la scheda ESP3D-BOARD utilizza dei moduli ESP32 e quindi la logica dei segnali UART sono a 3,3V e non tollerano i 5V che sono invece presenti normalmente sulla UART delle board delle stampante 3D (ATMEGA2560).

Abbiamo quindi inserito nella scheda ESP3D-ADAPTER, un regolatore a 3,3V e due traslatori di livello, utilizzando dei MOSFET BS138 (siglati Q1 e Q2) ed alcuni resistori. Inoltre abbiamo previsto la possibilità alimentare la ESP3D-ADAPTER e quindi la ESP3D-BOARD tramite un classico alimentatore (5V 1A) per dispositivi "mobile" dotato di connettore microUSB TypeB. La selezione della sorgente di alimentazione avviene tramite il Jumper JP1: nella posizione 1-2 si utilizza la tensione presente sul pin 1 del connettore CN5, nella posizione 2-3 si utilizza invece la tensione proveniente dall'eventuale alimentatore USB esterno.

Il collegamento tra ESP3D-ADAPTER e ESP3D-BOARD avviene tramite un cavo USB, collegando quest'ultimo ai rispettivi connettori presenti sulle due board. Fate attenzione perché si tratta di connettori USB, ma non trasportano segnali secondo il protocollo USB, bensì i segnali RX, TX, GND, 5V; abbiamo adottato questa tipologia di cavo perché è schermato, i connettori sono molto robusti e facili da saldare sul PCB.

Come potete vedere dallo schema a blocchi, abbiamo inserito (opzionalmente) due moduli SonOFF (cod. FR710 dal sito futurashop.it) che possono essere controllati dalla interfaccia grafica di ESP3D. Un SonOFF viene impiegato per controllare l'accensione e spegnimento di una lampada utilizzata per illuminare l'area di stampa mentre si effettua la ripresa video, l'altro modulo SonOFF viene invece utilizzato per controllare l'alimentazione di tutta la stampante. Quindi è possibile al termine della fase

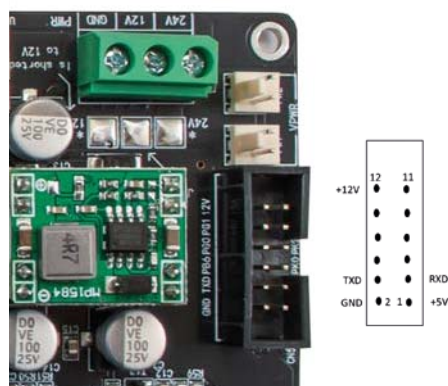
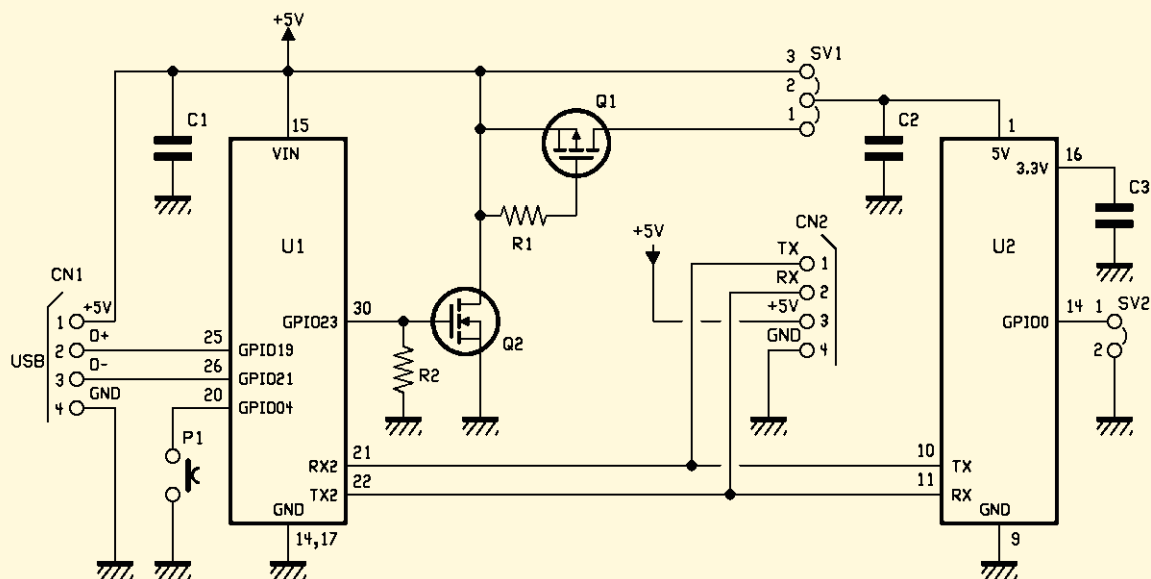


Fig. 2
Il connettore
di espansione.

di stampa spegnere completamente la stampante 3D. Inoltre (quando si alimenta la ESP3D-BOARD tramite l'alimentatore 5V USB esterno) è possibile accendere la stampante da remoto per esempio per preriscaldare il piatto e gli estrusori, e nel caso in cui il piatto sia libero dalla stampa precedente, lanciare anche una nuova sessione di stampa. Per quanto riguarda la camera, abbiamo utilizzato il modulo ESP32-CAM. Noterete dallo schema a blocchi che esiste una insolita comunicazione tramite UART tra il modulo ESP32-DEV ed il modulo ESP32-CAM. Questa comunicazione permette di evitare la configurazione manuale del WiFi per entrambi i moduli. Il funzionamento è molto semplice. Configureremo per l'accesso alla nostra rete WiFi solo il modulo ESP32-DEV al quale dovremo fornire SSID e password.

Al momento del Boot, il modulo ESP32-CAM richiede tramite seriale questi parametri al modulo ESP32-DEV e quando li ha ottenuti entrambi cerca di collegarsi alla rete WiFi. Se la connessione alla rete WiFi si conclude con successo, il modulo ESP32-CAM, comunica al modulo ESP32-DEV l'indirizzo IP acquisito e quindi nella interfaccia web del modulo ESP32-DEV avremo la possibilità di visualizzare direttamente lo "stream" video inquadrato dalla camera senza impazzire per trovare quale IP il nostro router WiFi abbia assegnato al modulo ESP32-CAM.

Il dispositivo è dotato di un pulsante di reset (P1) che ci permetterà di ripristinare la configurazione di fabbrica e reimpostare i parametri della rete WiFi ripresentando il Wizard di setup iniziale. Questo è utile nel caso in cui si debba spostare la stampante su altra rete WiFi. Per resettare alla configurazione di fabbrica è necessario mantenere pigiato (durante il riavvio) il pulsante P1 fintantoché il LED su ESP32-DEV non esegua due rapidi lampeggi. La configurazione viene fatta una sola volta e viene salvata sul modulo ESP32-DEV, mentre ad ogni



riavvio il modulo ESP32-CAM richiede le credenziali di accesso al modulo ESP32-DEV.

CARICAMENTO DEI FIRMWARE PRECOMPILATI

Per agevolarvi nell'utilizzo immediato del progetto abbiamo pensato di rilasciare i precompilati dei

codici e permettervi di caricarli direttamente, senza dovervi preoccupare di installare tutte le librerie necessarie, ed effettuare il setup dell'IDE di Arduino. È Infatti importante che vengano utilizzate versioni particolari di alcune librerie ed anche impiegare il core del chip ESP32 nella versione corretta (1.0.4). Abbiamo notato infatti che utilizzando la versione

```

C:\WINDOWS\system32\cmd.exe

=====
FUTURA ELETTRONICA
=====
Programmazione Modulo ESP323D
=====

Si prega di attendere, sto cercando le porte seriali.
Ho trovato le seguenti porte:

CPNAME is Silicon Labs CP210x USB to UART
CPNUM is Bridge (COM5)

Inserire in numero della porta da utilizzare COM:5

Selezionare il BoudRate A:921600, B:115200, C:256000, D:230400, E:512000 or Default? A
La porta : COM5
Il Boudrate: 921600 Bit/sec

VUOI PROCEDERE A PROGRAMMARE IL MODULO ESP323D? [Y/N]:y

Pigiare il pulsante [BOOT] del dispositivo e rilasciarlo appena iniziano ad apparire messaggi in console

esptool.py v2.6
Serial port COM5
Connecting.....
Chip is ESP320WDQ6 (revision 1)
Features: WiFi, BT, Dual Core, 240MHz, VRef calibration in efuse, Coding Scheme None
MAC: 3c:71:bf:9e:35:cc
Uploading stub...
Running stub...
Stub running...
Changing baud rate to 921600
Changed.
Configuring flash size...
Auto-detected Flash size: 4MB
Compressed 8192 bytes to 47...
Wrote 8192 bytes (47 compressed) at 0x0000e000 in 0.0 seconds (effective 5957.7 kbit/s)...
Hash of data verified.
Compressed 15856 bytes to 10276...
Wrote 15856 bytes (10276 compressed) at 0x00001000 in 0.1 seconds (effective 991.0 kbit/s)...
Hash of data verified.
Compressed 1141872 bytes to 642031...
Writing at 0x00024000... (15 %)
    
```

Fig. 3
Lo script relativo
alla configurazione.

```
C:\WINDOWS\system32\cmd.exe

#####
#          FUTURA ELETTRONICA          #
#                                     #
#          Programmazione Modulo ESP32D      #
#                                     #
#####

Si prega di attendere, sto cercando le porte seriali.
CPNAME is USB-SERIAL CH340 (COM4)
CPNUM is t4

Inserire in numero della porta da utilizzare COM:4

Selezionare il BoudRate A:921600, B:115200, C:256000, D:230400 , E:512000 or Default? A

La porta : COM4
Il Boudrate: 921600 Bit/sec

VUOI PROCEDERE A PROGRAMMARE IL MODULO ESP32D? [Y/N]:y

Pigiare il pulsante [BOOT] del dispositivo e rilasciarlo appena iniziano ad apparire messaggi in console

esptool.py v2.6
Serial port COM4
Connecting.....
Chip is ESP32D0W0Q6 (revision 1)
Features: WiFi, BT, Dual Core, 240MHz, VRef calibration in efuse, Coding Scheme None
MAC: ac:67:b2:37:69:84
Uploading stub...
Running stub...
Stub running...
Changing baud rate to 921600
Changed.
Configuring flash size...
Auto-detected Flash size: 4MB
Compressed 8192 bytes to 47...
Wrote 8192 bytes (47 compressed) at 0x0000e000 in 0.0 seconds (effective 16384.2 kbit/s)...
Hash of data verified.
Compressed 15856 bytes to 10276...
Wrote 15856 bytes (10276 compressed) at 0x00001000 in 0.1 seconds (effective 953.7 kbit/s)...
Hash of data verified.
Compressed 772928 bytes to 445053...
Writing at 0x00034000... (35 %)
```

Fig. 4
Lo script
upload.bat in
esecuzione.

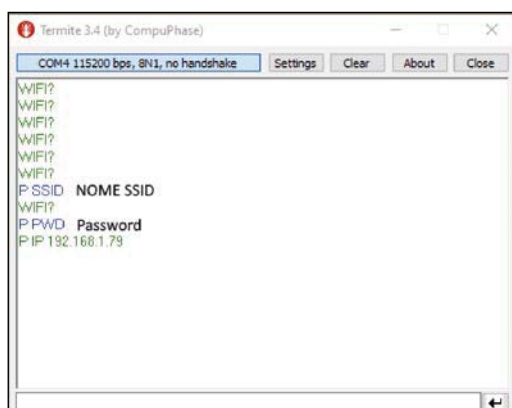
1.0.5 del core ESP32, insorgono dei problemi di compilazione. Analizzeremo in dettaglio il settaggio dell'ambiente di compilazione nel successivo articolo, dedicato alla comprensione del software ed alla sua personalizzazione. La cartella "Script Programmazione" contiene a sua volta le cartelle specifiche per i due moduli utilizzati. Per caricare il firmware precompilato, apriamo la cartella ESP32-DEV, colleghiamo con un cavo USB dati (attenzione, alcuni cavi che vengono usati per caricare i mobile sono solo cavi Power e quindi non sono adatti a questo scopo) il modulo ESP32-DEV ad una porta USB del PC. Nel momento in cui lo colleghiamo, dovremmo udire il classico segnale acustico associato al rilevamento di nuovo hardware, attendiamo il completamento della installazione degli eventuali driver. Completata l'installazione dei driver, individuiamo nella rispettiva cartella lo script **upload.bat**, facciamo doppio clic su esso, si dovrebbe aprire la schermata di **Fig. 3**. Lo script propone la lista delle porte COM trovate. Digitate il numero di quella corretta e continuate nella selezione delle varie opzioni fino al completamento della procedura.

Al termine della procedura verrà aperta un'istanza del monitor seriale **Termite**, per verificare dal log se il firmware è stato caricato in modo corretto.

ESP32-CAM

Ricordiamo che il modulo ESP32-CAM non è dotato a bordo di convertitore USB seriale per la programmazione. Quindi per poterlo programmare dovremo alloggiarlo sulla ESP3D-BOARD. Per alimentarlo e programmarlo, colleghiamo un convertitore USB TTL a 3,3V sul connettore CN2 (rispettando la piedinatura) ad una porta USB del nostro PC. Dovremo poi spostare il Jumper di SV1 sulla posizione 2-3, e mantenere chiuso SV2 con un Jumper. Anche in questo caso dovremmo udire il classico segnale acustico associato al rilevamento di nuovo hardware, attendiamo il completamento della installazione degli eventuali driver. Al termine, individuato nella rispettiva cartella lo script **upload.bat**, facciamo doppio clic su esso. Si dovrebbe aprire la schermata di **Fig. 4**. Terminata la procedura di caricamento del firmware non verrà lanciato automaticamente Termite, ma

Fig. 5
Test della rete WiFi.



sarà necessario rimuovere il Jumper su **SV2** e poi lanciare Termite manualmente eseguendo quello che si trova nella cartella ESP32-CAM. Per testare se il modulo della camera funziona in modo corretto possiamo inserire i dati di connessione della nostra rete WiFi sull'area di input di Termite (si trova in basso) e cliccare sul pulsante per l'invio (**Fig. 5**).

Notiamo infatti che il Modulo non avendo i parametri di collegamento alla rete WiFi, ogni tot secondi richiede tali parametri con la stringa *WiFi?* emessa sulla seriale. Nel funzionamento normale questa richiesta è indirizzata al modulo ESP32-DEV tramite UART, il quale essendo stato configurato tramite il Wizard (che poi vedremo) avrà già i dati memorizzati. Possiamo però simulare questa cosa fornendo i parametri manualmente tramite la console seriale di Termite.

Inseriamo il valore dell'SSID scrivendo *P SSID* NomeSSID poi clicchiamo sulla icona (Enter), ora inseriamo la password, scrivendo *P PWD* Password e nuovamente Enter.

Naturalmente NomeSSID e Password dovranno essere quelli validi della nostra rete WiFi. Se i dati sono corretti, dopo qualche secondo apparirà l'indirizzo IP acquisito dal nostro modulo ESP32-CAM. Nel nostro caso 192.168.1.79. Possiamo quindi aprire il browser all'indirizzo <http://192.168.1.79> ed apparirà lo streaming video di quello che la nostra camera sta inquadrando.

SETUP INIZIALE

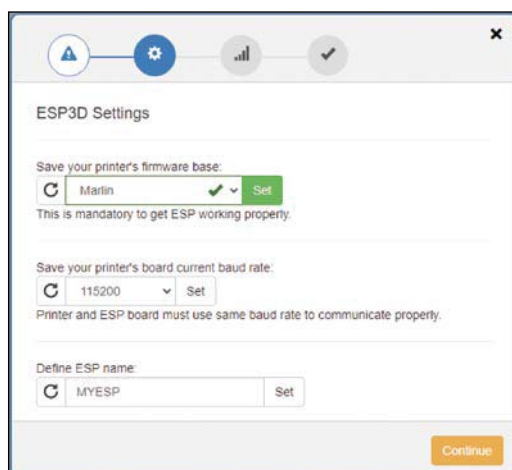
Avendo caricato il firmware su tutti i dispositivi interessati, possiamo procedere ad assemblare in modo definitivo il nostro modulo ESP3D-BOARD. Posizioniamo il modulo ESP-DEV ed il modulo ESP32-CAM nei rispettivi zoccoli, assicurandoci che i Jumper siano settati secondo lo schema seguente: SV1:1-2, SV2: OPEN. Per poter alimentare la ESP3D-BOARD dobbiamo quindi collegare il connettore USB CN1 al connettore USB del modulo ESP3D-ADAPTER tramite il cavo USB. Poi individuiamo sulla board della nostra stampante3D la posizione del connettore su cui innestare il connettore CN2 presente sulla ESP3D-ADAPTER.

Nel caso della nostra nuova MainBoard3D abbiamo previsto in fase di progettazione questo connettore CN5 (**Fig. 2**). Se invece siete dotati di una Motherboard 3D diversa, vi consigliamo di usare degli spezzoni di cavo e collegarli al connettore CN2.

Ricordatevi di posizionare in modo corretto il Jumper di selezione di alimentazione presente su ESP3D-ADAPTER, in questa fase possiamo anche limitarci ad utilizzare l'alimentazione proveniente dalla Board della stampante. Sinceratevi che il LED Power del modulo ESP32-DEV si accenda e rimanga acceso ad indicare la presenza della tensione di alimentazione.

Fatto questo, il secondo passo è quello di collegarsi all'hotspot WiFi implementato dal modulo ESP-DEV; questo perché in fase di configurazione il modulo ESP3D si comporta come un AccessPoint e genera un segnale WiFi al quale dovremo collegarci per poter poi configurare il resto dei parametri. Individuiamo quindi tra i vari SSID rilevabili da Windows, quello che ha il nome "ESP3D". Collegiamoci quindi allo stesso e terminata la fase di autenticazione e acquisizione dell'indirizzo IP, si aprirà automaticamente una finestra del browser che presenterà una interfaccia web di setup a step. La pagina avrà un indirizzo: <http://msftconnecttest.com> se aperto da un sistema Windows, oppure <http://captive.apple.com/hotspot-detect.html> se si

Fig. 6
Il tipo di firmware da caricare.



utilizza un computer iOS.

Se è la prima volta che eseguite la procedura e quindi il modulo ESP3D non è mai stato programmato apparirà la schermata in cui vi viene chiesto di caricare il file zippato che contiene l'interfaccia del nostro sistema di controllo. Il file da caricare è **Index.html.gz**. Ricerchiamo questo file nella cartella del nostro progetto e carichiamolo (upload) tramite l'interfaccia web.

Nel caso in cui abbiate necessità di resettare la configurazione, dovrete mantenere pigiato (durante il riavvio) il pulsante P1 fintantoché il LED su ESP32-DEV non esegua due rapidi lampeggi. Al termine del caricamento verrà visualizzata la pagina (Step1) di **Fig. 6**, nella quale dovremo specificare la tipologia di firmware caricato sulla nostra stampante 3D (Marlin, XXXX, ecc...), la velocità di comunicazione della seriale ed il nome con cui vogliamo identificare il nostro sistema di controllo, modifichiamo il nome proposto (MYESP) in ESP3D. Allo Step successivo (**Fig. 7**) dovremo scegliere se il modulo ESP3D debba emettere un proprio segnale WiFi (AP-Mode) e comportarsi quindi come un AccessPoint, oppure debba comportarsi da Client WiFi (Station-Mode) e collegarsi alla rete WiFi che abbiamo in casa. In questo caso dovremo fornirgli il nome SSID della nostra rete e la relativa

WiFi Configuration

Define ESP role:

Client Station ✓ Set

AP define access point / STA allows to join existing network

What access point ESP need to be connected to:

FRITZBOX_2.4GHz ✓ Set 🔍

You can use scan button, to list available access points.

Password to join access point:

XXXXXXXXXXXXX ✓ Set

Continue

Fig. 7
Impostazione
WiFi.

Password.

Per poter usufruire della videocamera è necessario scegliere la seconda opzione.

Attenzione quando si conferma il dato: dopo aver cliccato sul pulsante "Set", verificate che lo stesso diventi di colore verde. Nel terzo ed ultimo Step lo schermo ci ricorda che al termine della procedura il modulo si riavvierà e potremo configurare ulteriori dettagli relativi alla nostra configurazione utilizzando l'interfaccia completa. Al termine del setup ed in seguito al riavvio, dovremmo notare

MYESP

Non sicuro | 192.168.1.232

FUTURA ELETTRONICA

ESP3D Settings

Network Printer

Label	Value
Baud Rate	115200 Set
Sleep Mode	Modem Set
Web Port	80 Set
Data Port	8888 Set
Hostname	MYESP Set
WiFi mode	Client Station Set
Station SSID	FRITZBOX_2.4GHz Set 🔍
Station Password	Set
Station Network Mode	11g Set
Station IP Mode	DHCP Set
Station Static IP	192.168.0.1 Set

Fig. 8
In questa
schermata è
possibile settare
i parametri
principali del
modulo ESP3D.

prima un singolo lampeggio del LED sulla scheda ESP32-DEV che indica il termine della procedura di Setup, dopo una decina di secondi dovremmo vedere tre brevi lampeggi che indicano la corretta connessione con la rete WiFi.

A seguire dovremmo osservare un breve lampeggio del diodo FLASH sulla scheda ESP32-CAM, che indica il termine della procedura di setup e dopo una decina di secondi lo stesso led dovrebbe lampeggiare brevemente tre volte ad indicare la corretta connessione con la rete WiFi.

Terminato il setup, in seguito al riavvio del computer, potremo controllare la stampante 3D accedendo tramite browser (**Fig. 8**) da qualunque punto coperto dal segnale della nostra rete WiFi, utilizzando il seguente URL: <http://esp3d.local>.

Avrete sicuramente notato che l'interfaccia proposta nella **Fig. 9** è molto simile a quella di Octoprint e permette di avere il pieno controllo della stampante 3D. Le funzionalità di controllo sono raggruppate in quattro Tab principali ed inoltre avremo un menu ad Hamburger che permette di configurare l'interfaccia grafica ed alcune personalizzazioni avanzate.

INTERFACCIA GRAFICA

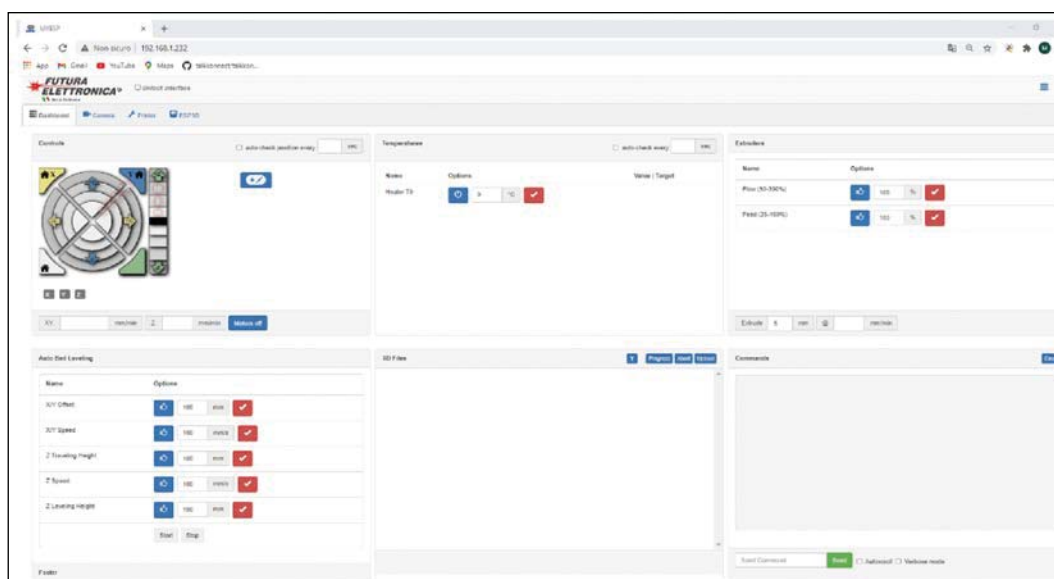
Come è possibile vedere dalla **Fig. 9** sono presenti quattro tab chiamate "**Dashboard**", "**Camera**", "**Printer**", "**ESP3D**". La Tab "**Dashboard**" è il vero e proprio quadro di comando che contiene tutti i principali pannelli di controllo o di visualizzazione utili alla gestione della stampante. I pannelli principali che abbiamo nella Dashboard

sono "**Controls**", "**Temperatures**", "**Extruders**", "**Auto Bed Leveling**", "**SD Files**", "**Commands**", alcuni di questi sono opzionali ed è possibile non visualizzarli selezionando le opportune spunte sulle opzioni del menu di configurazione presente nel menù ad Hamburger di destra.

Analizziamoli uno alla volta.

- **Controls.** Permette di controllare il movimento dell'estrusore tramite un controllo grafico molto efficiente che consente di spostare l'estrusore nelle varie coordinate, Home, X, Y e Z. Inoltre è possibile visionare in tempo reale la posizione dell'estrusore, e disattivare/attivare i motori cliccando sull'apposito pulsante. Cliccando sulla icona con la stella e la matita è possibile creare e salvare delle macro per eseguire, richiamando delle operazioni ripetitive.
- **Temperatures.** Questo pannello permette di impostare le temperature desiderate di Setpoint degli estrusori e del letto di stampa e poi verificarne l'andamento in tempo reale, tramite un grafico che riporta lo scostamento del valore corrente dal Setpoint impostato.
- **Extruders.** Questo pannello permette di controllare il **flow rate** e **Feed** degli estrusori.
- **Auto Bed Levelling** (Opzionale); questo pannello permette di controllare la modalità di auto livellamento del piatto di stampa.
- **SD Files.** Questo pannello permette di gestire i file formato **.GCO** caricati sullo spazio SPIFF del modulo ESP3D e che possono essere lanciati per eseguirne la stampa.
- **Commands;** permette di visualizzare i co-

Fig. 9
L'interfaccia di controllo con le relative tab è simile a quella di Octoprint.



mandi inviati alla Stampante 3D, quelli ricevuti dalla stampante, oppure consente di inviarne direttamente digitandolo nella input box "Send Commands" e poi cliccando sul pulsante "Send".

- **Camera** (Opzionale): permette di visualizzare lo stream video ripreso dalla camera a bordo del nostro dispositivo ESP32-CAM. È possibile inoltre accendere o spegnere la lampada che illumina la scena, stoppare la stampa (Panic Button), oppure spegnere completamente la stampante. La presenza di questo Tab è opzionale.
- **Printer**: permette di leggere e configurare tutti i parametri di settings della stampante e che di norma vengono salvati sulla EEPROM della board che controlla la stampante 3D. Naturalmente possono variare a seconda della stampante configurata.
- **ESP3D**: presenta cinque icone proposte nella **Fig. 10**.

La prima permette di visualizzare una lista riepilogativa in sola lettura di tutte le configurazioni del modulo di controllo ESP3D. La seconda permette di accedere al file-system del modulo ESP3D e permette di verificare i file caricati. La terza consente di controllare l'aggiornamento del firmware. La quarta permette di riavviare il modulo. Il quinto consente di aggiornare la pagina. Nella parte sottostante alle cinque icone è possibile selezionare due opzioni da una **radio button** "Network" o "Printer". La selezione **Network** (**Fig. 11**) apre un serie di opzioni tra cui troviamo, il Baud Rate della seriale, Il tipo di firmware della stampante, i parametri di accesso al WiFi e verso la fine la configurazione della modalità di notifica degli allarmi. Abbiamo tre possibilità per il momento, **E-mail**, **Lime** e **Pushover**. Nei prossimi articoli vedremo come configurarli ed adattarli alle nostre esigenze.

La selezione **Printer** apre una tab (**Fig. 12**) in cui si può specificare il tipo di firmware della stampante 3D, il fatto che il codice M117 sia attivo o meno, la modalità di comunicazione dei messaggi di Output. Come indicato, in questo progetto avremo quindi due PCB. Per quanto riguarda lo schema elettrico del modulo ESP3D-BOARD, notare la presenza dei due transistor MOS Q1 e Q2 che vengono impiegati per controllare l'alimentazione del modulo ESP32CAM tramite il segnale proveniente dal pin GPIO23. Quando questo pin viene portato alto, il MOS Q1 va in conduzione e quindi il suo drain va ad un livello di tensione basso e questo porta in conduzione il MOS Q2 e quindi il modulo ESP32-CAM viene alimentato.

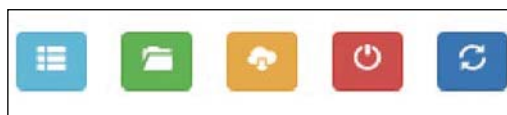


Fig. 10
Icône della
tab ESP3D.

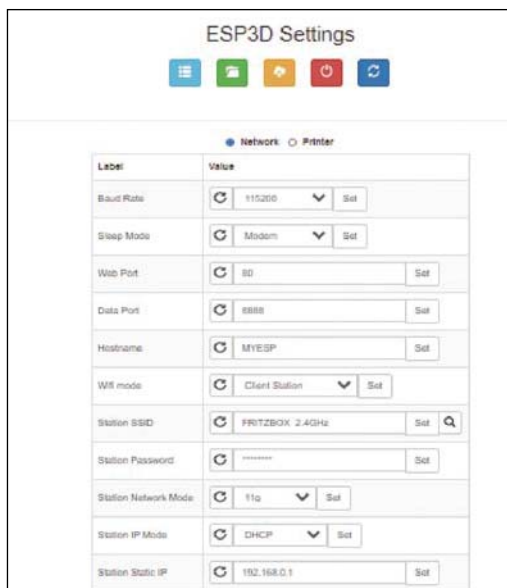


Fig. 11
La finestra
di dialogo
Network.

Altra cosa da notare è la comunicazione tramite UART dei due moduli. Quando si carica il firmware sul modulo ESP32-CAM è opportuno che il modulo ESP32-DEV non sia alloggiato sulla board ESP3D-BOARD per evitare che il collegamento tra le seriali provochi un malfunzionamento durante il caricamento del firmware visto che viene utiliz-

Fig. 12
Definizione del
firmware e di altri
parametri della
stampante 3D.

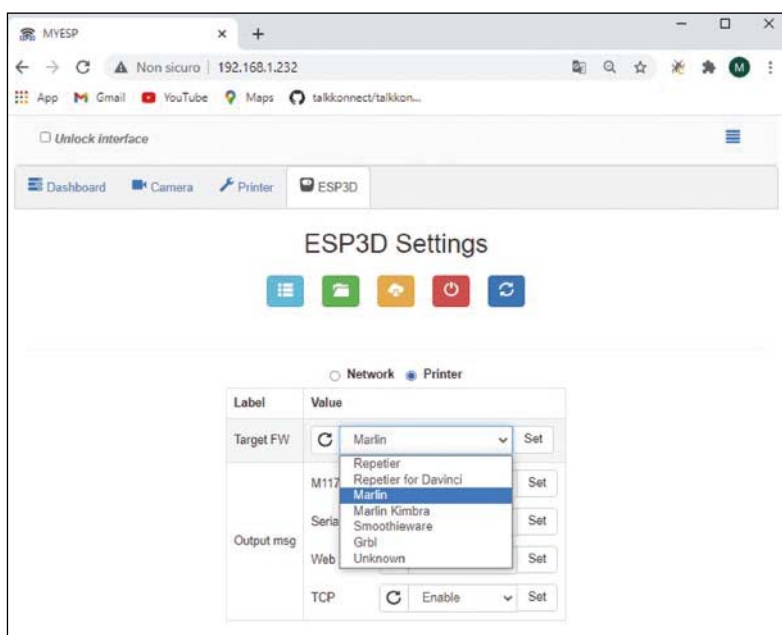
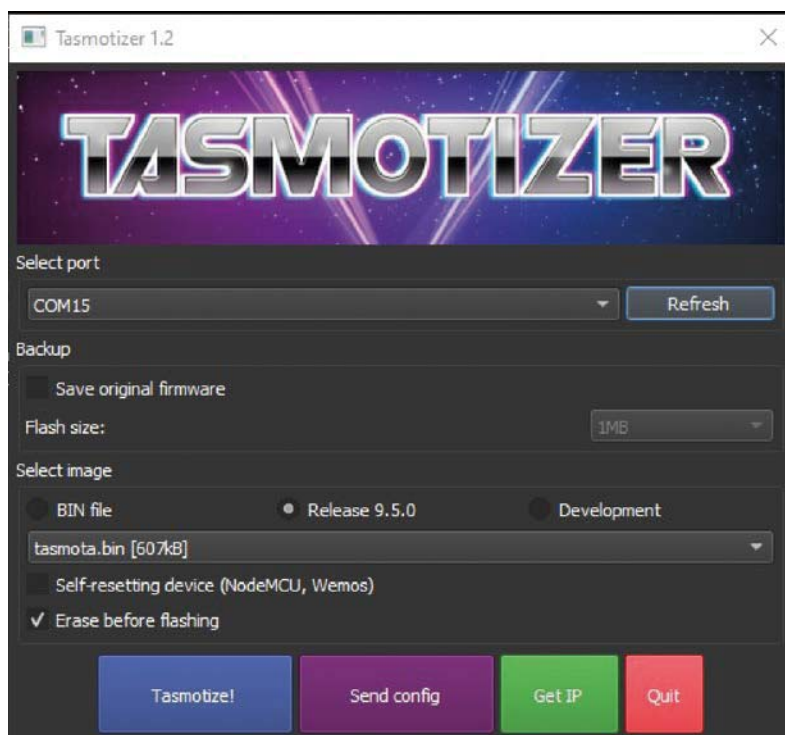




Fig. 13
Modulo
SonOFF.

zata la stessa seriale del modulo ESP32-CAM. Da notare la presenza anche dei due Jumper SV1 e SV2. SV1 bypassa il controllo della alimentazione tramite GPIO23 se configurato nella posizione 2-3 e quindi è da usare in questa posizione nella fase di programmazione del modulo ESP32-CAM. Mentre il jumper SV2 viene usato appunto in modalità chiuso durante la fase di programmazione e poi dovrà essere rimosso.

Fig. 14
Il Software
Tasmotizer.



Per quanto riguarda invece il modulo ESP32-ADAPTER, notare la presenza dei due traslatori di livello realizzati con dei MOS Q1 e Q2 e del regolatore lineare a 3.3v U1 che serve ad alimentare la parte a 3.3v dei due traslatori. Notare anche la presenza del Jumper SV2 da utilizzare per selezionare la provenienza della tensione di alimentazione. Posizione 1-2 esterna tramite alimentatore USB, posizione 2-3 interna tramite derivazione della tensione presente sullo zoccolo del connettore della stampante 3D.

MODULI SONOFF

I moduli SonOFF (**Fig. 13**) sono dei piccoli dispositivi domotici dotati di connettività WiFi che utilizzano il chip ESP8285 oppure ESP8266 e che permettono di controllare dei carichi collegati alla loro morsetti. Nel loro utilizzo normale sono da impiegare abbinati ad una APP (eWeLink) proprietaria ed una connessione al Cloud sempre proprietaria. Fortunatamente, una comunità di sviluppatori ha realizzato un firmware alternativo che si chiama Tasmota raggiungibile al seguente url <https://tasmota.github.io/docs/>. Utilizzando questo firmware alternativo è possibile controllare il dispositivo in modo completamente svincolato da software e infrastrutture proprietarie.

Uno dei moduli su cui è possibile installare Tasmota è Sonoff Basic R2 (quella di **Fig. 13**).

Per caricare Tasmota sul dispositivo è necessario seguire i seguenti passi:

1. Attenzione: Scollegare (se lo fosse) il cavo di alimentazione 220V !
2. Rimuovere le viti di chiusura del case e rimuoverne la parte superiore;
3. Individuare il connettore che riporta le diciture RX, TX , GND, 3.3V;
4. Saldare un connettore femmina a 4 poli con passo 2.54 mm;
5. Scaricare un software che si chiama Tasmotizer dal seguente url: <https://github.com/tasmota/tasmotizer>;
6. Collegare al PC un convertitore USB UART che abbia la possibilità di selezionare l'alimentazione di uscita a 3.3V. Attenzione questo è essenziale: i pin del chip ESPXXXX non sono 5V tollerant. Per esempio il modulo USBTOUART (disponibile dal sito futurashop.it) può essere utilizzato rispettando questi collegamenti: RX con TX , TX con RX , GND con GND e 3.3V con 3.3V;
7. Aprire il software Tasmotizer e se tutto funziona a dovere dovremmo vedere la porta COM

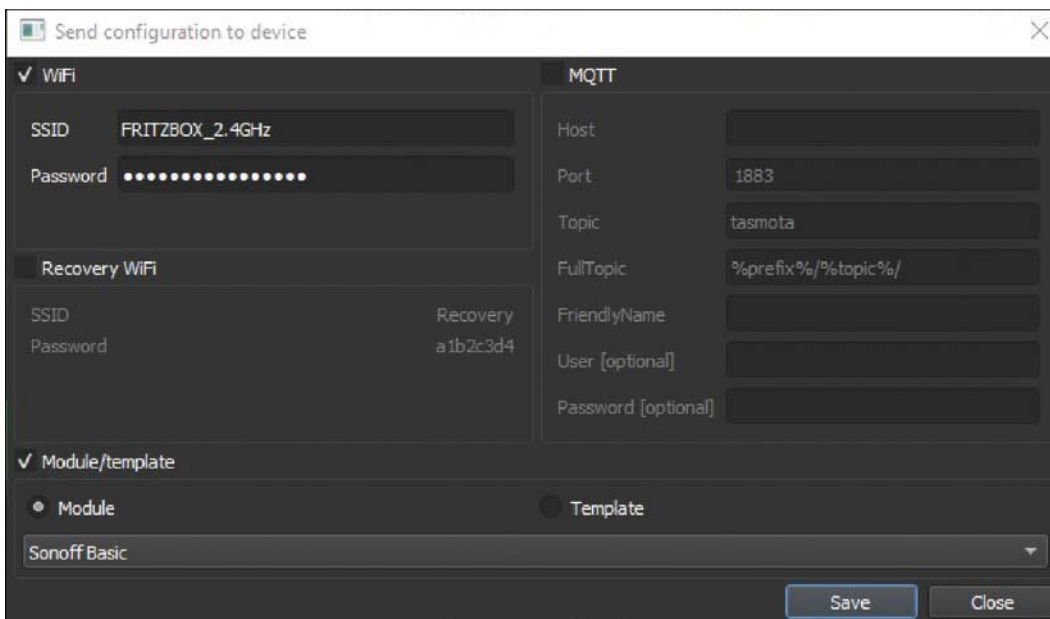


Fig. 15
Finestra di
configurazione
del dispositivo

attiva, selezionare le opzioni di **Fig. 14**. Dove si seleziona l'immagine selezionare l'ultima Release (al momento in cui scriviamo è la 9.5.0) ed infine clicchiamo sul pulsante "Tasmotizer". Se tutto funziona a dovere si aprirà una finestra di progressione. Al termine del caricamento si aprirà una finestra di configurazione del dispositivo (**Fig. 15**) in cui dovremo inserire le credenziali del nostro Router Wi-Fi e selezionare la tipologia di dispositivo Sonoff. Selezionate Sonoff Basic. Salvate la configurazione e se l'autenticazione avviene con successo, alla chiusura della form potremo (**Fig. 14**) sapere l'ip acquisito dal dispositivo cliccando sul pulsante "Get IP".

In **Fig. 16** è possibile vedere il cablaggio del Modulo SonOFF.



Attenzione al fatto che i contatti del relè non sono contatti puliti e quindi non è possibile usarli per interrompere un'alimentazione a basso voltaggio perché uno dei morsetti è in comune con la rete 220V. Vi ricordiamo che il carico massimo sul relè è di circa 10 A, ma non esagerate con il carico.

Terminata la configurazione è richiesto un ulteriore step. E' consigliabile fare in modo che il nostro dispositivo, ogni volta che si collega assuma sempre lo stesso indirizzo IP. Questo non è scontato: Alcuni router lo fanno di default, altri devono essere configurati.

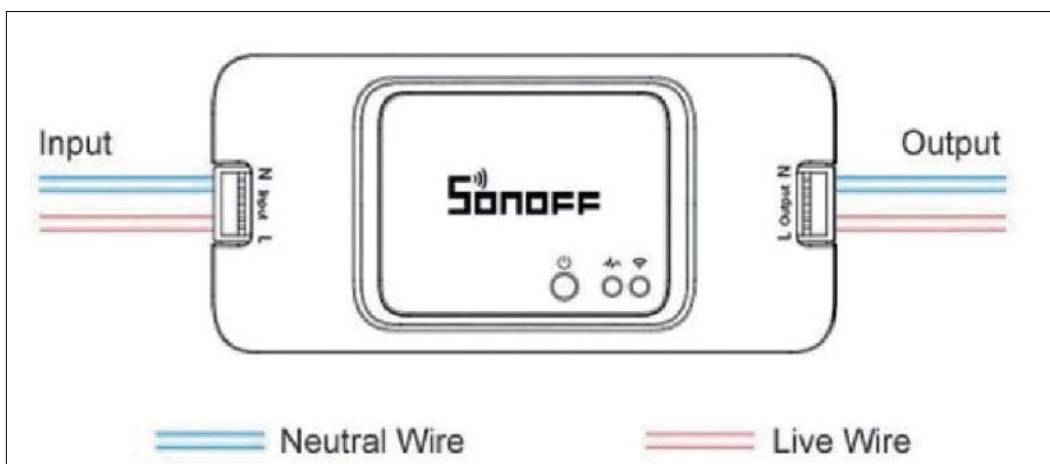


Fig. 16
Schema di
cablaggio
del modulo
Sonoff.



Il modulo WiFi con ESP montato sul controller della stampante 3D.

Per trovare l'indirizzo IP del dispositivo SonOff possiamo utilizzare un software gratuito al seguente url: <https://www.advanced-ip-scanner.com/it/> Ci servirà l'indirizzo IP assegnato ed il MAC address univoco del dispositivo, poi dovremo entrare nella pagina di configurazione del nostro router e fare in modo che al MAC address specifico sia sempre assegnato lo stesso IP.

TESTIAMO IL MODULO SONOFF

Per testare il modulo è sufficiente collegarsi al suo indirizzo IP (supponiamo che il nostro dispositivo abbia un indirizzo IP pari a 192.168.1.30) e per attivare la sua uscita dovremo aprire un browser nella nostra rete e digitare il seguente url:

<http://192.168.1.30/cm?cmd=Power%20On>

per spegnerla useremo il comando:

<http://192.168.1.30/cm?cmd=Power%20Off>

per conoscerne lo stato attuale useremo:

<http://192.168.1.30/cm?cmd=Power>

Se il comandi funzionano a dovere avremo, oltre all'effetto desiderato sul relè del dispositivo, anche una risposta in formato Json nella pagina del Browser.

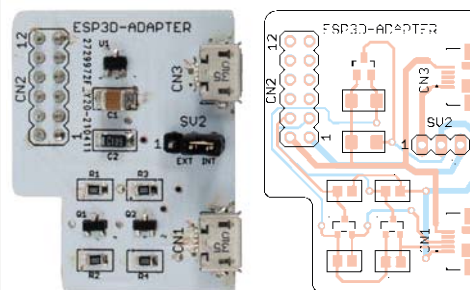
REALIZZAZIONE PRATICA

Passiamo ora alla costruzione del sistema, per la quale è necessario prima di tutto ottenere i PCB scaricando il file da www.elettronica.in.it. Potrete poi utilizzare il servizio di realizzazione PCB di Futura Elettronica ([*duzionePCB*\), oppure provvedere da voi, stampando le tracce su carta da lucido o acetato e ricorrendo alla fotoincisione, meglio se partendo da basette ramate presensibilizzate a doppia faccia.](https://www.futurashop.it/pro-</p>
</div>
<div data-bbox=)

I PCB sono a doppia ramatura ed essendo i componenti utilizzati di tipo SMD, la realizzazione pratica richiede un minimo di esperienza e una certa attrezzatura, vale a dire:

- una stazione saldante che abbia un potenza almeno di 40-50W;
- una punta che non è necessario sia sottilissima, ma che sia in buono stato (non ossidata) e che abbia la parte terminale corta, in modo da trasferire velocemente calore alla piazzola;
- del filo in lega di stagno (preferibile Sn-Ag) con flussante, da 0,5 mm o 1 mm;
- una pinzetta per prelevare e posizionare i componenti;
- della trecciola dissaldante (molto importante nel caso in cui vengano a crearsi dei ponticelli tra i pin più ravvicinati dei componenti);
- una lente di ingrandimento.

piano di montaggio ESP3D-ADAPTER

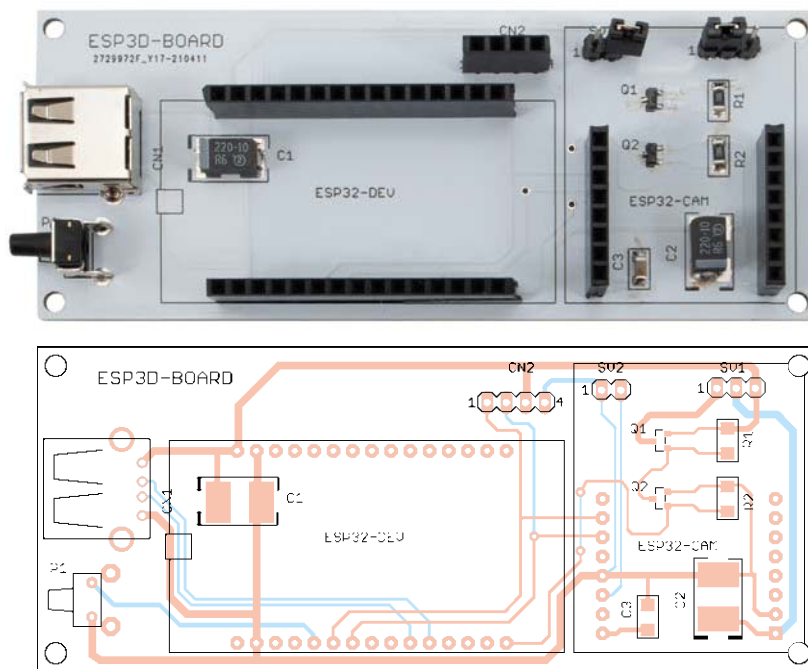


Elenco Componenti:

R1, R2, R3, R4: 10 kohm 5% SMD 1206
 C1: 10uF 10 V SMD 1206 (ceramico)
 C2: 100nF 50V SMD 1206 (ceramico)
 Q1,Q2: Mosefet Canale N BS138 SMD SOT23
 U1: Regolatore LDO MCP1700T-3302E/TT
 CN1, CN3: Connettore micro USB SMD
 CN2: Strip Femmina 4 poli
 SV1: Strip Maschio 3 poli
 - mini-jumper

Varie:
 stampato S1572 (37x28 mm)

piano di montaggio EESP32-CAM



Elenco Componenti:

- R1, R2: 10 kohm 5% SMD 1206
- C1, C2: 100uF 20 V SMD 2917
- C3: 100nF 50V SMD 1206 (ceramico)
- Q1: Moseft Canale P NTR4101P SMD SOT23
- Q2: Mosefet Canale N FDFV301N SMD SOT23
- P1: Pulsante NO a 90°
- CN1: Connettore Femmina USB-A
- CN2: Strip Femmina 4 poli
- SV1: Strip Maschio 3 poli
- SV2: Strip Maschio 2 poli
- Strip Femmina 8 poli
- Strip Femmina 5 poli
- mini-jumper
- ESP32-CAM
- ESP32-DEVKIT 30 pin

Varie
stampato S1572 (103x43 mm)

Per ultimo, ma non meno importante... tanta pazienza e la mano ferma. Facciamo adesso una nota sulla camera OV2640 presente sul modulo ESP32-CAM: normalmente essa è già collegata al connettore ed assicurata con un piccolo pezzetto di nastro biadesivo alla parte metallica dello slot miniSD. Per evitare di danneggiarla, non flettete mai in modo ripetuto il sottile connettore flessibile di cui è dotata e sinceratevi che sia fissata con il predetto nastro biadesivo. Al solito, iniziate il montaggio dai componenti a basso profilo e proseguite in ordine di altezza.

CONCLUSIONI

Bene, per il momento abbiamo terminato. Ovviamente abbiamo appena accennato alle possibilità di configurazione e di personalizzazione del modulo ESP3D, ma con ciò dovreste essere in grado già da subito di utilizzarlo per controllare la vostra stampante3D. Nella prossima puntata di questo articolo esporremo il codice che vi occorre e vi spiegheremo nel dettaglio come personalizzarne l'interfaccia grafica e modificare il sorgente ed eventualmente aggiungere delle nuove caratteri-

stiche. Sappiate comunque che il codice e le librerie per l'ambiente Arduino utilizzati nel progetto, sono disponibili al seguente URL di Github: <https://github.com/open-electronics/ESP3D> ed anche sul nostro sito www.elettronica.in.it. Per ora vi lasciamo, augurandovi buon lavoro e rimandandovi alla prossima puntata. ■



Cosa occorre?

Il materiale principale utilizzato in questo articolo è disponibile presso Futura Elettronica. La board di controllo per stampanti 3D dual extruder completa di driver (cod. FT1370M) è disponibile a Euro 119,00, la board ESP32 con telecamera (cod. ESP32-CAM) è in vendita a Euro 16,50, l'interruttore smart Wi-Fi (cod. FR710) è disponibile a Euro 15,90. 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>**

CATTURA L'ENERGIA DEL FUTURO

L'energia pulita che ti fa risparmiare!

Pannello solare monocristallino semirigido 12V



Pannello solare monocristallino rigido 12V



Pannelli solari monocristallini

Sono composti da singoli e sottilissimi strati di cristallo di silicio che opportunamente trattato aumenta la loro conduzione. Le celle che compongono il pannello monocristallino sono facilmente riconoscibili grazie al loro colore molto scuro. La purezza del silicio che compone le celle monocristalline permette di ottenere la massima efficienza energetica e grazie a questa caratteristica è possibile installare un numero di pannelli inferiore rispetto ai tradizionali, permettendo così di occupare superfici meno estese.

L'alta qualità del materiale utilizzato permette di avere una durata maggiore del pannello.

E per dare la giusta corrente alle tue batterie...

REGOLATORI DI CARICA PER PANNELLI FOTOVOLTAICI A 12 E 24V



cod. PWMSOLAR30
€ 39,00

30A



cod. PWMSOLAR20
€ 22,00

20A



cod. FR699
€ 99,00

15A

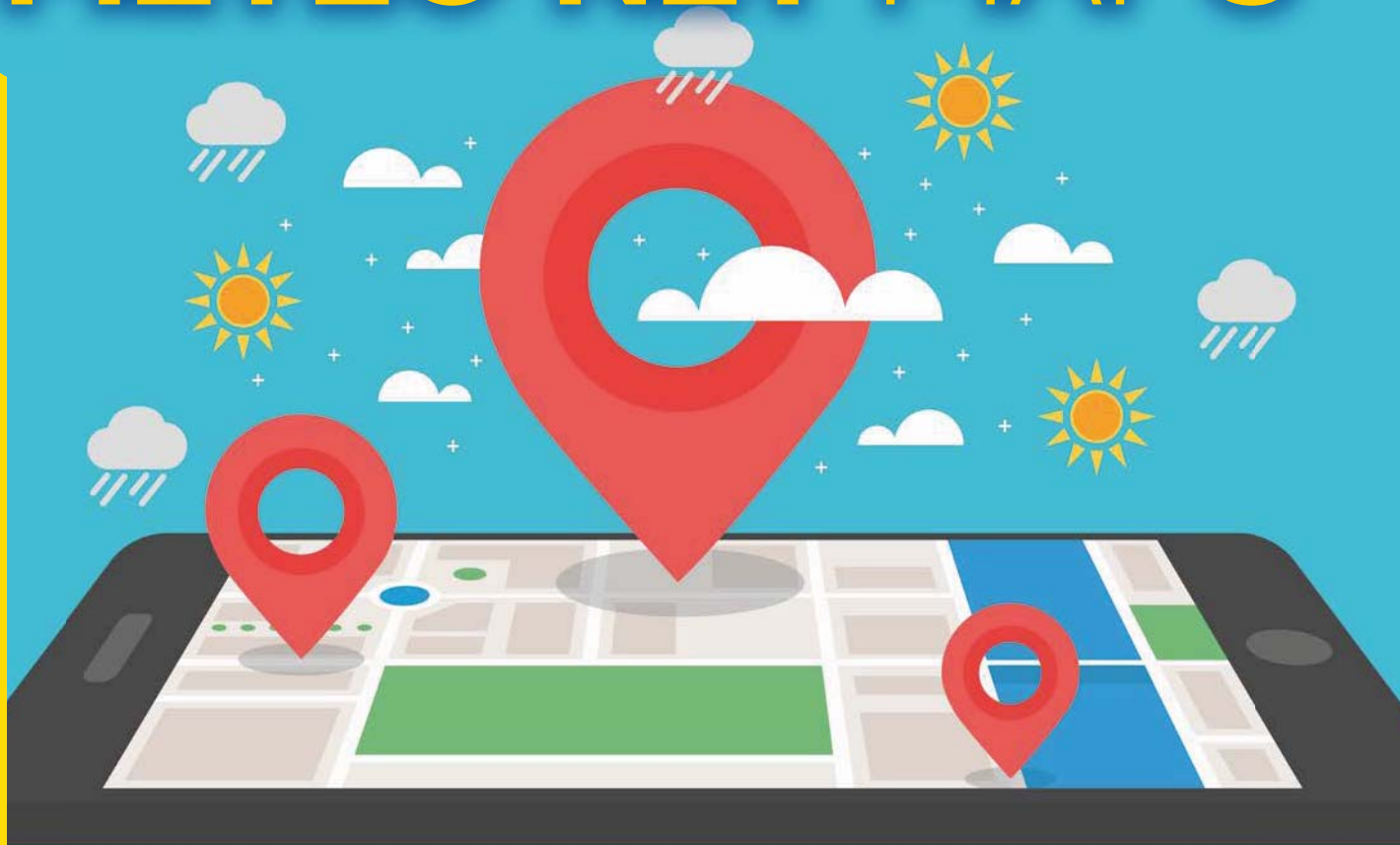


cod. SOLAREG10
€ 16,00

10A

Forniscono alle batterie la quantità di corrente necessaria, proteggendole da un eccesso di carica o scarica.

METEO NET MAPS

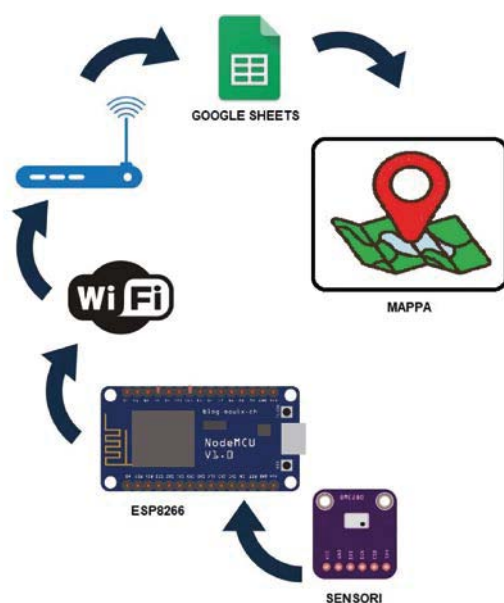


dell'ing. MIRCO SEGATELLO

Permette di geolocalizzare in un'unica mappa più dispositivi IoT e di visualizzare i parametri ambientali rilevati da ciascuno di essi.

Eccoci nuovamente a parlare di dispositivi IoT e mappe: lo facciamo riprendendo l'articolo pubblicato nel numero 259 della nostra rivista dedicato a MeteoNetMap. In quell'articolo avevamo visto come utilizzare un modulo NodeMCU (con a bordo un microcontrollore ESP8266) per inviare dati ambientali verso un foglio di calcolo online di Google. La cosa di particolare interesse di quel progetto era che più dispositivi potevano interagire con il foglio di calcolo, popolando ciascuno con i propri dati da collocare ciascuno in righe diverse. Il foglio di calcolo era poi disponibile per ulteriori elaborazioni e comunque visibile da qualsiasi web browser. Ma non ci siamo limitati a questo, perché il passo successivo è stato mettere a disposizione questi dati in una mappa che potesse visualizzare la posizione dei vari dispositivi e, a richiesta, visualizzare gli ultimi dati acquisiti. Il nome stesso del progetto -MeteoNetMap- indica che il principale utilizzo di questo sistema è la gestione di più

Fig. 1
Rappresentazione
schematica del
sistema.



stazioni meteo distribuite nel territorio. Sempre nell'articolo del MeteoNetMap avevamo messo in evidenza due criticità: la prima era riguardante l'accesso dei dispositivi alla rete, mentre la seconda era dovuta alla gestione dei livelli di sicurezza di Google: nel primo caso, siccome i microcontrollori ESP8266 dispongono solo di un modulo WiFi, il sistema proposto richiedeva che ciascun dispositivo potesse accedere ad una rete WiFi. La seconda criticità nasce dal fatto che affinché ciascun dispositivo potesse scrivere direttamente in un foglio di calcolo della Google, era necessario scrivere e rendere attivo un apposito script: un'operazione non molto agevole. In questo articolo vi spiegheremo come abbiamo risolto queste criticità, offrendo nuovi spunti allo sviluppo del progetto, la cui impostazione di sistema rimane sostanzialmente la stessa, ed è rappresentata in **Fig. 1**.

GESTIONE GOOGLE SHEETS

Per la prima parte abbiamo sfruttato i servizi gratuiti di Google, sempre molto affidabili e di facile utilizzo. Per prima cosa dovremmo poter disporre di un account Google e poter accedere all'applicativo "fogli" che è la versione web del foglio di calcolo e svolge le funzioni di Excel della Microsoft, noi lo useremo come database per salvare i valori forniti dai vari dispositivi. Create un nuovo foglio, ed assegnate alla prima riga le label che identificheranno il relativo dato, tutte le altre righe saranno vuote. Come potete vedere nella **Fig. 2**, le prime due colonne contengono un identificativo numerico ed una label, le successive due colonne le coordinate GPS e le successive due la data e l'ora dell'ultima serie di dati caricata, mentre tutte le altre colonne contengono i dati rilevati dal dispositivo, nel nostro esempio la tensione di alimentazione e i parametri ambientali.

Tutti i dati facenti parte del foglio di calcolo verranno salvati in automatico ad ogni aggiornamento in drive. A questo punto è necessario poter gestire la scrittura nelle celle del foglio di calcolo da un dispositivo esterno pertanto accediamo al tab strumenti e clicchiamo su editor script come indicato nella **Fig. 3**. Create un nuovo script (rinominabile a piacimento) ed inserite il listato contenuto nel file script.gs fornito assieme ai file di questo numero della rivista, come visibile in **Fig. 4**, quindi salvate il file. All'interno di questo listato troverete due funzioni update_row (**Listato 1**) e add_row (**Listato 2**) che permettono a un dispositivo esterno di aggiornare i dati di una riga oppure di aggiungere una nuova riga. In entrambe le funzioni è presente un riferimento URL al foglio di calcolo, che coincide con l'indirizzo della barra indirizzi del foglio di calcolo, vedasi **Fig. 3** e **Fig. 4**. Per evitare che qualcuno inavvertitamente possa andare a sovrascrivere i nostri dati abbiamo oscurato alcuni caratteri del

Fig. 2
Tabella del foglio
di calcolo.

MeteoNetData										
File Modifica Visualizza Inserisci Formato Dati Strumenti Componenti aggiuntivi Guida Appena modificato										
N31										
	A	B	C	D	E	F	G	H	I	
1	ID	Label	Latitude	Longitude	Date	Time	Power	Temperature	Humidity	
2	2	VA-567	45.6605746	8.7578234	04/07/2021	22:10:45	2.90V	26.10	73	
3	3	MI-296	45.4628328	9.1076926	13/12/2020	22:26:19	3.36V	10.00	90	
4	4	BG-589	45.6111051	9.5715243	17/05/2020	12:09:45	4.87V	21.80	87	
5	5	LC-567	45.8511235	9.3661942	04/05/2020	19:39:41	4.83V	28.30	56	
6	6	CO-267	45.8109765	9.0885206	04/05/2020	19:39:41	3.15V	15.40	76	
7	7	NO-390	45.4538839	8.5807616	06/12/2020	22:39:12	2.90V	20.00	85	
8										
9										

Listato 1

```
function update_row(e)
{
  // aggiorna la riga di posizione id
  var sheet = SpreadsheetApp.openByUrl(...);
  var row = e.parameters.id;
  var lbl = e.parameters.lbl;
  var lat = e.parameters.lat;
  var lon = e.parameters.lon;
  var pow = e.parameters.pow;

  var CurrentDate = new Date();
  var Date_ = Utilities.formatDate(CurrentDate, "IST", "dd/MM/YYYY");
  var Time_ = Utilities.formatDate(CurrentDate, "IST", "HH:mm:ss");

  // aggiorna la riga di posizione id
  sheet.getRange("A" + row).setValue(row);
  sheet.getRange("B" + row).setValue(lbl);
  sheet.getRange("C" + row).setValue(lat);
  sheet.getRange("D" + row).setValue(lon);
  sheet.getRange("E" + row).setValue(Date_);
  sheet.getRange("F" + row).setValue(Time_);
  sheet.getRange("G" + row).setValue(pow);
  return 1;
}
```

link di riferimento che dovrà essere sostituito con quello presente nel vostro foglio di calcolo:

```
SpreadsheetApp.openByUrl('https://docs.google.com/spreadsheets/d/1TpzxGyEAhQ5n*****Wn8ZyTwZvscWoo3b9K-nU/edit#gid=0');
```

Per rendere operativo lo script è necessario che venga pubblicato, pertanto clicchiamo sul tab "pubblica" e selezioniamo "distribuisci come applicazione web", allorché si aprirà una finestra nella quale andremo a specificare: versione progetto "nuovo", il nostro indirizzo mail e l'autorizzazione per "Anyone, even

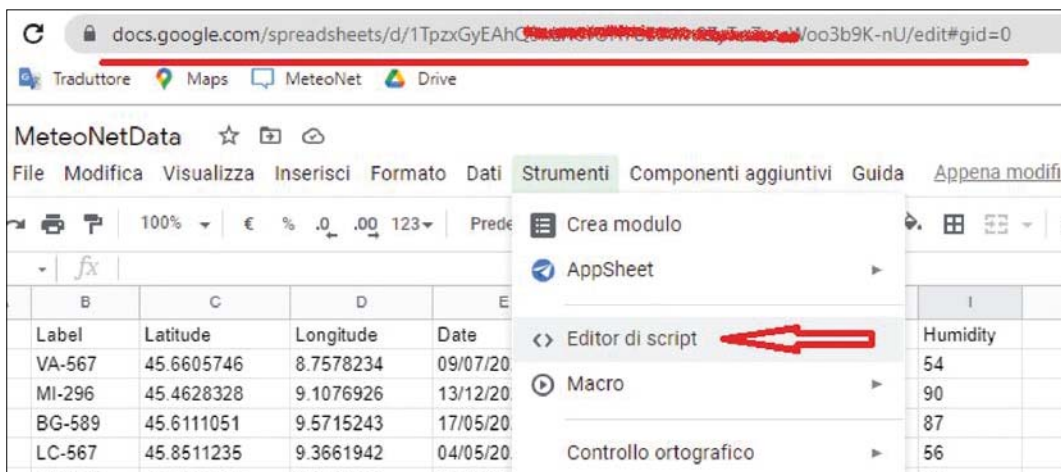


Fig. 3
Creazione di uno script in Google sheet.



Fig. 4
Editor degli script.

▼ Listato 2

```
function add_row(e)
{
  //aggiunge una riga alla fine di tutte le altre
  var sheet = SpreadsheetApp.openByUrl(...);
  var lastVal = sheet.getRange("A1:A").getValues();
  var num = lastVal.filter(String).length;

  var val1 = e.parameters.val1;
  Logger.log(val1);
  console.log(num);
  var CurrentDate = new Date();
  var Date_ = Utilities.formatDate(CurrentDate, "IST", "dd/MM/YYYY");
  var Time_ = Utilities.formatDate(CurrentDate, "IST", "HH:mm:ss");

  //aggiunge una riga alla fine
  sheet.appendRow([num, Date_, Time_, e.parameter.val1]);

  return 1;
}
```

➔ **Fig. 5**
Pubblicazione dello script: il primo step.

Deploy as web app

Current web app URL: https://script.google.com/macros/s/AKfycbxxk_N27xk0cA [Disable web app](#)

Test web app for your latest code.

Project version:
Nuovo

Describe what has changed...

Execute the app as:
Me (anonymous@gmail.com)

You need to authorize the script before distributing the URL.

Who has access to the app:
Anyone, even anonymous

[Aggiorna](#) [Annulla](#) [Guida](#)

➔ **Fig. 6**
Richiesta autorizzazione per lo script.

Deploy as web app

Project version:
1

Authorization required
MeteoNetScript needs your permission to access your data on Google.

[Rivedi autorizzazioni](#) [Annulla](#)

[Deploy](#) [Annulla](#) [Guida](#)

anonymous" come indicato in **Fig. 5**. Non appena cliccate su *aggiorna* si aprirà una finestra di pop-up che vi avvisa che affinché lo script possa modificare il foglio di calcolo serve l'autorizzazione (**Fig. 6**); quindi cliccate su *rivedi autorizzazioni*, allorché vi verrà chiesto l'account che autorizzerà lo script, selezionate lo stesso che avete usato per creare il foglio di calcolo. Ora si aprirà una nuova finestra che vi avvisa che l'app non è stata verificata (**Fig. 7**); per proseguire cliccate sul link "Avanzate" in basso a sinistra e così potete accedere alla schermata di autorizzazione (**Fig. 8**), nella quale cliccate su "Consenti" per autorizzare definitivamente lo script.

Ora si aprirà un'ulteriore schermata che vi conferma l'avvenuta autorizzazione e sarà visualizzato il link di condivisione che dovrete copiare per il suo successivo utilizzo (**Fig. 9**). Per ulteriore sicurezza riceverete anche una mail che vi avvisa che è stato autorizzato uno script ad eccedere al vostro account Google; a questo punto lo script è predisposto per poter aggiornare il foglio di calcolo in base ai dati provenienti da un dispositivo esterno. Come avrete intuito la sicurezza è molto importante per Google che non lascia nulla al caso. Potete verificare il funzionamento dello script e del foglio di calcolo con una semplice chiamata da web browser inserendo ad esempio il link seguente:

```
https://script.google.com/macros/s/AKfycbxxk_N27xk0cA1xd*****CQDbZpOsMEG7XoV6v_DC8bK69k/exec?func=updateRow&id=2&lbl=VA-529&lat=45.6605746&lon=8.7578234&pow=3.33&tmp=23.2&hum=56
```

Questa riga è formata dall'indirizzo dello script seguito dai valori da inserire secondo la denominazione specificata nello script. Cliccando sul tab file e successivamente su pubblica sul web otterrete il link

della tabella, facilmente incorporabile in un sito già esistente, come visibile in **Fig. 10**.

DISPOSITIVI IOT

Adesso ci occupiamo di programmare un dispositivo esterno per interagire con lo script, utilizzando un economico modulo ESP8266, nel nostro caso una scheda NodeMCU (distribuita da Futura Elettronica con codice NODEMCUESP) alla quale è collegato un sensore ambientale AM2302 (codice DHT22). Lo scenario potrebbe essere quello di voler rilevare i parametri ambientali di alcune abitazioni oppure piccole stazioni meteo asservite da segnale WiFi.

In **Fig. 11** potete vedere il cablaggio tra la scheda NodeMCU ed il sensore ambientale; il collegamento tra il pin DO e RST è necessario per il risveglio del dispositivo, infatti programmeremo la CPU del modulo ESP per rimanere in modalità a basso consumo per tutto il tempo (Deep Sleep) e di risvegliarsi per l'invio dei dati ad intervalli prestabiliti (updateTime nel listato). Qui di seguito proponiamo lo spezzone dello sketch sviluppato in ambiente Arduino IDE che si occupa della definizione dei parametri di funzionamento, i campi SSID e password sono le credenziali di accesso alla propria rete WiFi.

```
const char *ssid = "*****";
const char *password = "*****";
String ScriptURL = "AKfycbxxk*****G7XoV6v_DC8bK69k";
String stationID = "2";
String stationLabel = "PD-567";
String latitude = "45.9451837";
String longitude = "12.7339713";
const unsigned long updateTime=10000;
```

ScriptURL è l'indirizzo dello script fornito durante la fase di autorizzazione (**Fig. 9**) mentre stationID e stationLabel identificano il dispositivo remoto e possono essere impostate a piacimento.

I valori di latitudine e longitudine sono le coordinate GPS del luogo dove si trova il dispositivo e possono essere facilmente ricavate tramite Maps di Google. Il valore di stationID coincide con la riga del foglio di calcolo sulla quale andremo a scrivere, è fondamentale che ciascun dispositivo abbia un valore diverso consecutivo a partire dalla prima riga libera, la numero due. Tralascieremo la parte di listato che si occupa della lettura della tensione di alimentazione e la lettura del sensore e riportiamo nel **Listato 3** la parte di programma che costruisce la stringa html che sarà inviata al server della Google (script.google.com) per la pubblicazione. Notate la parte "/exec?func=updateRow" che specifica che vogliamo



Fig. 7
Notifica
dirichiesta di
verifica.

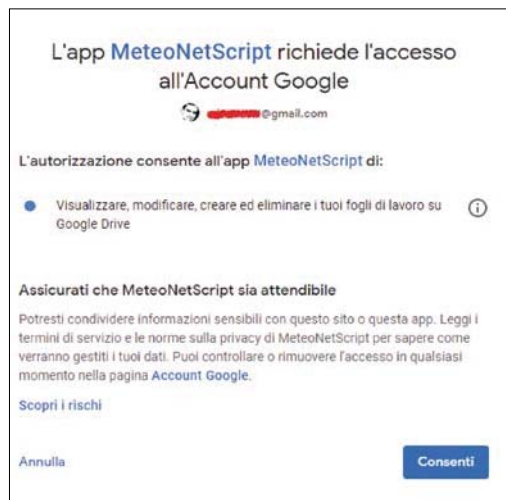


Fig. 8
Autorizzazione
all'accesso
all'account
Google.

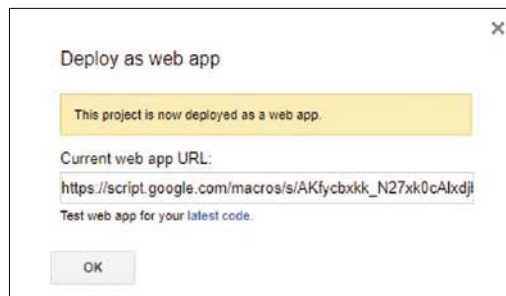
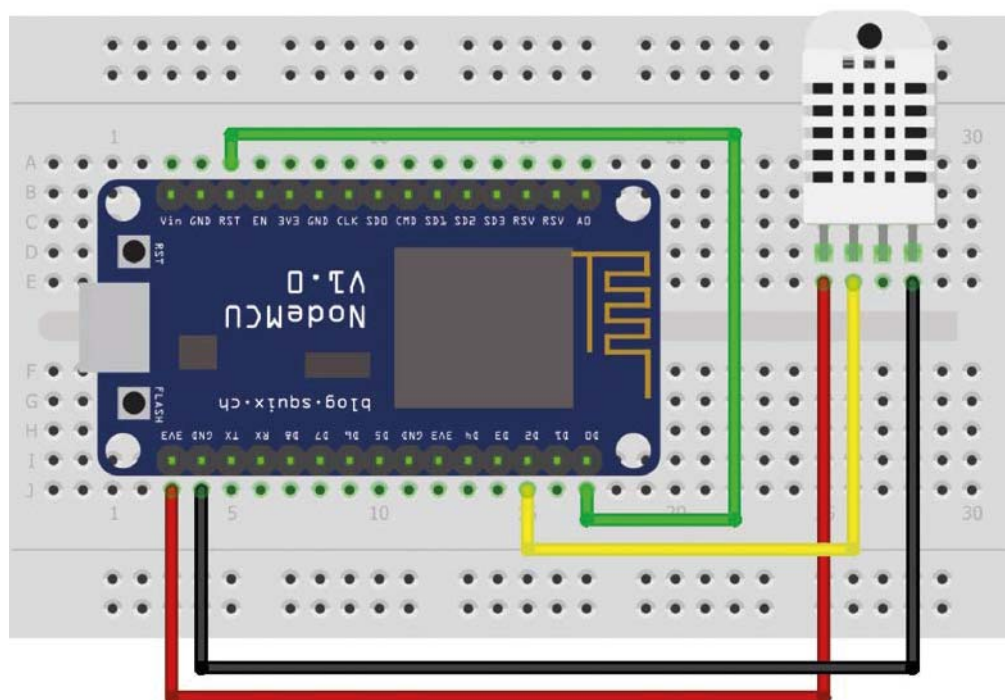


Fig. 9
Link di
condivisione
per lo script.



Fig. 10
Pubblicazione
dello script.

Fig. 11
Cablaggio della
scheda NodeMCU.



↓ Listato 3

```
url = "/macros/s/" + ScriptURL + "/exec?func=updateRow";
url += String("&id=") + stationID;
url += String("&lbl=") + stationLabel;
url += String("&lat=") + latitude;
url += String("&lon=") + longitude;
url += String("&pow=") + String(vcc) + "V";
url += String("&tmp=") + String(tmp);
url += String("&hum=") + String(hum);
...
client.print(String("GET ") + url + " HTTP/1.1\r\n" +
               "Host: " + host + "\r\n" +
               "Connection: close\r\n\r\n");
```

eseguire la funzione dello script che aggiorna la riga e la costruzione della stringa url che contiene i parametri misurati.

Il listato completo dello sketch (file di nome **Meteo-NetMap_ESP8266.ino**) è disponibile assieme ai file di questo numero della rivista.

Quando i vari dispositivi sono attivi andranno ad aggiornare le rispettive righe della tabella con i nuovi dati, la tabella contiene quindi sempre l'ultima lettura eseguita con data e ora della rilevazione. Trattandosi di un foglio di calcolo sarebbe possibile inserire un'elaborazione dei dati acquisiti come il calcolo del valore medio, massimo, minimo ecc. oppure inserire specifici grafici con la possibilità di pubblicare sul web l'intero lavoro. In **Fig. 12** potete vedere i messaggi di debug di Serial Monitor quando lo sketch è in esecuzione.

VISUALIZZARE I DATI SU UNA MAPPA

Visto che i vari dispositivi IOT sono geolocalizzati è molto interessante pubblicare i dati presenti nella tabella in una mappa. Per fare questo è necessario accedere al foglio di Google, estrapolare i dati necessari e di inserirli in una mappa come punti di interesse. Esistono diversi applicativi che permettono di ricavare una mappa dai dati presenti in un foglio di Google, come l'applicativo My Maps (sempre di Google), però essi non permettono di ricaricare la pagina con i dati aggiornati e continuano a mostrare solo i valori di quando è stata creata la mappa. Siccome vogliamo evitare servizi a pagamento, utilizziamo il JavaScript di nome Papa Parse (papaparse.com). Si tratta di uno script che va incorporato in una pagina web e che ci dà la possibilità di accedere ad ogni singolo campo di ogni singola riga della tabella creata con Google sheets.

Nel **Listato 4** (file html) è riportato un esempio di utilizzo di questo script per la lettura della colonna identificata con Label. Affinché il tutto funzioni, è necessario che modifichiate la sequenza alfanumerica denominata key con i valori dell'indirizzo del vostro foglio di lavoro (riferitevi alla **Fig. 3**).

Quando avvierete questa pagina web verranno visualizzati a video i dati relativi alla colonna identificata con Label, ma il nostro scopo finale è quello di popolare una mappa inserendo dei segnaposto in relazione ad ogni dispositivo IOT e poterne visualizzare i parametri ambientali rilevati, cosa non proprio

```
COM12

.....
WiFi connected
IP address:
192.168.178.38

Data read: Vcc 2.90V    Temperature: 28.40C    Humidity: 61%
connecting to script.google.com
Requesting URL: /macros/s/AKfycb:***C8bK69k/exec?func=updateRow&id=2&lbl=VA-567&lat=45.
closing connection

Sleep...
```

Fig. 12
Serial
Monitor di
Arduino con
lo sketch in
esecuzione.

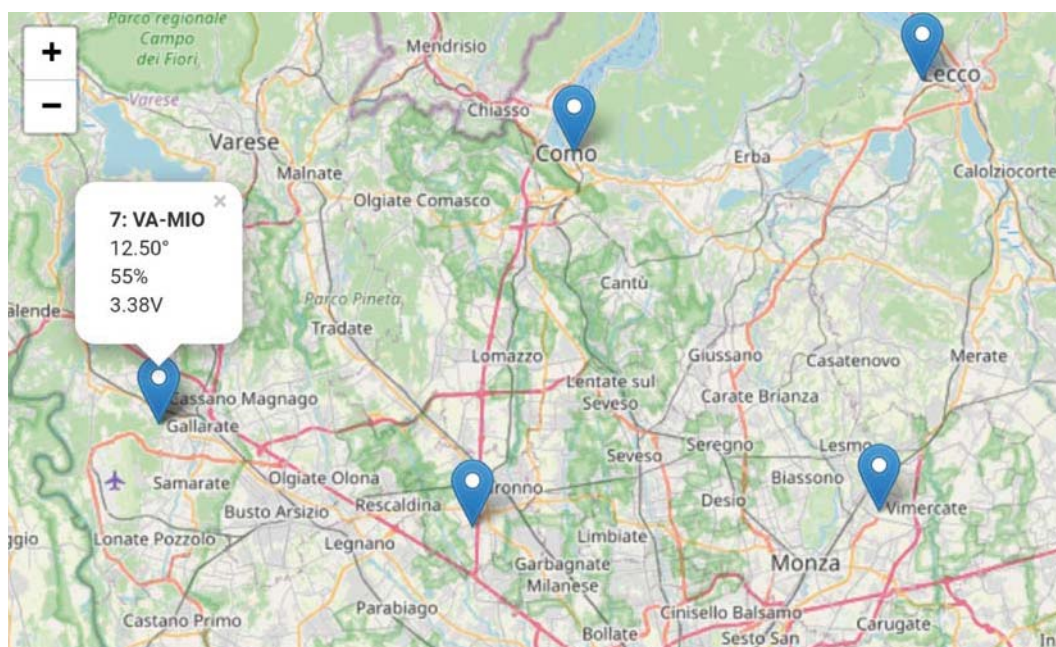
immediata, ma per fortuna possiamo usufruire della libreria java-script open source della Leaflet (<https://leafletjs.com/>) specificatamente creata per la realizzazione di mappe interattive a partire dalle mappe OpenStreetMap, il tutto senza necessità di alcuna iscrizione a differenza dei servizi Google che per gli sviluppatori oltre ad un account richiede un sistema di pagamento valido. Occorre creare una pagina web che integri entrambi gli script per visualizzare su una

mappa i segnaposto in corrispondenza della posizione dei dispositivi IOT, con la possibilità di visualizzare i dati rilevati di ciascun dispositivo. Nella documentazione fornita assieme a questo numero della rivista troverete una cartella di nome SITO che contiene già tutti i file necessari, è sufficiente che modifichiate il "spreadsheet_url" con quello relativo al vostro foglio di calcolo. La pagina web dovrà essere avviata da un server remoto, pertanto utilizzeremo il servizio

Listato 4

```
<!DOCTYPE html>
<html>
<body>
  <p id="meteo"></p>
  <script type="text/javascript" src="../../src/tabletop.js"></script>
  <script type="text/javascript">
    var public_spreadsheet_url = 'https://docs.google.com/spreadsheet/pub?hl=en_US&hl=en_US&key=1TpzxGyEA
h*****OH7bS8Wn8ZyTwZvscWoo3b9K-nU&output=html';
    function init() {
      Tabletop.init( { key: public_spreadsheet_url, callback: showInfo, simpleSheet: true } );
    }
    window.addEventListener('DOMContentLoaded', init)
    function showInfo(data) {
      // data comes through as a simple array since simpleSheet is turned on
      alert("Successfully processed " + data.length + " rows!");
      document.getElementById("meteo").innerHTML = "<strong>Position:</strong> " + [ data[0].ID, data[1].
ID ].join(", ");
      console.log(data);
      var step;
      for (step = 0; step < data.length; step++) {
        //scrive nella console click pulsante sx ispeziona console
        console.log('Walking one step');
        console.log(data[step].Label);
        document.write(data[step].Label + "<br>");
      }
      document.write("The published spreadsheet is located at <a target='_new' href='" + public_spreadshe
et_url + "'> + public_spreadsheet_url + "</a>");
    }
  </script>
</body>
</html>
```

Fig. 13
La mappa creata
con i dati acquisiti.



Google Sites (<https://sites.google.com/new>) per creare il nostro sito dove visualizzare la mappa; la Google, come al solito, ha reso le cose semplici anche per chi non ha alcuna esperienza di pagine web. Create una nuova pagina e dal menù a destra di nome "Inserisci", cliccate su "Incorpora", si aprirà un editor nel quale dovrete copiare il testo contenuto nel file `MeteoNet-MapSite.txt` fornito assieme ai file di questo numero della rivista. Assicuratevi di sostituire l'indirizzo "spreadsheet_url" con quello relativo al vostro foglio di calcolo, ricavato durante la fase di "Pubblica sul Web", come visibile in **fig.10**. Il risultato finale è visibile in **Fig. 13**, dove si vedono i segnaposto relativi a ciascun dispositivo IoT riguardante la nostra installazione di test; se cliccate su uno di essi nella vostra mappa, si aprirà una finestra di pop-up che visualizzerà i parametri ambientali relativi all'ultima misura effettuata. La pagina web non prevede un auto-refresh ma ogni volta che verrà avviata mostrerà gli ultimi valori acquisiti.

CONCLUSIONI

Lungi dall'esse vicini ad un prodotto commerciale come quello proposto dall'azienda Netatmo, visibile alla pagina <https://weathermap.netatmo.com>, oppure il servizio wundermap della Wunderground, dalla nostra abbiamo il vantaggio di poter personalizzare il sistema anche al di fuori dei parametri ambientali, sarebbe infatti facilmente adattabile per monitorare l'altezza dell'acqua in diversi invasi piovani. Alcune criticità sono emerse nello sviluppare questo progetto, prima fra tutte il vincolo della presenza del

WiFi per far dialogare i dispositivi, mentre sarebbe utile voler misurare i parametri ambientali in luoghi non asserviti da segnale WiFi. Abbiamo riscontrato poi alcune difficoltà a rendere operativo lo script che gestisce la scrittura sul foglio di calcolo per la presenza (necessaria) dei controlli di sicurezza della Google. Un ulteriore fattore da prendere in considerazione riguarda l'alimentazione, infatti la scheda NodeMCU necessita di un alimentatore da parete a 5V, come quelli usati per la ricarica degli smartphone, quindi niente funzionamento stand alone. Stiamo già lavorando ad una rivisitazione di questo progetto per poter integrare anche dispositivi che comunicano in rete tramite segnale GSM e vorremmo anche semplificare l'accesso al foglio di calcolo utilizzando specifici servizi web, ma di questo vi daremo conto in futuro. □



Cosa occorre?

Il modulo con ESP8266 (cod. NODEMCUESP) è disponibile a Euro 9,50, il sensore di temperatura e umidità (cod. DHT22) è in vendita a Euro 9,00. Chi volesse realizzare un proprio applicativo può utilizzare il servizio PCBPRODUCTION disponibile sul sito www.futurashop.it per realizzare il PCB. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

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

Analizza i circuiti nei minimi dettagli!

Completa il tuo laboratorio di elettronica con le lampade a LED con lente d'ingrandimento e realizza i tuoi progetti in modo più semplice e preciso.

Lampada LED con lente d'ingrandimento 3 diottrie - 1.200 lumen



Lampada costituita da 60 LED bianchi con lente d'ingrandimento Ø127 mm a 3 diottrie con flusso luminoso di 1.200 lumen. Particolarmente indicata per compiere lavori di precisione. Utile anche per uso domestico, dispone di regolazione della luminosità su quattro livelli, un doppio braccio che ne permette l'orientamento nella posizione più adatta. Fornita di morsetto metallico per il fissaggio al banco. Alimentazione 220 VAC. Colore bianco. Consumo 14 W.



€78,00
Cod. LAMPDESK3

DISPONIBILI ANCHE NELLE
VERSIONI A 5 DIOTTRIE
COD. LAMPDESK5 € 78,00
E A 8 DIOTTRIE
COD. LAMPDESK8 € 78,00

Lampada professionale a LED dimmerabile - 880 lumen



Lampada da tavolo professionale dimmerabile, dotata di 90 LED SMD. Fornita di morsetto metallico per il fissaggio al banco, dispone di doppio braccio con una lunghezza di 400 mm. Alimentazione 220 VAC - 240 VAC / 50 Hz. Colore bianco. Offre la possibilità di variare la luminosità tramite i pulsanti touch.

€82,00
Cod. TP233

Lampada a LED con lente d'ingrandimento 5 diottrie 600 lumen

Lampada costituita da 30 LED bianchi con lente d'ingrandimento Ø100 mm a 5 diottrie. Ideale per lavorazioni di precisione, può essere posizionata ovunque grazie al braccio flessibile. L'elevato peso della base garantisce una eccellente stabilità. Consumo 7,5 W. Alimentazione 12 VDC, tramite adattatore da rete 230 VAC / 50 Hz.



€39,00
Cod. LAMP9101AE

Prezzi IVA inclusa.

STAZIONE DI MONITORAGGIO QUALITÀ DELL'ARIA

di **VALTER SARTORI**



L'argomento "Ambiente" rimane sempre ai vertici dell'attualità e non

passa giorno che i media non propongano notizie di situazioni che fino a qualche anno fa parevano impensabili. Anche la Giustizia viene coinvolta pesantemente in questi temi, con indagini nei confronti di vertici amministrativi sia pubblici, sia privati, con accuse particolarmente pesanti riguardanti la salute dei cittadini.

La lotta allo smog e a qualsiasi tipo di inquinamento è la prossima battaglia cui dovremo partecipare tutti, cercando di preservare e rispettare l'ambiente che circonda ciascuno di noi. Non esiste un migliore inizio se non renderci conto personalmente del livello di inquinamento che ci circonda. Per renderlo possibile occorre prendere in considerazione un discreto numero di fattori,

tra i quali l'esame di particolari gas inquinanti emessi solitamente dalla combustione delle nostre caldaie o dai focolari, i gas derivanti dalle colture agricole e dagli allevamenti e non meno importante dagli inquinanti derivati dal traffico stradale. Qualcuno penserà: "Ma come è possibile realizzare un analizzatore in grado di

Realizzata con l'ausilio di sensori dedicati alla rilevazione di elementi inquinanti e di grandezze meteorologiche, permette un'analisi approfondita dell'inquinamento atmosferico. I valori rilevati vengono trasmessi a distanza permettendone un'accurata valutazione.

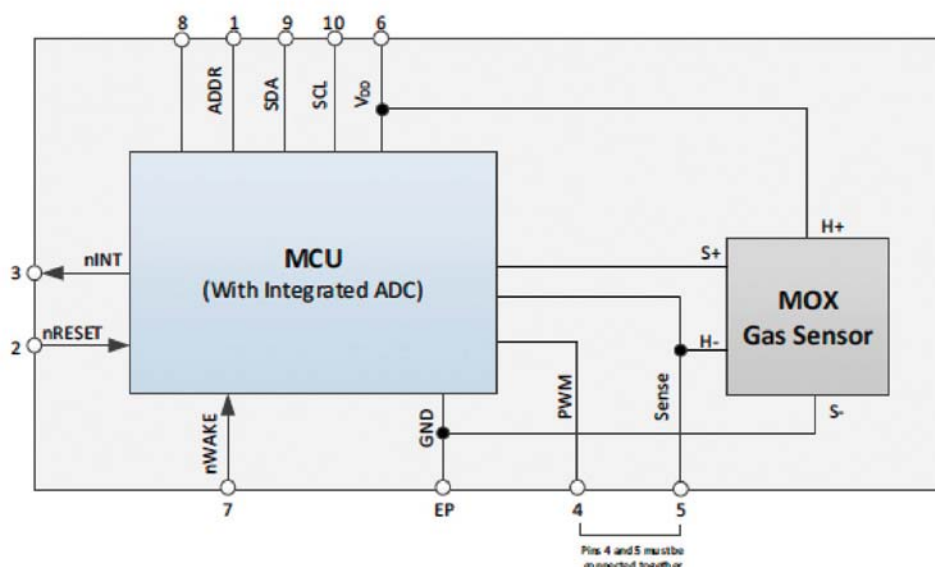
fornire misure simili a quelle effettuate dalle centraline di monitoraggio aria dislocate in varie aree della città?" Ebbene, non pretenderemo di arrivare a tali livelli, ma non è impossibile effettuare misurazioni che, confrontandole con i valori resi pubblici dagli organismi preposti, hanno un livello di accuratezza molto interessante; e ve lo

dimosteremo con questo progetto.

Naturalmente per ottenere risultati affidabili, oltre ad una buona progettazione elettronica, occorre dare molta importanza al metodo di acquisizione dei campioni, onde evitare di alterare le condizioni fisiche e chimiche dell'aria analizzata. È di fondamentale importanza non solo non riscaldare, tramite ad esempio alimentatori o altri circuiti, l'aria in esame, ma anche mantenere i percorsi dei campioni di analisi il più brevi possibile.

Per ottenere tali risultati abbiamo realizzato un apposito shield collocato su una scheda Arduino Uno e interfacciato con una scheda di supporto per la gestione dei comandi e la visualizzazione dei dati tramite display LCD. L'alimentazione primaria del sistema è prevista a 12Vcc e viene implementata da un convertitore DC/DC switching, che dovrà essere installato all'esterno del sistema di analisi, allontanan-

Fig. 1
Lo schema a blocchi del CCS811.



do così eventuali fonti di calore. La scelta di questo tipo di alimentazione è nata con il presupposto di poter alimentare la stazione mediante una batteria locale, eventualmente ricaricabile mediante pannello fotovoltaico. Tale soluzione permetterebbe l'installazione della stazione di monitoraggio in qualsiasi luogo. Bene, stabilito ciò possiamo ad analizzare per primi i sensori utilizzati, per poi descrivere l'elettronica nel suo insieme.

I SENSORI

Come abbiamo anticipato, il progetto ha come finalità un'analisi chimico fisica dell'aria più precisa ed esauriente possibile; per ottenere risultati particolarmente esaustivi abbiamo preso in considerazione una nutrita serie di sensori, valutando le specifiche tecniche e i relativi costi.

La nostra attenzione si è rivolta verso l'adozione di particolari sensori prodotti da SGX, BOSCH, Texas

e Ams, integrati in particolari moduli denominati MiCS-6814 e CJMCU-8128 (rispettivamente MICS6814MOD e CJMCU8128MOD disponibili dal sito www.futurashop.it).

Per completare l'analisi non potevano non mancare le misure del PM2.5 e del PM10, che abbiamo affidato al polverimetro SDS011 della Nova (cod. PMSSENSOR) e ampiamente descritto nei fascicoli n° 241 e n° 246 di Elettronica In, cui rimandiamo per gli approfondimenti del caso.

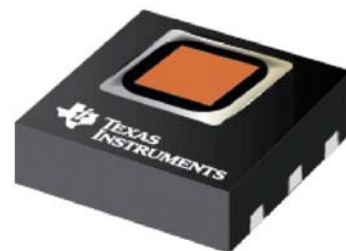
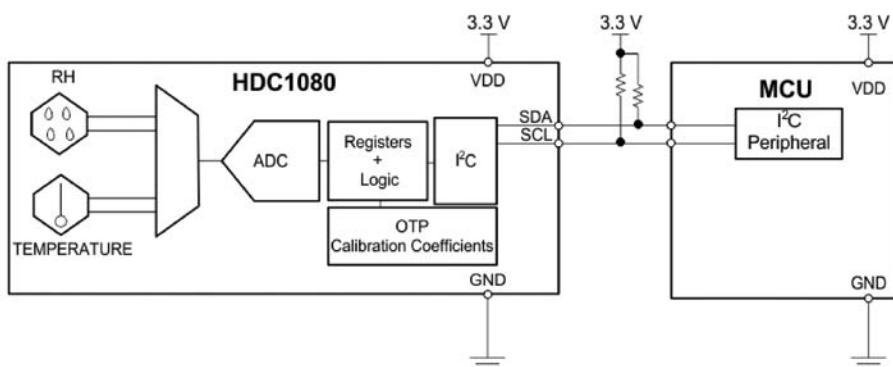
Il modulo CJMCU-8128 include tre sensori: il CCS811, l'HDC1080 e il BMP280.

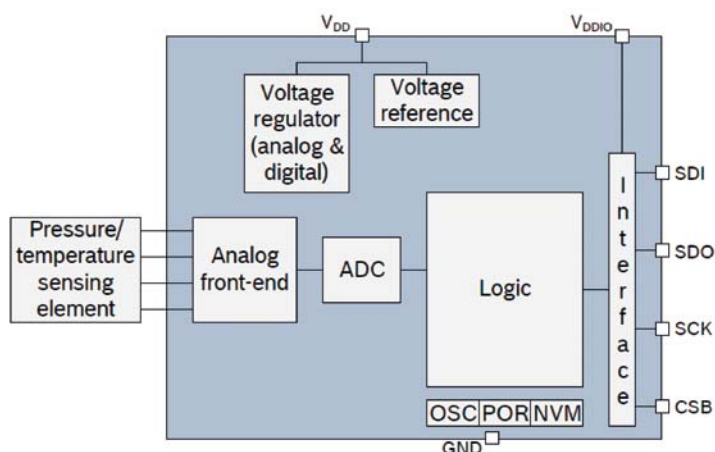
Il CCS811 della Ams, del quale nella **Fig. 1** è rappresentato lo schema a blocchi, è un sensore a bassissimo consumo di tipo MOx (metal-oxide) che rileva un'ampia gamma di composti organici volatili (VOCs) e valori equivalenti di CO₂ (e CO).

Il CCS811 racchiude in sé non solo il sensore MOx, ma anche una MCU cui è destinato il compito di convertire il segnale analogico, elaborarlo, e met-

Fig. 2
Schema a blocchi HDC1080.

Fig. 3
Package HDC1080.



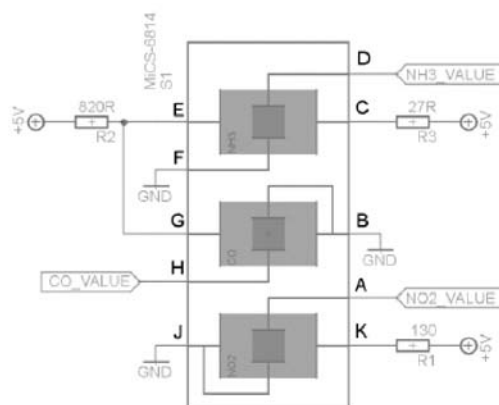


← **Fig. 4**
Schema a blocchi del
BMP280.

↑ **Fig. 5**
Breakout board
CJMCU-8128.

tere i dati a disposizione su bus I²C.
Il CCS811 dispone di un particolare algoritmo che è in grado di elaborare i dati grezzi forniti dal sensore MOx compensandoli con le misure di temperatura e umidità forniti dal nostro programma software. Il range di uscita della CO₂ equivalente parte da 400 ppm per arrivare fino a 29.206 ppm.
L'HDC1080 della Texas Instruments è impiegato per il rilievo dell'umidità e della temperatura dell'aria. La misura dell'umidità ha un'accuratezza del $\pm 2\%$, mentre la temperatura ha una precisione di $\pm 0,2^\circ\text{C}$. Come il CCS811, l'HDC1080 dispone di una MCU che elabora i dati forniti dai sensori di temperatura e umidità e fornisce su bus I²C i risultati. In **Fig. 2** è rappresentato lo schema a blocchi e in **Fig. 3** trovate il package del dispositivo.
Il BMP280 della Bosch è un sensore di temperatura e pressione barometrica assoluta di tipo piezoresistivo, con campo di misura $300 \div 1100$ hPa; nel nostro caso è stato utilizzato esclusivamente per la misura di pressione barometrica, avendo già utilizzato l'HDC1080 per la misura di temperatura. Le peculiarità di questo sensore oltre che nella linearità e stabilità, sono evidenziate anche nella sua accuratezza relativa che la Bosch stima nel suo datasheet di $\pm 0,12$ hPa, pari a ± 1 m di colonna d'aria. In **Fig. 4** vedete il diagramma a blocchi che evidenzia l'architettura del sensore e la sua interfaccia I²C-Bus di comunicazione.
I tre sensori illustrati dianzi fanno parte di un'unica breakout board, commercializzata con la sigla CJMCU-8128, la quale attraverso il bus I²C che accomuna i tre sensori fornirà i dati meteorologici di pressione atmosferica, temperatura ambien-

te e umidità e la misura della CO₂ equivalente, utilizzando le apposite librerie create per l'ambiente Arduino, tra le quali la SparkFunCCS811, la SparkFunBME280 e la ClosedCube_HDC1080. In **Fig. 5** vedete la breakout board CJMCU-8128 con i suoi tre sensori.
Il MiCS-6814 della SGX è un sensore di tipo MEMS (Micro Electro-Mechanical Systems) che racchiude tre elementi sensibili indipendenti all'interno del suo package. Ciascun sensore è realizzato in ossido di metallo e presenta una resistenza variabile in presenza di gas, in particolare il monossido di carbonio (CO), l'ammoniaca (NH₃) e il biossido di azoto (NO₂). Concentreremo la nostra attenzione sugli elementi inquinanti fino a qui descritti e analizzeremo in particolare i metodi e gli algoritmi alla base della valutazione dei composti misurati dal modulo MiCS-6814.



← **Fig. 6**
Schema
descrittivo
del MiCS-
6814.

Fig. 7
Grafico della
relazione tra
CO e il rapporto
RS/R0.

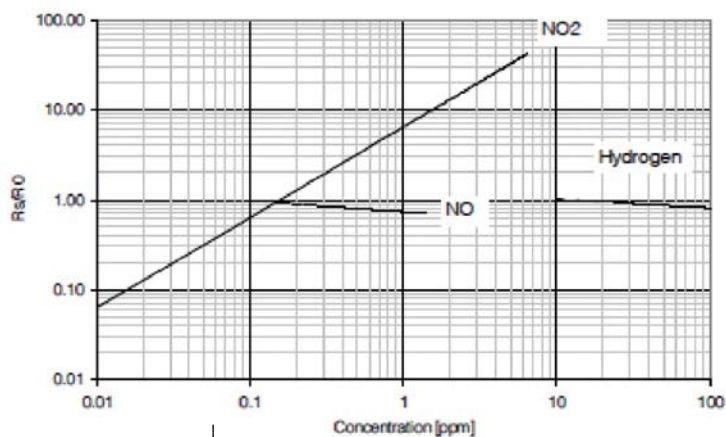
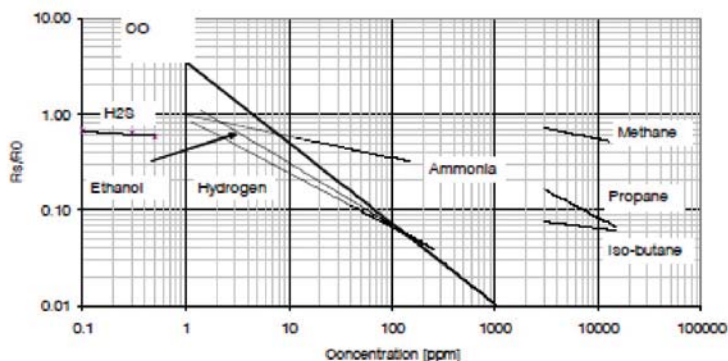
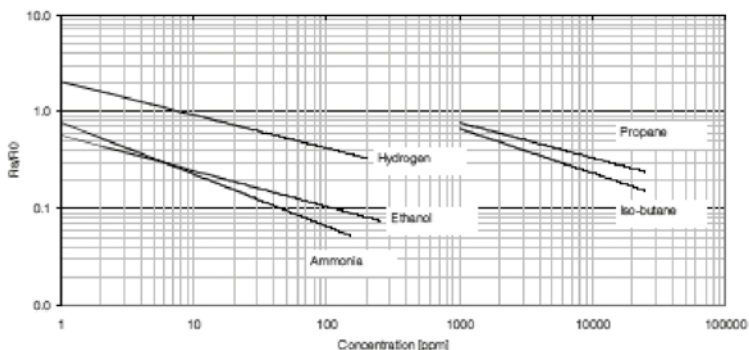


Fig. 8
Grafico della
relazione tra
NO2 e il
rapporto RS/R0.

Fig. 9
Grafico della
relazione tra
NH3 e il
rapporto RS/R0.



Prendendo in considerazione la **Fig. 6**, riguardante il circuito descrittivo del MiCS-6814, si può notare che l'alimentazione a 5V è collegata a tre diversi resistori R1, R2 e R3 di valore 130Ω, 820Ω e 27Ω. La funzione di questi resistori è di mantenere il circuito di riscaldamento di ciascun elemento sensibile alla corretta temperatura di funzionamento. Ogni elemento sensibile presenta una connessione verso massa (GND) e una connessione verso uno specifico pin del modulo che verrà interfacciato

con un circuito in grado di analizzare il valore della resistenza del MOx.

Nelle **Fig. 7**, **Fig. 8** e **Fig. 9** vengono evidenziate le curve di ogni singolo sensore relazionando la concentrazione in ppm (parti per milione) del composto in esame con il rapporto tra la resistenza misurata e la resistenza R0 che il sensore assume quando non è presente l'inquinante. È importante notare che tali grafici hanno una scala logaritmica, per cui la funzione che lega queste due grandezze non è lineare.

Il principio di base utilizzato dai sensori MOx (metal-oxide) presenti nel MiCS-6814 è la variazione di resistenza in presenza di gas. Per i gas ossidanti come l'ozono o gli ossidi di azoto a contatto con il sensore, avviene una reazione di ossidazione, cioè l'elemento sensibile acquista elettroni e di conseguenza aumenta la sua resistenza. Per gas come il monossido di carbonio avviene una reazione opposta denominata riduzione, con perdita di elettroni da parte dell'elemento sensibile e conseguente diminuzione della resistenza.

IL METODO DI ELABORAZIONE

Prima di illustrare i dettagli del progetto, occorre capire al meglio quali saranno i parametri che andremo a rilevare attraverso sensori descritti in precedenza. All'interno di ciascuno dei sensori della breakout board CJMCU-8128 c'è un microcontrollore, ognuno predisposto con algoritmi specifici sia per l'acquisizione, sia per la gestione dei parametri rilevati. Per mezzo del bus I²C, otterremo dati elaborati di pressione atmosferica in hPa, temperatura in gradi Celsius e umidità in percentuale. Il dato riferito all'anidride carbonica equivalente verrà fornito in ppm (parti per milione).

Il misuratore SDS011, interfacciato tramite la sua uscita seriale, ci fornirà dati già pre elaborati da algoritmi presenti all'interno della sua MCU e presentati in uscita in μg/m³ sia per la misura del PM2.5 sia per il PM10.

Uno spazio rilevante deve invece essere dedicato al modulo MiCS-6814 su breakout board.

Come accennato precedentemente le misure di resistenza che verranno effettuate forniranno dei dati grezzi che dovranno essere elaborati alla luce dei diagrammi riportati nelle **Fig. 7**, **Fig. 8** e **Fig. 9**. Risulta assai rilevante dalla lettura delle note sui datasheet, l'importanza di non alterare in alcun modo la temperatura dell'elemento sensibile per non influenzare la misura.

I tre elementi sensibili presenti all'interno del MiCS-6814 vengono preriscaldati con l'ausilio

Tabella 1
Caratteristiche
elemento Red
Sensor.

Performance RED sensor

Characteristic RED sensor	Symbol	Typ	Min	Max	Unit
Sensing resistance in air (see note 1)	R_0	-	100	1500	k Ω
Typical CO detection range	FS	-	1	1000	ppm
Sensitivity factor (see note 2)	S_{SO}	-	1.2	50	-

Performance OX sensor

Characteristic OX sensor	Symbol	Typ	Min	Max	Unit
Sensing resistance in air (see note 1)	R_0	-	0.8	20	k Ω
Typical NO ₂ detection range	FS	-	0.05	10	ppm
Sensitivity factor (see note 3)	S_R	-	2	-	-

Tabella 2
Caratteristiche
elemento OX
Sensor.

Red Sensor, l'Ox Sensor e l'NH₃ Sensor.

A questo punto il gioco sembrerebbe fatto: abbiamo una indicazione della concentrazione dei nostri tre composti, ma purtroppo i valori volumetrici in ppm non forniscono una unità di misura standard

Performance NH3 sensor

Characteristic OX sensor	Symbol	Typ	Min	Max	Unit
Sensing resistance in air (see note 1)	R_0	-	10	1500	k Ω
Typical NH ₃ detection range	FS	-	1	300	ppm
Sensitivity factor (see note 4)	S_R	-	1.5	15	-

Tabella 3
Caratteristiche
elemento NH3
Sensor

come si evince anche dalle tabelle che gli enti preposti al controllo ambientale espongono nei loro report, dove la concentrazione degli inquinanti viene espressa come valori in massa $\mu\text{g}/\text{m}^3$, mg/m^3 o addirittura g/m^3 .

Per ottenere il valore di concentrazione corretto occorre conoscere il peso molecolare delle nostre sostanze in esame ed applicare la formula:

$$C \left[\frac{\text{mg}}{\text{m}^3} \right] = \text{ppm} \frac{M}{22,41 \frac{T}{273} \frac{1013}{P}}$$

nella quale:

- C = concentrazione gas in $[\text{mg}/\text{m}^3]$
- ppm = concentrazione gas in [ppm]
- M = peso molecolare del gas $[\text{g}/\text{mol}]$
- T = temperatura assoluta gas misurato
- P = pressione barometrica [hPa]

Notare che il peso molecolare o massa molare M

delle resistenze R1, R2 e R3, pertanto occorrerà non alterare la temperatura con un circuito che potrebbe far circolare anche pochi milliAmpere nell'elemento sensibile.

Risulterebbe assai semplice collegare una rete resistiva, come ipotizzato in alcuni esempi, e utilizzare gli ingressi analogici di Arduino Uno a nostra disposizione per misurare la tensione presente sul MOX e di conseguenza ricavarne la resistenza, ma ad un più approfondito esame la soluzione tecnica che abbiamo adottato è risultata molto più performante. È stato realizzato un circuito a corrente impressa per ogni singolo elemento sensibile, calibrato per poter leggere con assoluta precisione i valori di resistenza raggiunti dai MOX.

In **Tabella 1** possiamo vedere le caratteristiche dell'elemento sensibile per la misura del CO, detto anche 'Red Sensor', dove troviamo i possibili valori che può assumere la resistenza R0: da un minimo di 100 k Ω ad un massimo di 1500 k Ω .

Per la misura dell'NO₂ l'Ox Sensor, così denominato, può raggiungere una resistenza di 20 k Ω partendo da un minimo di 0,8 k Ω (**Tabella 2**).

L' NH3 Sensor potrà assumere valori di resistenza compresi tra 10 k Ω e 1.500 k Ω (**Tabella 3**).

Valutando il massimo valore che potrebbero raggiungere le resistenze R0, comparabili con la RS, e il valore di tensione che verrebbe letto ai loro capi, la corrente impressa per il Red Sensor sarà di 1 μA , per l'NH₃ Sensor sarà di 1 μA , mentre per l'Ox Sensor sarà di 100 μA .

Con questa soluzione è possibile non risentire dell'autoriscaldamento dei sensori dovuto al passaggio di corrente al loro interno.

In precedenza avevamo accennato al fatto che la curva di risposta dei sensori non è lineare, bensì logaritmica, per cui occorrerà relazionare il valore di resistenza letto con la curva logaritmica che permette di ricavare il valore del composto in esame in ppm. Per essere più precisi dovremo rapportare il valore di resistenza letto ad un valore chiamato R0 che verrà introdotto nel paragrafo dedicato alla configurazione. Per ora definiremo questo rapporto RS/R0 come indicato anche nei grafici.

Il prossimo passo consisterà nell'ottenere da un grafico a base logaritmica una formula matematica che possa permetterci di calcolare, dato il rapporto RS/R0 il valore in ppm dell'agente inquinante. Per ottenere ciò abbiamo rilevato dai grafici una serie di punti che sono stati interpolati mediante una funzione potenza di regressione, ottenendo i grafici e le rispettive funzioni evidenziate nelle **Fig. 10**, **Fig. 11** e **Fig. 12**, rispettivamente per il

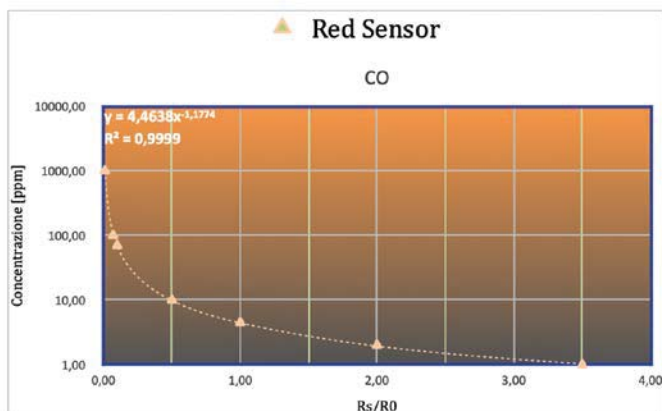


Fig. 10
Correlazione tra i punti di concentrazione della CO [ppm] e il rapporto RS/R0 e la conseguente formula.

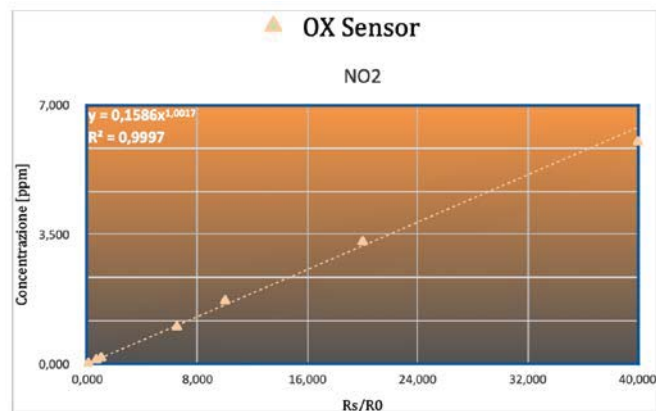


Fig. 11
Correlazione tra i punti di concentrazione di NO2 [ppm] e il rapporto RS/R0 e la conseguente formula.

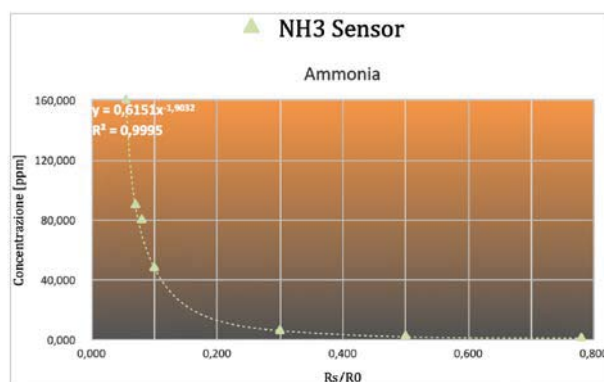


Fig. 12
Correlazione tra i punti di concentrazione di NH3 [ppm] e il rapporto RS/R0 e la conseguente formula.

ha valore differente in base al gas considerato:

- CO2 = 44,01 g/mol
- CO = 28,01 g/mol
- NO2 = 46,0055 g/mol
- NH3 = 17,031 g/mol

Gli algoritmi descritti sono stati implementati direttamente nel firmware che la scheda Arduino

Uno elaborerà. Verranno forniti in uscita a display i dati acquisiti ogni minuto, mentre sarà trasmessa, mediante il modulo LoRa, la media per ogni singolo elemento su dieci minuti.

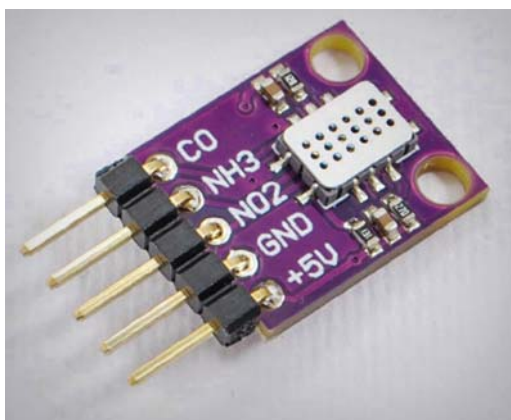
L'HARDWARE DELLA CENTRALINA

Il progetto è composto da tre schede, oltre alla Arduino Uno e al polverimetro SDS011: una scheda alimentatore, una scheda di analisi e una scheda di gestione del sistema. A quest'ultima è collegato un display LCD 2004 a 20 caratteri e 4 righe (cod. LCD20X4AI2C).

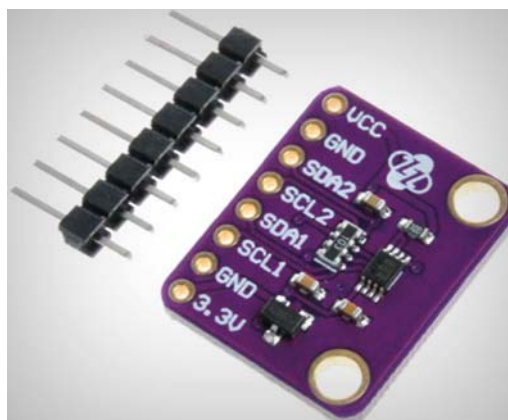
La scheda di alimentazione dispone di un ingresso a tensione compresa tra 7 V e 15 Vcc e quattro uscite; la prima di queste alimenta la scheda Arduino Uno con la tensione applicata all'ingresso, la seconda, con tensione variabile tra 1,5 V e 4 Vcc, è destinata alla gestione della ventilazione per il trasporto del campione d'aria dall'esterno verso la scheda di analisi. La terza uscita alimenta la scheda di analisi (AnalyzeShield, così verrà chiamata d'ora in poi questa scheda) con tensioni di 3,3 V e 5 Vcc, mentre la quarta alimenta con 5 Vcc il display. La AnalyzeShield viene montata diret-

Tabella 4

MODULO CONFIGURAZIONE	NO2 (100 μ A)	NH3 (1 μ A)	CO (1 μ A)
R1 = 1 k Ω	100 mV		
R3 = 10 k Ω		10 mV	
R2 = 100 k Ω			100 mV
R1 = 20 k Ω	2,00 V		
R3 = 1 M Ω		1,00 V	
R2 = 1 M Ω			1,00 V



← **Fig. 13**
(a sinistra)
La breakout
board
MiCS-6814.



← **Fig. 14**
(a destra)
La breakout
board
PCA9306.

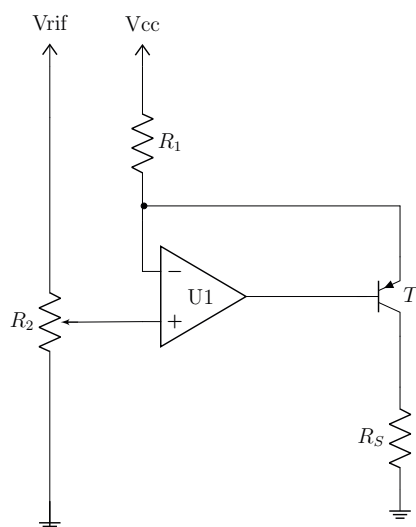
tamente sopra la scheda Arduino Uno. Dalla AnalyzeShield partono i collegamenti, a mezzo flat-cable, verso il display LCD 2004 e verso l'SDS011. Il progetto prevede anche la realizzazione di due moduli di taratura con resistenze di simulazione dei sensori MOx comprese tra i valori più bassi e quelli più alti raggiungibili e dichiarati nei data-sheet. Questi moduli serviranno per tarare le correnti dei circuiti a corrente impressa rispettando i valori riportati nella **Tabella 4**.

SCHEDA ANALYZESHIELD

La scheda preposta alla misura delle sostanze inquinanti e agli aspetti ambientali è stata realizzata mantenendo le dimensioni e la disposizione dei connettori di Arduino Uno, ottenendo così uno shield assolutamente prestazionale per ingombro e funzionalità. L'AnalyzeShield ospiterà i moduli CJMCU-8128, il MiCS-6814 (**Fig. 13**) e l'ADS1115, convertitore A/D 4 canali a 16 bit utilizzato per la determinazione della resistenza R_S . Il modulo CJMCU-8128 è interfacciato con Arduino Uno mediante bus I²C ed è alimentato ad una tensione di 3,3V.

Per interfacciare in sicurezza i moduli sul bus I²C sono stati predisposti due traslatori di tensione bidirezionale: il primo è costituito da un PCA9306 (cod. PCA9306MOD) (**Fig. 14**) che permette di realizzare un bus I²C a livello 3,3V oltre a quello a 5V. Il secondo traslatore, un TXS0108 della Texas Instruments (cod. TXS0108MOD dal sito www.futurashop.it), viene impiegato per realizzare una rete SPI a 3,3V per la connessione del modulo trasmettitore LoRa, impiegato per trasmettere a distanza le misure effettuate dal sistema.

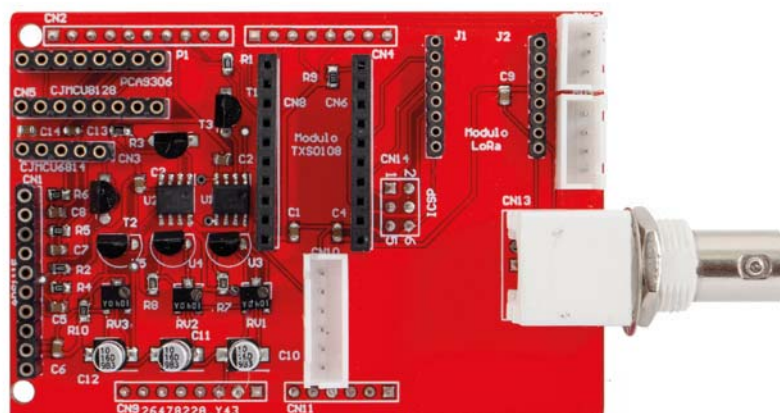
Le uscite del MiCS-6814 sono collegate ciascuna ad un circuito generatore di corrente costante costituito da un amplificatore operazionale, un



← **Fig. 15**
Generatore
di corrente
costante.

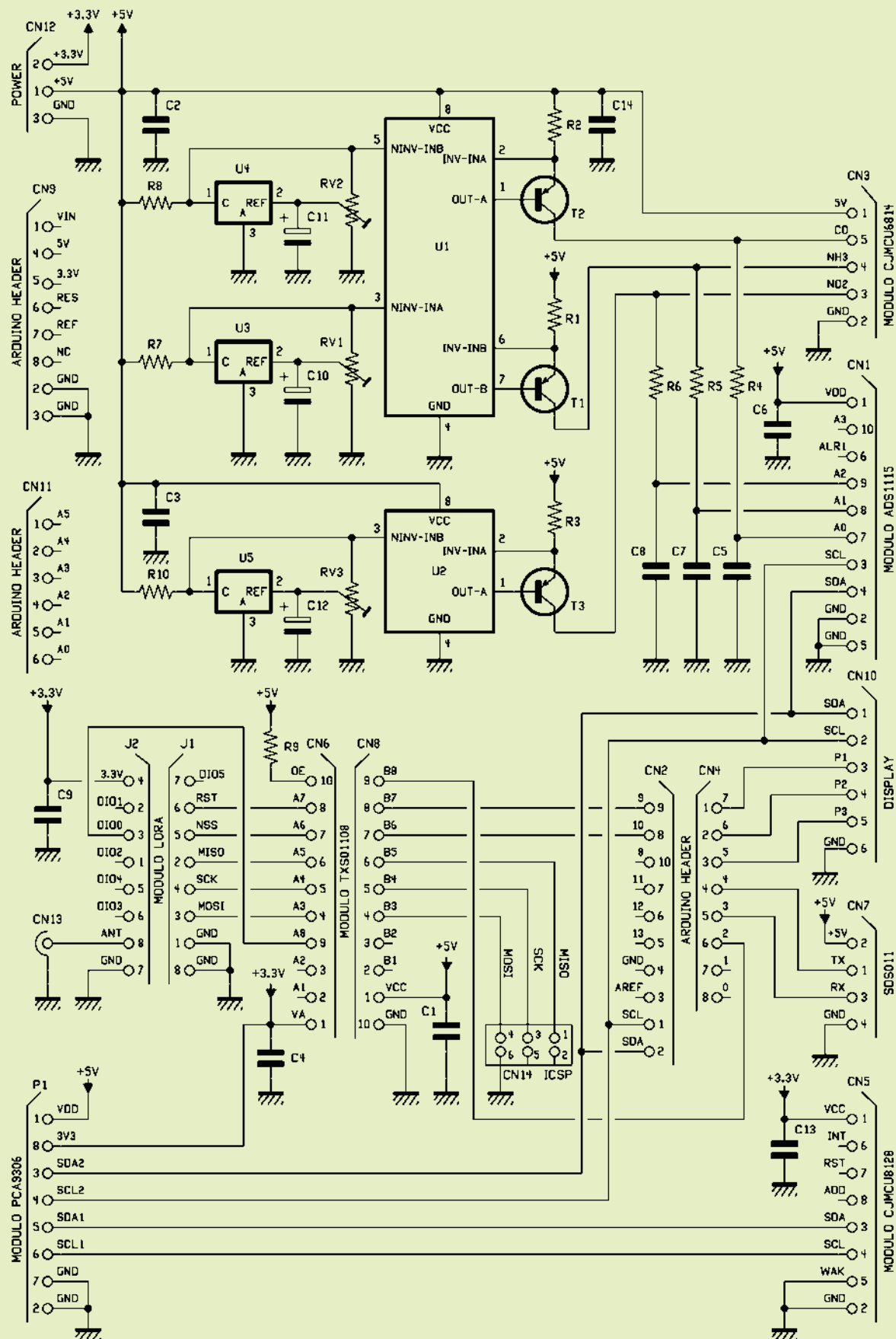
transistor tipo PNP, un diodo Zener di precisione regolabile ed una rete di resistenze che servono a gestire la retroazione di controllo (**Fig. 15**); anche questo generatore di corrente è a bordo dell'Analyze Shield di **Fig. 16**.

↓ **Fig. 16**
Scheda
AnalyzeShield.



schema elettrico

ANALYZESHIELD



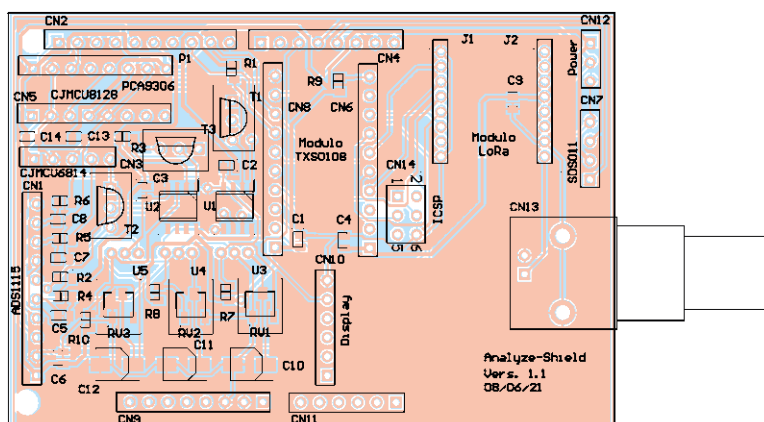
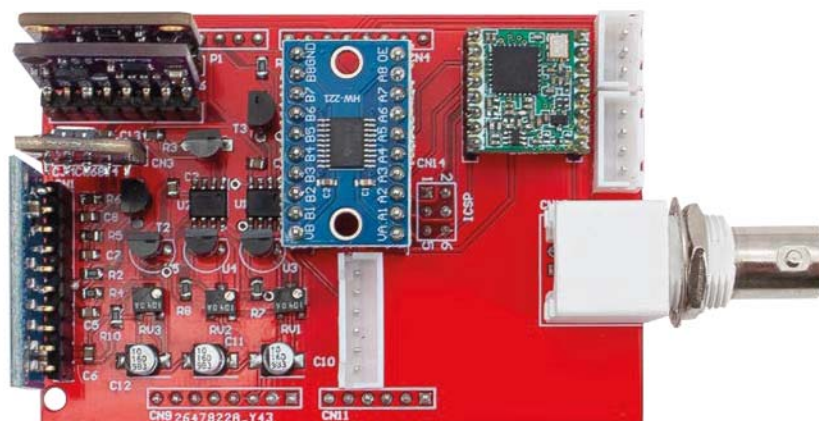
piano di montaggio ANALYZESHIELD

Elenco Componenti:

R1, R2: 2 Mohm (0805)
 R3: 22.3 kohm (0805)
 R4, R5, R6: 1 kohm (0805)
 R7, R8, R10: 220 ohm (0805)
 R9: 10 kohm (0805)
 RV1, RV2, RV3: 100 kohm
 trimmer multigiro
 C1, C2, C3, C4, C5, C6, C7, C8, C9,
 C13, C14: 0.1uF 16V multistrato SMD
 (0805)
 C10, C11, C12: 10uF 16V elettrolitico
 T1, T2, T3: BC32725TA
 U1, U2: LM358MX_NOPB
 U3, U4, U5: TL431CZ

Varie:

- CN1, CN6, CN8: Strip femmina 10 poli
- CN2: Strip maschio 10 poli
- CN3: Strip femmina 5 poli
- CN4, CN9: Strip maschio 8 poli
- CN5, P1: Strip femmina 8 poli
- CN7: XHPX-4, 4-Pin
- CN10: XHPX-6, 6-Pin
- CN11: Strip maschio 6 poli
- CN12: XHPX-3, 3-Pin
- CN13: connettore BNC da stampato femmina
- CN14: strip 3x2 femmina
- J1, J2: Strip femmina 8 poli 2mm tornito
- Circuito stampato S1575 (80x55 cm)



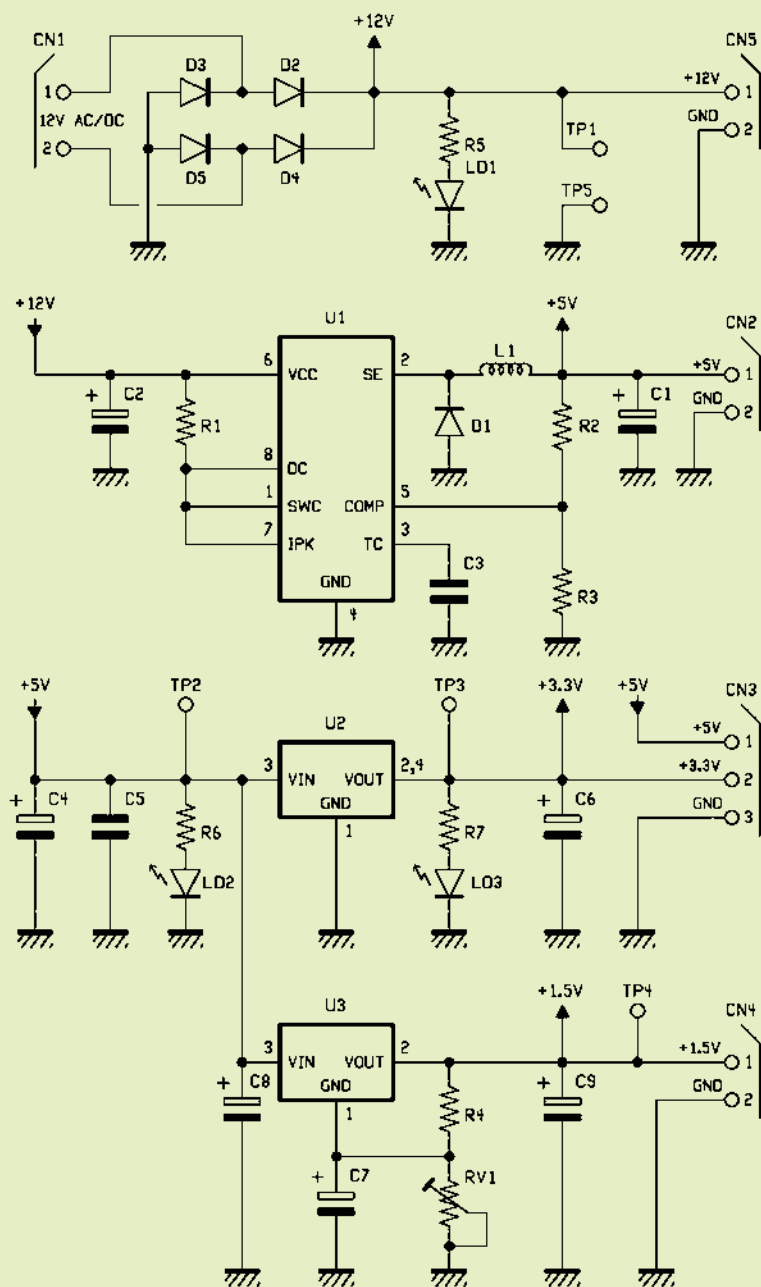
Nel nostro caso, per realizzare lo schema di principio del generatore di corrente, abbiamo utilizzato un amplificatore operazionale tipo LM358, un transistor BC327, un TL431, dal quale otteniamo

una tensione di riferimento stabile.

La regolazione della corrente impressa sarà effettuata tramite la variazione del trimmer usato per impostare la tensione di riferimento. Per



Fig. 17
Modulo di
calibrazione.



ottimizzare e facilitare la corretta impostazione dei tre generatori di corrente, sono stati realizzati due moduli di calibrazione come quello mostrato nella **Fig. 17**. Questi saranno inseriti prima del modulo MiCS-6814, nello stesso connettore e, se facciamo riferimento alla **Fig. 15**, sostituiscono la resistenza RS. Ogni modulo di configurazione monta su ogni singola uscita (NO₂, NH₃, CO) una resistenza corrispondente al valore minimo o massimo, definiti dal datasheet del MiCS-6814, che ci permette di

simulare le resistenze dei sensori. Su un modulo di configurazione verranno montate le resistenze da 1 k Ω (R1) per il canale dedicato all'NO₂, 10 k Ω (R3) per l'NH₃ e 100 k Ω (R2) per il CO, corrispondenti ai valori di resistenza minimi possibili. Sul secondo modulo di configurazione verranno installate le corrispondenti resistenze da 20 k Ω per il canale NO₂, 1 M Ω per l'NH₃ come per il CO.

Come già accennato, occorrerà impostare una corrente di 1 μ A per il canale CO e per il canale NH₃, mentre per l'NO₂ imposteremo 100 μ A. I moduli di configurazione sono dotati di test point con i quali possiamo misurare la tensione ai capi delle resistenze R1, R2 e R3. In **Tabella 4** sono elencati i valori di tensione da impostare sui test point corrispondenti.

Occorre, in questa fase, non trascurare i tempi di regimazione termica dei circuiti, per cui dovremo inserire il modulo di taratura, impostare le tensioni di **Tabella 4** e attendere 30-40 minuti, controllando le tensioni e agendo eventualmente sui trimmer RV1-2-3.

A questo punto è possibile sostituire il modulo di taratura e riaggiustare le tensioni presenti sui test point. Questa procedura può essere ripetuta più volte fino a quando saremo riusciti ad ottenere una buona configurazione mantenendo il più possibile uguali le tensioni presenti sui test point con i valori indicati in **tabella 4** per entrambi i moduli di taratura. Le tensioni presenti ai capi dei sensori NO_x verranno lette dall'ADS1115 (cod. ADS1115MOD), un convertitore A/D a 16 bit, quattro canali, interfacciato su bus I²C e alimentato a 5V.

La rete I²C è stata suddivisa su due livelli di tensione per poter raggiungere il modulo CJMCU-8128 alimentato a 3,3V e il modulo ADS1115 (**Fig. 18**) alimentato a 5V. Il display sarà collegato all'AnalyzeShield mediante un PCF8574 che utilizza la rete I²C a 5V per la gestione di un LCD2004 a quattro righe e venti colonne. Sulla scheda di analisi (Analyze Shield) è presente il collegamento per il modulo LoRa, implementato attraverso il traslatore di segnali TXS0108 con cui realizzeremo il bus SPI a 3,3V.

Oltre ai segnali MISO, MOSI e SCK prelevati dal connettore ICSP di Arduino Uno, sono stati utilizzati i segnali NSS, Reset e DIO0 rispettivamente dai pin 10, 9 e 2 di Arduino. Come accennato precedentemente tutti i segnali elencati sono stati ricondotti a 3,3 V attraverso il TXS0108.

Le soluzioni adottate hanno permesso di risparmiare un discreto numero di I/O di Arduino, riuscendo così ad implementare tutte le funzioni

piano di montaggio scheda ALIMENTAZIONE

Elenco Componenti:

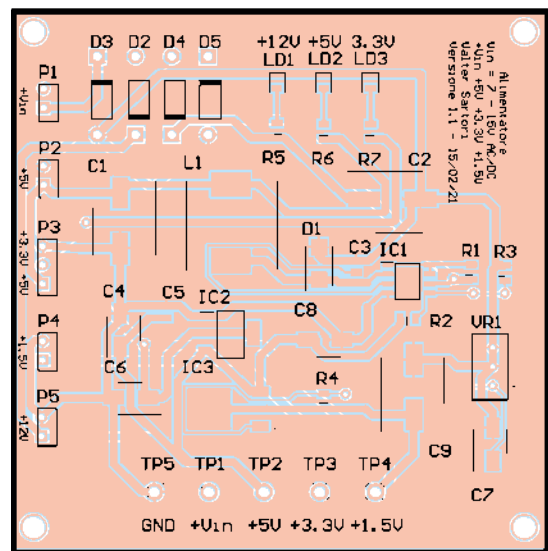
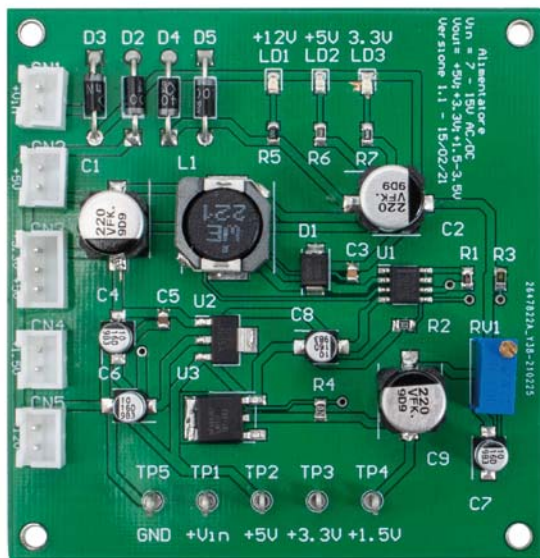
R1: 0.22 ohm
R2: 3 kohm
R3: 1 kohm
R4: 120 ohm
R5: 1.2 kohm
R6: 560 ohm
R7: 360 ohm
RV1: 470 ohm
C1, C2, C9: 220uF 35V

elettrolitico SMD
C3: 220pF 16V multistrato
C4, C6, C7, C8: 10uF 16V elettrolitico SMD
C5: 0.1 uF 50V multistrato SMD
D1: MBR5140T3 diodo Schottky
D2, D3, D4, D5: 1N4007
L1: 220uH
LD1, LD2, LD3: LED verde SMD
U1: MC34063ADR2G

U2: LM1117MP-3.3_NOPB
U3: LM1117IDTX-ADJ_NOPB
TP1÷TP5: TEST PONT

Varie:

- CN1, CN3, CN4, CN5: XHPX-2, 2-Pin
- CN2: XHPX-3, 3-Pin
- Circuito stampato S1574 (73x70 mm)



necessarie alla gestione del sistema di monitoraggio ambientale. Al modulo LoRa (**Fig. 19**) è stato assegnato il compito di trasmettere a distanza via radio i rilievi dei parametri ambientali effettuati. La scelta è ricaduta sul modulo RFM95 Low Power Long Range Transceiver Module (cod. RFM95-868), che permette la trasmissione a distanze particolarmente notevoli con un'alta immunità alle interferenze. All'RFM95 sono stati saldati due header maschi a 10 pin con passo 2 mm per permettere una installazione sulla scheda AnalyzeShield con possibilità di rimozione in caso di guasto senza dissaldature che comprometterebbero la funzionalità della scheda stessa. L'uscita del modulo RFM95 è stata collegata ad un BNC saldato sulla scheda e collegato ad una antenna esterna. La trasmissione dei dati raccolti avviene ogni dieci minuti ed è la media risultante

dei campioni rilevati con cadenza 60 s.

In questo progetto non è stato introdotto alcun modulo RTC per il controllo di data e ora, in quanto sarà affidato al sistema ricevente il compito di as-

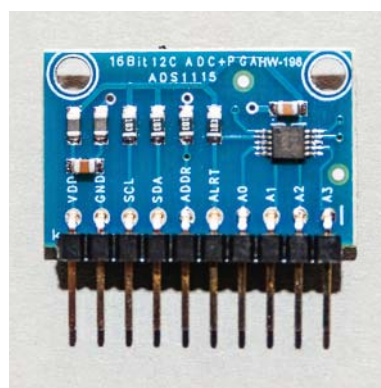


Fig. 18
Il modulo
ADS1115.

Fig. 19
Modulo LoRa.



segnare ai dati in arrivo le informazioni necessarie per la memorizzazione a database.

SCHEMA ALIMENTATORE

La scheda (Fig. 20) fornisce le tensioni utili all'alimentazione dei circuiti a 5V, 3,3V e fornisce una tensione variabile tra 1,5 e 4 V per regolare il flusso dell'aria da analizzare modificando la velocità della ventola del condotto di ingresso.

In ingresso è prevista la possibilità di alimentare la scheda con tensioni comprese tra 7 e 15 V. Nel nostro caso abbiamo adottato un'alimentazione in ingresso sul connettore CN1 a 12Vcc.

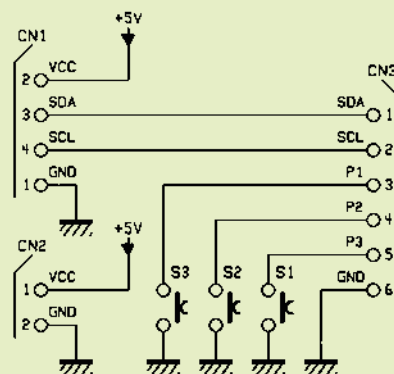
La scheda Arduino Uno verrà alimentata attraverso il connettore CN5. L'uscita a 5V è stata realizzata sfruttando l'MC34063 (U1) un DC/DC converter in configurazione step-down.

I 5V prodotti dal convertitore step-down costituiscono non solo una prima rete di alimentazione, ma anche l'ingresso sia di un LM1117MP (U2) con uscita a 3.3V fissa, sia l'ingresso di un ulteriore LM1117-ADJ (U3) con uscita variabile tra 1.5 e 4V regolata tramite il trimmer multigiri da 470 Ω (RV1). La scheda dispone di una serie di test-point (TP1÷TP5) che permettono di verificare il valo-

Fig. 20
Scheda
alimentatore.



schema elettrico scheda DISPLAY



re e la presenza di ogni singola tensione. I LED LD1÷LD3 permettono una verifica visiva della presenza di ogni singola alimentazione. In ingresso troviamo i diodi D2, D3, D4 e D5 che permettono un ingresso sia in alternata sia in continua proteggendo altresì i circuiti elettronici da un eventuale scambio di polarità dell'alimentazione in ingresso.

Le alimentazioni in uscita vengono distribuite attraverso connettori JST-XH: il connettore CN2 fornisce l'alimentazione a 5V per la scheda display, il connettore CN3 alimenterà la scheda Analyze-Shield con tensioni a 5V e 3,3V. In ultimo il connettore CN4 alimenterà la ventola con tensione variabile da 1,5V a 4V.

SCHEMA DISPLAY

Per poter interfacciare la AnalyzeShield al display e permetterci di interagire localmente con il sistema, è stata realizzata una scheda che accetta in ingresso i segnali del bus I²C a 5V per il pilotaggio del display LCD2004 (quattro righe per venti colonne) e tre ingressi digitali di Arduino Uno collegati a tre pulsanti che serviranno per la configurazione dei coefficienti di calcolo delle concentrazioni volumetriche degli inquinanti e per eventuali offset di aggiustamento per le misure meteorologiche (Fig. 21).

Il display LCD2004 è equipaggiato con un adattatore seriale per bus I²C e bus LCD, utile proprio a consentire il corretto pilotaggio.

REALIZZAZIONE PRATICA

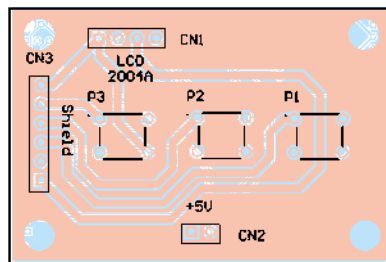
Partiamo dalla scheda alimentatore, che è es-

piano di montaggio scheda DISPLAY



Elenco Componenti:

CN1: JST-XHPX, 4-Pin
CN2: JST-XHPX, 2-Pin
CN3: JST-XHPX, 6-Pin



P1, P2, P3: Microswitch da stampato
Varie:

- Circuito stampato S1576 (34x50 cm)

senzialmente composta da componenti SMD che impongono di selezionare una scaletta di priorità nella saldatura dei componenti. Il circuito stampato può essere facilmente realizzato utilizzando il servizio PCBProduction di Futura Elettronica. Risulta essenziale iniziare il montaggio saldando dapprima gli integrati U1+U3. A seguire si possono saldare le resistenze e i LED. Seguendo questa scaletta risulta molto più facile posizionare la punta del saldatore sulle piazzole senza incontrare elementi di ingombro.

Per ultimo conviene saldare i condensatori elettrolitici, l'induttanza L1, il trimmer RV1 e, infine, i test point e i connettori.

Per verificare la funzionalità della scheda possiamo collegare il connettore CN1 (Vin) ad una sorgente di alimentazione a 12V DC e controllare, con l'ausilio di un multimetro, le tensioni presenti sui TP1+4. Su TP4 occorre verificare, mediante la rotazione del trimmer RV1, il range di tensione, che varierà tra 1,5V e 4V.

Per quanto riguarda la scheda AnalyzeShield, il progetto prevede l'utilizzo di componenti SMD per cui è consigliabile saldare dapprima gli integrati U1 e U2. A seguire si dovranno saldare le resistenze e le capacità con package 0805.

Gli ultimi componenti SMD che verranno saldati saranno i trimmer RV1-2-3 e i condensatori elettrolitici C10-11-12. Terminati i componenti a saldatura superficiale si può passare ai transistor T1-2-3 e agli zener U3-4-5. Infine potranno essere saldati i connettori, dagli header di collegamento con l'Arduino Uno ai connettori dedicati ai vari moduli precedentemente descritti.

Per ultimi potranno essere saldati i connettori XHPX per l'interfacciamento con le altre schede che compongono il progetto.

La scheda Display ha il compito di interfacciare il display LCD2004 mediante bus I²C e supportare i tre pulsanti che serviranno per la configurazione del sistema. La board supporta tre pulsanti tattili e tre connettori tipo XHPX di cui uno per l'interfacciamento con la AnalyzeShield, l'altro per il collegamento con il display e l'ultimo per l'alimentazione a 5V. I tre pulsanti tattili denominati P1, P2 e P3 verranno utilizzati per la configurazione del sistema.

Il collegamento del polverimetro Scheda SDS011 non presenta particolari difficoltà. Utilizzando un terminale JST-XHPX-7 (7 pin femmina passo 2.54 mm) dovranno essere cablati i pin dedicati all'alimentazione a 5 V e al collegamento seriale. Nel realizzare il cavo di interfaccia tra l'SDS011 e la scheda Analyze-Shield occorre

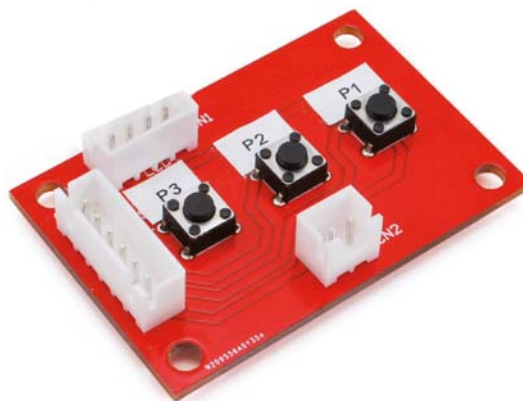
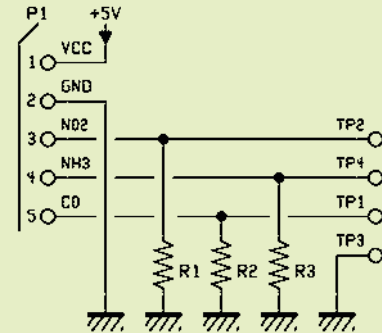


Fig. 21
Scheda display con i tre pulsanti.

Tabella 5

PIN	NOME	DESCRIZIONE
1	NC	Non connesso
2	1 μ m	Uscita PWM PM2.5
3	5 V	Alimentazione
4	2.5 μ m	Uscita PWM PM10
5	GND	Ground
6	RX	UART RX (collegare al pin 1 TX di CN7)
7	TX	UART TX (collegare al pin 3 RX di CN7)

schema elettrico scheda CALIBRAZIONE



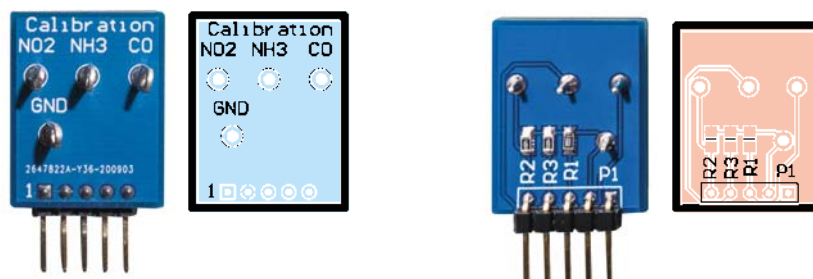
prestare attenzione al collegamento seriale che prevede l'uscita RX del polverimetro connessa al pin TX dell'Arduino e l'ingresso TX al pin RX di Arduino. Nella **Tabella 5** trovate la piedinatura dell'SDS011, con indicate, nella colonna riservata alle descrizioni, le relative connessioni previste nel nostro circuito.

SCHEDE DI CALIBRAZIONE

I moduli utilizzati per la taratura della corrente co-

stante impressa nei tre canali di misura relativi alla CO, NH3 e NO2 sono quelli mostrati nella **Fig. 17** nelle pagine precedenti. Con due di queste schede simuleremo le resistenze minime e massime riguardanti il Red Sensor, l'Ox Sensor e l'NH3 Sensor. Verranno predisposte due schede di calibrazione, una con le resistenze massime e l'altra con le resistenze minime. Tornando alla **Tabella 4**, si possono rilevare i valori delle resistenze per ogni scheda e i valori di tensione a cui appostare i riferimenti.

piano di montaggio scheda CALIBRAZIONE



Elenco Componenti (vedi Tabella 4):

R1: 1 kohm / 20kohm (0805)
R2 100 kohm / 1Mohm (0805)
R3 10 kohm / 1Mohm(0805)

Varie:

- P1: Strip maschio 5 poli
- TP1-4: test point
- Circuito stampato S1577

Per il montaggio dare sempre la priorità ai componenti SMD e poi passare alla saldatura dei test point. Per ultimo bisogna saldare il connettore header con passo 2,54 e pin piegati a 90°.

REGOLAZIONE FLUSSO D'ARIA

In precedenza abbiamo accennato all'uscita a tensione variabile dell'alimentatore per collegare un ventilatore e produrre uno stream di misura.

Il ventilatore adottato è un brushless fun DC da 5V, con telaio quadrato delle dimensioni di 40x40 mm.

Installandolo alla base di un contenitore interponendo tra il ventilatore e la presa d'aria un filtro a maglie larghe, atto ad evitare l'ingresso di insetti o pulviscolo, regolando l'alimentazione ad una tensione tra 2,5 - 3 V otteniamo un flusso d'aria che non interferisce con le misure di pressione atmosferica, temperatura e umidità e nel contempo crea un campione di misura ottimo per le analisi del modulo MiCS6814.

Osservando la disposizione dei due moduli MiCS6814 e CJMCU8128, allineati parallelamente uno con l'altro e con i componenti affacciati, si viene a creare un corridoio che verrà lambito dal flusso d'aria creato dalla ventilazione. La distanza tra il ventilatore e la scheda AnalyzeShield dovrà essere ridotta al minimo essenziale.

Per la misura delle polveri occorrerà installare un tubo in silicone con diametro interno 6 mm tra l'ingresso dell'aria dell'SDS011 e una presa d'aria simile a quella del ventilatore corredata da un filtro a maglie larghe (1 mm). Il ventilatore incorporato nel polverimetro preleverà i campioni d'aria direttamente dall'esterno, così da evitare eventuali afflussi che possano alterare la misura.

Nell'immagine di **Fig. 22** si possono osservare i dettagli costruttivi, con la disposizione all'interno dell'armadio da noi utilizzato dei vari componenti che permettono il prelievo e l'analisi del campione di aria.

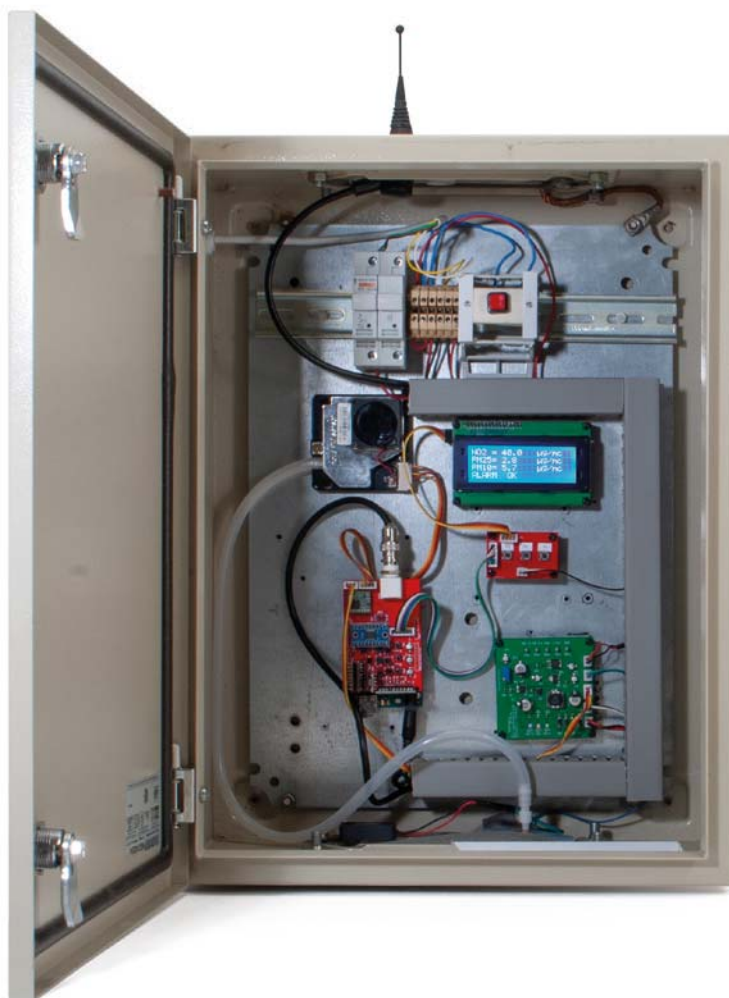


Fig. 22
L'armadio in cui abbiamo inserito il prototipo.

CONCLUSIONI

Bene, per il momento ci fermiamo qui, dopo avervi descritto l'hardware della nostra centralina di monitoraggio della qualità dell'aria.

Riprenderemo il discorso nella prossima e conclusiva puntata, esponendovi il firmware ed il software di gestione da PC del nostro progetto. □



Cosa occorre?

I componenti principali utilizzati in questo progetto sono disponibili presso Futura Elettronica. I gerber delle schede 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>

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

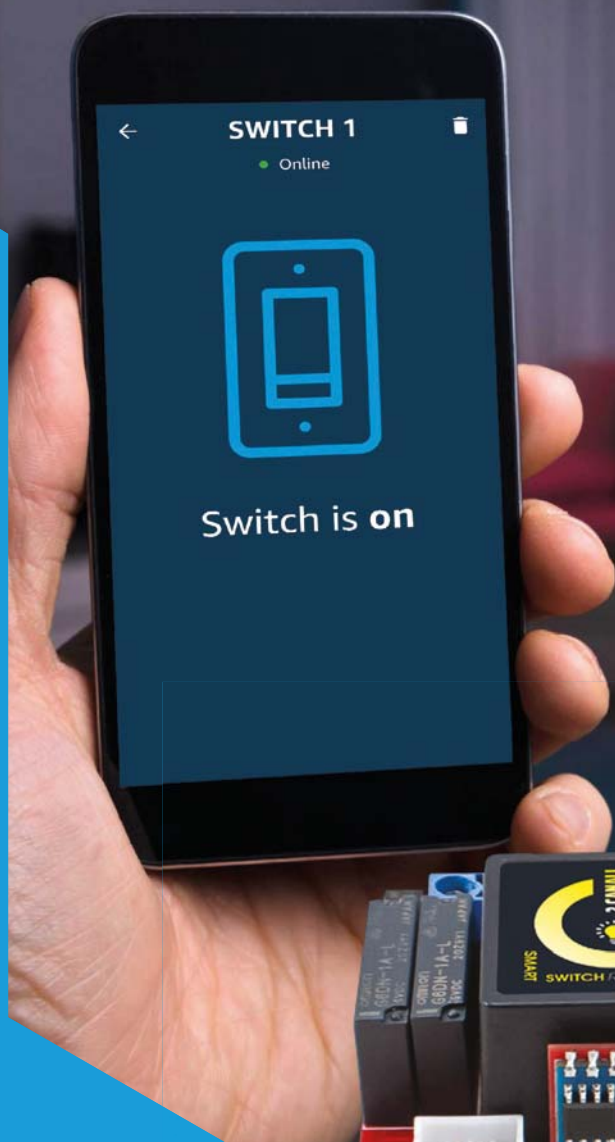
Sanificatore automatico con LED UVC, particolarmente indicato in tutte le situazioni dove si rende necessario igienizzare, sanificare e comunque disinfettare oggetti potenzialmente depositari di particelle infettive (virus, batteri, ecc.) come mascherine chirurgiche, utensili, occhiali, chiavi, ecc. 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





DOUBLE SMART SWITCH

di MAURIZIO ŠKERLIČ



Doppio interruttore IoT controllato da smartphone o mediante assistente vocale Amazon Alexa o Google Assistant. Configurabile da browser, è proposto sia con uscita a relé, sia con uscita a SSR.

Con l'articolo dedicato allo Smart Switch, proposto nel fascicolo n° 260, abbiamo esplorato e messo in pratica la possibilità di creare un interruttore singolo a controllo sia touch che da assistente vocale; di quell'interruttore "smart", in questo articolo vi proponiamo la versione "twin" a due canali, utile per controllare più dispositivi da un unico punto di comando. Il progetto battezzato "Double smart switch", viene proposto in due versioni: a relé elettromeccanico e a relé allo stato solido (SSR) che è più silenzioso e più veloce nella commutazione. Inoltre, grazie a circuiti interni per il rilevamento del passaggio della semionda per lo zero volt (ZCD), 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 più adatta alle nostre esigenze, prediligendo la versione con relé allo stato solido per pilotare carichi induttivi o capacitivi. Poiché proponiamo due versioni del Double Smart Switch, che si differenziano soltanto nel pilotaggio del carico in uscita, analizzeremo contemporaneamente lo schema elettrico di entrambe le versioni soffermandoci solo sulle differenze circuitali negli stadi di commutazione del carico in uscita. Lo schema elettrico è praticamente identico a quello dello Smart switch pubblicato in passato con la differenza che qui ci sono un doppio ingresso e una doppia uscita. Incominciamo partendo dal modulo ESP03, siglato U5 basato sul microprocessore Espressif ESP8266. Il circuito viene alimentato da un alimentatore switching da 5V 1A, nello schema elettrico siglato TR1, la cui tensione viene abbassata a 3,3V dal regolatore di tensione NPC1117, nello schema elettrico siglato U4. La tensione a 5V è filtrata dal condensatore C4 e C5 mentre quella a 3,3V dai condensatori C6 e C7 indispensabili per evitare reset o crash di sistema casuali. I piedini 10 (GPIO 12) e 11 (GPIO 13) 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. I piedini 1 (IO0) e 2 (IO1) di U1 sono configurati come porte *OUTPUT* mentre i piedini 3 (IO2) e 5 (IO3) sono configurati come porta *INPUT*. Queste sono mantenute ad alto livello dalle resistenze R5 e R6 e i segnali, proveniente dal collettore dei transistor presenti nei fotoaccoppiatori U2 e U3, sono filtrati dai condensatori C1 e C2. La tensione alternata a 230 V proveniente dai pulsanti collegati ai piedini 1 e 2 del connettore CON1 viene disaccoppiata grazie ai doppi diodi LED, dei fotoaccoppiatori U2 e U3 e alle resistenze limitatrici di corrente R1, R2 e R3, R4. Le resistenze R7 e R8, usate come resistenze

di pullup, mantengono alto rispettivamente le linee SCL e SDA del segnale I²C. Il pulsante SW1 collegato al piedino 9 (GPIO2) del modulo ESP03, mantenuto ad alto livello dalla resistenza R9, serve a reimpostare la password predefinita di sistema "123456789", in caso dimenticaste quella memorizzata. I piedini 1 e 2 dell'integrato PCA9536 (U1), a seconda della versione SSR o EMR, pilotano una coppia di relé allo stato solido oppure una coppia di relé elettromeccanici mediante due doppi mosfet a canale N e canale P, nello schema elettrico siglati con T1 e T2. Nella versione a relé elettromeccanici in parallelo ad ogni bobina troviamo i diodi di riciclo (*ingl. Flyback*), siglati D2 e D3, i quali limitano le sovratensioni induttive, sopprimendo le extratensioni di segno opposto generate dalle bobine quando i relé vengono disattivati e servono a non far bruciare i due transistor T1 e T2.

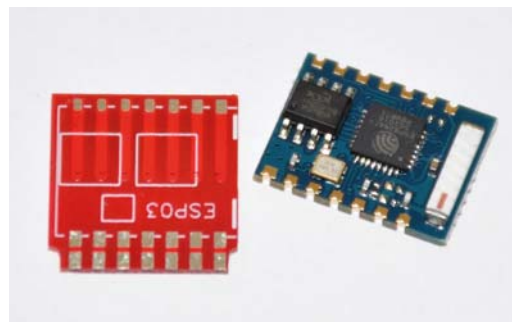
È possibile calcolare l'extratensione utilizzando la seguente formula:

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

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 3 e 5 (OUTPUT1) e i piedini 4 e 5 (OUTPUT2), mentre la tensione di rete, neutro e fase, viene collegata rispettivamente ai piedini 5 e 6 di CON1. Il circuito è protetto dalle 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 elettromeccanici, in parallelo ai contatti, sono stati inseriti degli snubber RC formati da R17, C12 e R18, C13. Questi hanno una duplice funzione, sopprimono l'arco sui contatti proteggendoli dalla precoce

➔ **Fig. 1**
Montaggio del
modulo ESP03 in
posizione verticale
sul PCB.



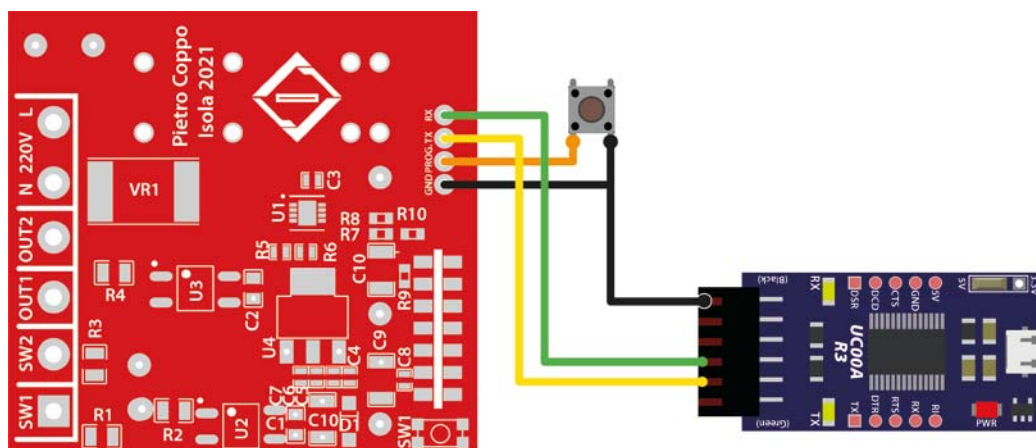


Fig. 2
Collegamento del
programmatore
seriale al Double
smart switch.

usura con carichi induttivi o capacitivi e riducono i disturbi elettromagnetici durante la commutazione. Nella versione a relé SSR, gli snubber non sono necessari poiché integrati all'interno dei relé allo stato solido, inoltre, grazie a un circuito zero crossing, il carico viene acceso sempre al passaggio della semionda per lo zero diminuendo i disturbi e gli stress elettrici su di esso.

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 watts) 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. 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. Come al solito iniziate a saldare i semiconduttori e successivamente tutti i componenti passivi, facendo attenzione al corretto orientamento dei semiconduttori come visibile nel piano di montaggio. Nel saldare i componenti passivi stagnate prima quelli con case più piccolo e successivamente quelli con case più grande. Per ottimizzare lo spazio il modulo ESP03, che nello schema elettrico è siglato U5, andrà prima stagnato sul PCB adattatore e successivamente montato verticalmente come visibile in **Fig. 1**. Per ultimi andranno montati i componenti a foro passante.

Una volta terminato il montaggio di tutti i componenti consigliamo sempre di pulire il PCB dal fluxante con alcol isopropilico. Con una lente di ingrandimento controllate attentamente che non si siano formati ponticelli di stagno tra i piedini dei componenti in particolar modo tra i piedini di U1. A questo punto potete alimentare il prototipo, seguendo il cablaggio illustrato nello schema elettrico.

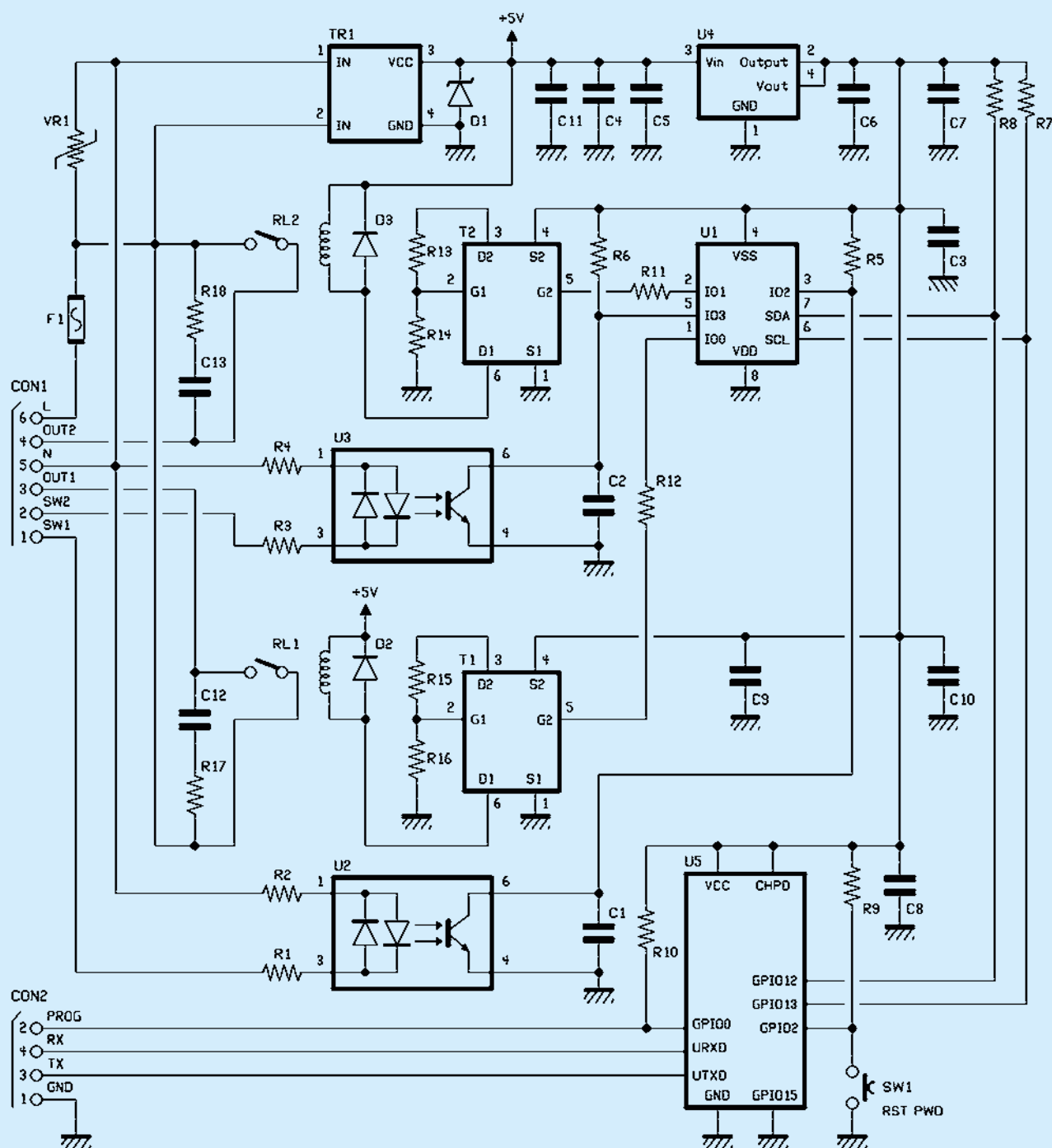
Come sempre, e non ci stancheremo di ripeterlo, non toccate con le mani nude i circuiti quando sono allacciati alla rete elettrica a 230 V, altrimenti rischiate di rimanere folgorati. Siccome gli impianti elettrici domestici per legge hanno l'interruttore differenziale, impropriamente chiamato "salvavita", molti di voi pensano che questo interverrà e lo proteggerà da eventuali scosse. Il differenziale scatta se ci sono dispersioni di corrente verso terra, quindi se toccate con le dita la fase e il neutro probabilmente il differenziale non interverrà e voi vi becherete una scossa che potrebbe essere anche mortale. La prudenza non è mai troppa.

PROGRAMMAZIONE E COLLAUDO

Poiché la configurazione e il collaudo dei due circuiti qui proposti sono identici a quella dello Smart Switch, rimandiamo i dettagli all'articolo pubblicato nel fascicolo n° 260, riportando in questo solo i passaggi più importanti. La programmazione del modulo ESP03 deve essere effettuata mediante un adattatore USB per ESP8266 oppure con un qualsiasi convertitore USB TTL, purché compatibile con logica a 3,3 V, usato per programmare i moduli Arduino Mini. Per il collegamento fate riferimento alla **Fig. 2**. Ricordatevi di collegare l'uscita TX del converti-

CARATTERISTICHE TECNICHE

- **Dimensioni ridotte**
40 x 43 x 21 mm
- **Montaggio in scatola da incasso oppure su supporti Vima, Gewiss o BTicino**
- **Adatto per carichi resistivi o induttivi**
- **Protezione termica, da sovracorrenti e sovratensioni**
- **Timer spegnimento automatico**
- **Impostazione in cinque modalità di funzionamento**
- **Configurazione tramite interfaccia web**
- **Collegamento alla rete WiFi domestica**
- **Protezione con password**
- **Reset password tramite pulsante**
- **Compatibile con gli assistenti vocali Amazon Alexa e Google Assistant**
- **Carico massimo 350 W versione SSR e 550W versione a relé elettromeccanico**



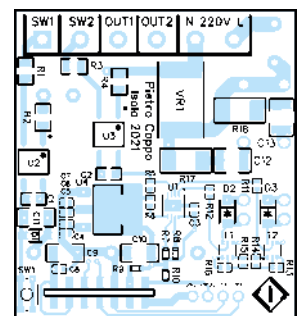
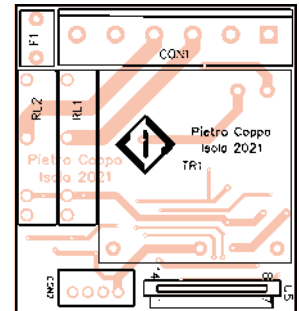
tore all'entrata RX pin 3 del connettore CON2 e l'entrata RX del convertitore all'uscita TX pin 4 di CON2. Per mettere il Double smart switch in modalità di programmazione premete il pulsante *Prog*, collegato al pin 2 del connettore CON2, prima di allacciarlo alla corrente elettrica, rilasciandolo dopo un secondo. Per il corretto cablaggio del pulsante sul connettore CON2 fate riferimento alla stessa **Fig. 2**. A scopo informativo diciamo

che lo sketch è stato compilato utilizzando il package core ESP8266 versione 2.7.4 e le librerie "*lotwebconf*" versione 3.2.0, "*SinricPro*" versione 2.9.10, "*SparkFun_PCA9536_Arduino_Library*" versione 1.2.2 e "*Arduino/son*" versione 6.18.5. Fate attenzione che la versione del package core dell'ESP8266 in Arduino Ide sia aggiornato almeno alla versione 2.7.4, altrimenti avrete problemi di funzionamento con la libreria *lotWebConf*. Al

| piano di montaggio SWITCH A RELÉ

Elenco Componenti:

R1, R2, R3, R4: 200 kohm 300 V (0805)	D1: SMF5.0A
R5, R6: 470 kohm (0402)	D2, D3: LL4148 (SOD123FL)
R7, R8: 2,2 kohm (0603)	T1, T2: BSD235CH
R9, R10: 10 kohm (0603)	VR1: V430CH8
R11, R12: 120 ohm (0402)	TR1: RAC05-05
R13, R15: 10 ohm (0402)	F1: F 3,15 A
R14, R16: 10 kohm (0402)	U1: PCA9536TK
R17, R18: 100 ohm 300 V (2010)	U2, U3: TLP184
C1, C2: 1 µF 6,3 V (0603)	U4: NCP1117ST33T3G
C3, C4, C6, C8: 100 nF (0402)	U5: ESP-03
C5, C7: 10 µF 6,3 V (0402)	RL1, RL2: G6DN-1A-SL DC5
C9, C10: 100 µF 6,3 V (1206)	CON1: Morsettiera 6 poli P - 5 mm
C11: 220 µF 6,3 V (1206)	CON2: DIP-1x4 P - 2 mm
C12, C13: 100 nF 500 V (1210)	SW1: Pulsante SMD 4.6 X 2.8 mm
	Varie
	- Circuito stampato S1587 (41x37 mm)



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 verrà risolto.

All'inizio dello sketch ci sono delle righe commentate che servono per impostare la compilazione del sorgente. Ad esempio se la memoria Flash montata sul modulo ESP03 è marchiata *Puya*, e

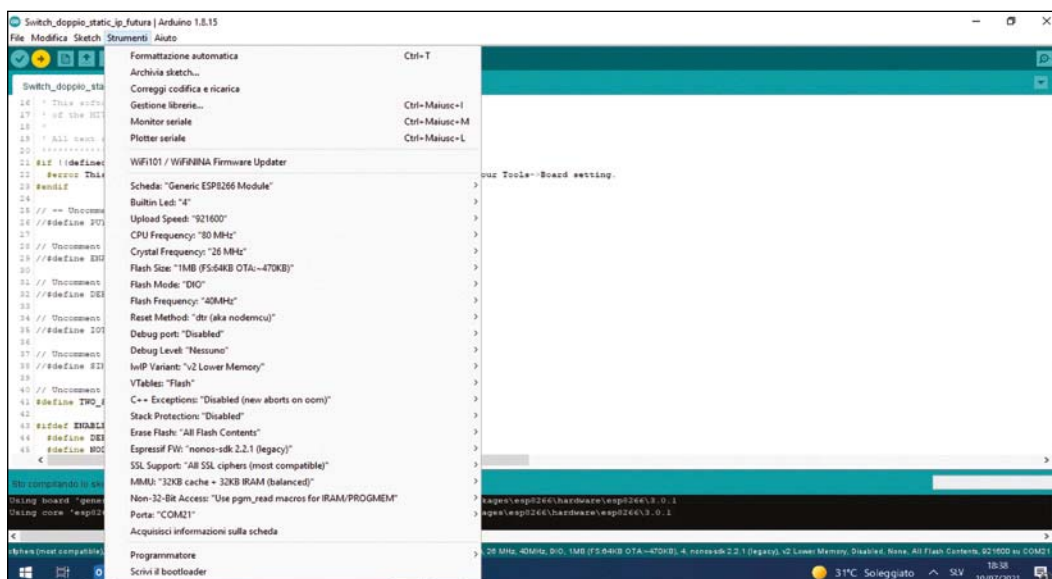
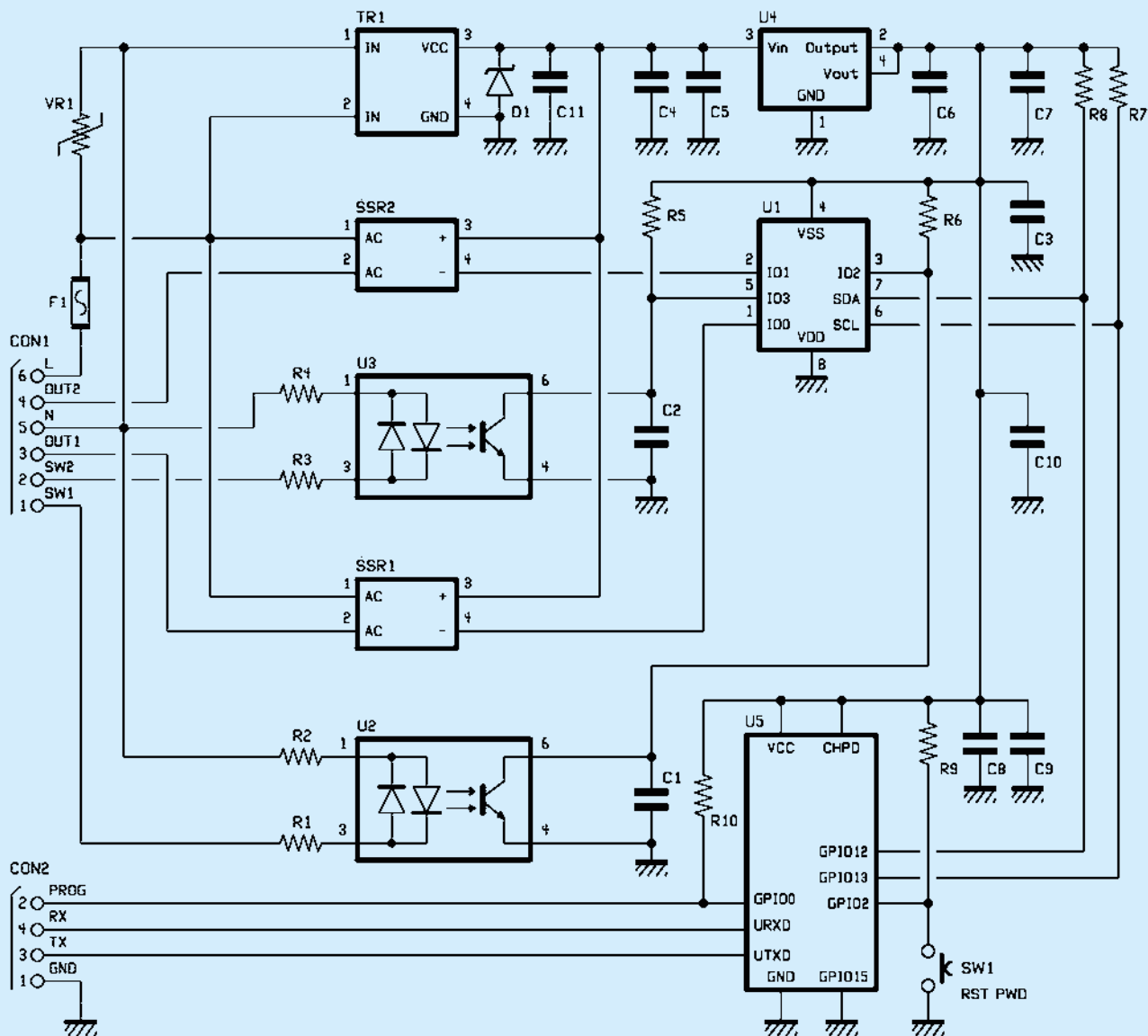


Fig. 3
Impostazioni di programmazione del modulo ESP03 in Arduino IDE.



volete abilitare il supporto di tale memoria in fase di compilazione, dovete togliere il commento `"/` davanti a `#define PUYA_SUPPORT 1`.
Se volete attivare il debug togliete il commento

davanti a `#define ENABLE_DEBUG`; e poiché state compilando lo sketch per il Double smart switch dovete togliere il commento davanti a `#define TWO_SWITCH`. La libreria *lotwebconf* di default fa visualizzare sulla seriale le chiavi del vostro account IoT in chiaro. Per motivi di sicurezza vi consigliamo di disabilitare tale opzione aggiungendo all'inizio del file *lotWebConfSettings.h*, che si trova nella cartella *SRC*, la seguente riga di testo `#define IOTWEBCONF_DEBUG_DISABLED`. Per chi fosse interessato, in **Fig. 3** riportiamo la configurazione di programmazione del modulo ESP03 usata da noi nell'ambiente di sviluppo Arduino IDE. Tra i file scaricabili dal sito web www.elettronica.in.it, nella cartella STL, trovate i file per la stampa 3D degli adattatori e frutti per i

Il prototipo del Double smart switch in versione a relé elettromeccanici.



| piano di montaggio

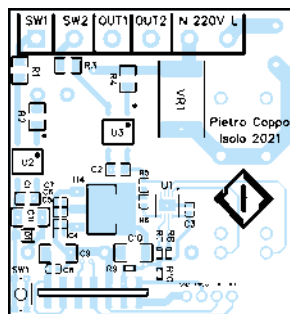
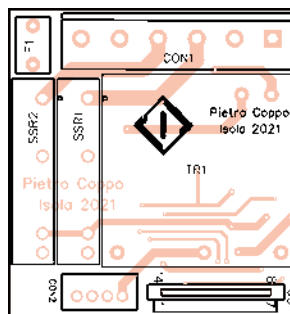
VERSIONE A SSR

Elenco Componenti:

R1, R2, R3, R4: 200 kohm 300 V (0805)
 R5, R6: 470 kohm (0402)
 R7, R8: 2,2 kohm (0603)
 R9, R10: 10 kohm (0603)
 C1, C2: 1 µF 6,3 V (0603)
 C3, C4, C6, C8: 100 nF (0402)
 C5, C7: 10 µF 6,3 V (0402)
 C9, C10: 100 µF 6,3 V (1206)
 C11: 220 µF 6,3 V (1206)
 D1: SMF5.0A
 VR1: V430CH8
 TR1: RAC05-05
 F1: F 1,5A
 U1: PCA9536TK
 U2, U3: TLP184
 U4: NCP1117ST33T3G
 U5: ESP-03
 SSR1, SSR2: G3MB-202P
 CON1: Morsettiera 6 poli P - 5 mm
 CON2: DIP-1x4 P - 2 mm
 SW1: Pulsante SMD 4.6 X 2.8 mm

Varie

- Circuito stampato S1588
 (41x39 mm)



supporti Vimar Plana, Vimar Idea, Gewiss System, Gewiss Playbus e BTicino living, che faciliteranno l'installazione del Double smart switch nel vostro impianto elettrico domestico. Fate attenzione al corretto orientamento dei coperchi dei contenitori, operando in modo che il forellino laterale sia posizionato sopra al pulsante SW1, per poterlo schiacciare. 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, spesso 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 sito www.futurashop.it.

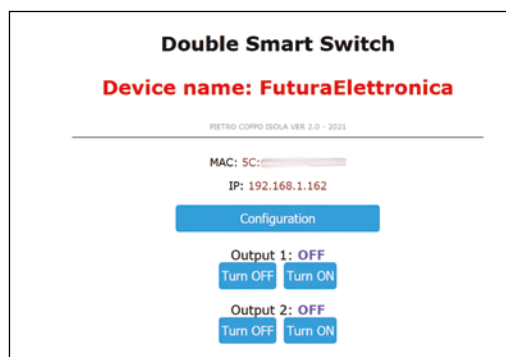
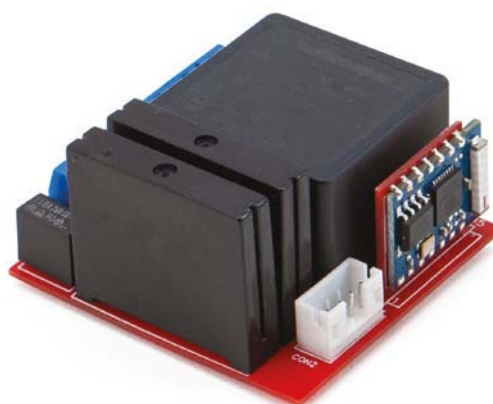
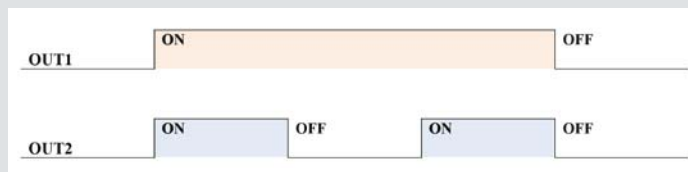


Fig. 4
Home page
del Double
smart switch.



Il prototipo
del circuito
nella versione
a due SSR.

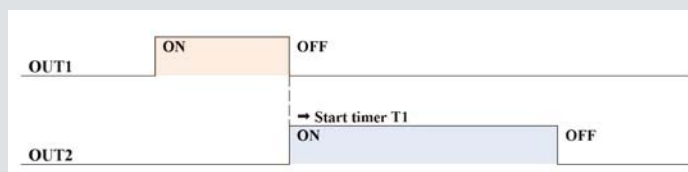
In questo riquadro, allo scopo di rendere visivamente comprensibile cosa comportano in output le modalità di funzionamento del Double smart switch descritte nell'articolo, proponiamo i diagrammi relativi allo stato delle uscite correlati tra loro nelle diverse (cinque) modalità operative previste e configurabili da interfaccia web.



MODE 1: Le due uscite sono indipendenti. L'interruttore SW1 comanda l'uscita OUT1, mentre l'interruttore SW2 comanda l'uscita OUT2.



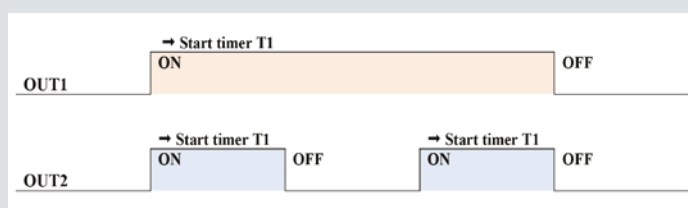
MODE 2: All'accensione dell'interruttore SW1 si attiva l'uscita OUT2. Il timer T1 parte allo spegnimento dell'interruttore SW1 e mantiene l'uscita OUT2 attiva per il tempo impostato in *Timer SW2*.



MODE 3: Allo spegnimento dell'interruttore SW1 si attiva l'uscita OUT2 che rimane attiva per il tempo impostato in *Timer SW2*.



MODE 4: All'accensione dell'interruttore SW1 si attiva l'uscita OUT2 e resta attiva per il tempo impostato in *Timer SW2*.



MODE 5: Le due uscite sono indipendenti e funzionano come temporizzatori. L'interruttore SW1 attiva l'uscita OUT1 che rimane accesa per il tempo impostato in *Timer SW1*, viceversa l'interruttore SW2 attiva l'uscita OUT2 che rimane accesa per il tempo impostato in *Timer SW2*. Con SW1 e SW2 si possono solo attivare i timer e non arrestarli. Se il pulsante viene premuto prima della scadenza del tempo, il timer viene fatto ripartire.

UTILIZZO DELLO SMART SWITCH

Dopo essere entrati nella pagina iniziale, per accedere alla pagina di configurazione del Double smart switch cliccate su "*Configuration*" (**Fig. 4**) e nelle due pagine seguenti 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** e **Fig. 6**.

I settaggi sono uguali a quelli dello Smart switch che abbiamo analizzato in modo approfondito nell'articolo già pubblicato. Per questo motivo andremo a elencare le varie impostazioni soffermandoci solo su quelle presenti esclusivamente in questa versione.

In "*Timer - SW1*" potete impostare il valore in minuti dell'autospegnimento della luce collegata all'uscita 1.

La casella di spunta "*Turn on SW1 at startup*" vi permette di scegliere se il carico sull'uscita 1 deve accendersi o rimanere spenta all'avvio.

In "*Timer - SW2*" potete impostare il valore in minuti dell'autospegnimento della luce collegata all'uscita 2.

La casella di spunta "*Turn on SW2 at startup*" vi permette di scegliere se il carico sull'uscita 2 deve accendersi o rimanere spenta all'avvio.

Il menu a tendina "*Switch mode*" vi permette di scegliere cinque diverse impostazioni di funzionamento.

1. "*SW1-SW2*" le due uscite funzionano in modo indipendente. Funzionamento normale.
2. "*SW1+2 → T*" all'accensione del relé 1 si accende anche il relé 2, allo spegnimento del relé 1 il relé 2 rimane attivo per il tempo impostato nella casella "*Timer - SW2*".
3. "*SW1 → SW2+T*" all'accensione del relé 1 il relé 2 rimane spento. Allo spegnimento del relé 1 si accende il relé 2 e rimane attivo per il tempo impostato nella casella "*Timer - SW2*".
4. "*SW1+T → SW2*" all'accensione del relé 1 si accende anche il relé 2, il quale rimane attivo per il tempo impostato nella casella "*Timer - SW2*".
5. "*SW1+T-SW2+T*" le due uscite lavorano come timer impostando il tempo nelle caselle "*Timer-SW1*" e "*Timer-SW2*".

Un utilizzo tipico delle modalità di funzionamento 2, 3 e 4 potrebbe essere quello dell'accensione della luce e della ventola del bagno, proposto in **Fig. 7**. Sull'uscita 1 (pin 2 di CON1) collegate la lampadina mentre sull'uscita 2 (pin 3 di CON1) collegate la ventola. Nella modalità 2, quando ac-

System configuration

Thing name
FuturaElettronica

AP password
[password icon]

WIFI SSID
[text input]

WIFI password
[password icon]

Startup delay (seconds)
30

Device IP

Enable Static IP: ☐

IP address
0.0.0.0

Subnet mask
0.0.0.0

Gateway
0.0.0.0

DNS
0.0.0.0

Iot Credentials

Gateway
0.0.0.0

DNS
0.0.0.0

Iot Credentials

Api Key
[text input]

Api Secret
[text input]

Device Id - SW1
[text input]

Device Id - SW2
[text input]

Settings

Switch mode
SW1-SW2

Timer - SW1 (0 - 1440 min)
0

Timer - SW2 (0 - 1440 min)
0

Turn on SW1 at startup: ☐

Turn on SW2 at startup: ☐

Fig. 5
Schermate di configurazione del Double smart switch.

Settings

Switch mode
SW1-SW2

Timer - SW1 (0 - 1440 min)
0

Timer - SW2 (0 - 1440 min)
0

Turn on SW1 at startup: ☐

Turn on SW2 at startup: ☐

Use toggle switch for SW1: ☐

Use toggle switch for SW2: ☐

Enable external switch: ☒

Enable assistant control: ☒

Enable web control: ☐

Reboot device on save: ☐

Apply

Firmware config version '2.0 PIETRO COPPO ISOLA 2021'

Fig. 6
Schermata della pagina web di configurazione del Double smart switch.

cenderete la luce del bagno si accenderà contemporaneamente anche la ventola, quando uscite e spegnerete la luce la ventola continuerà a girare per il tempo impostato in "Timer - SW2".

Se il rumore della ventola vi dà fastidio, selezionate il modo di funzionamento 3. All'accensione della luce la ventola non gira, quando uscite e spegnete la luce la ventola si accende e continua a girare per il tempo impostato in "Timer - SW2".

Con l'impostazione 4 accendendo la luce si accen-

Fig. 7
Cablaggio del Double smart switch all'interruttore del bagno.

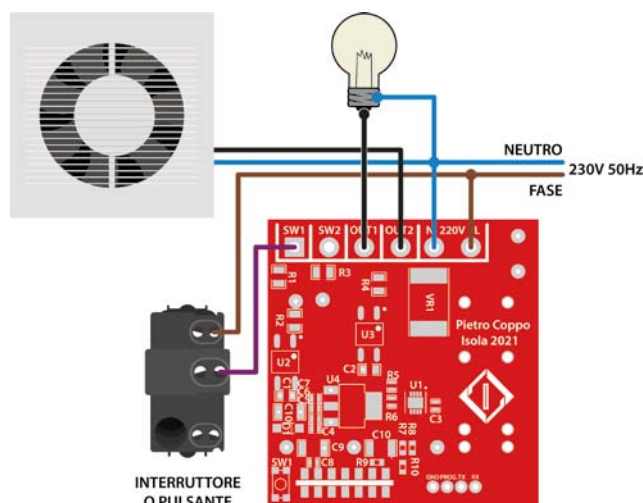


Fig. 8
Installazione del Double smart switch per il comando di due punti luce da più posizioni, con deviatori e invertitori (schema in alto) o con pulsanti (schema in basso).

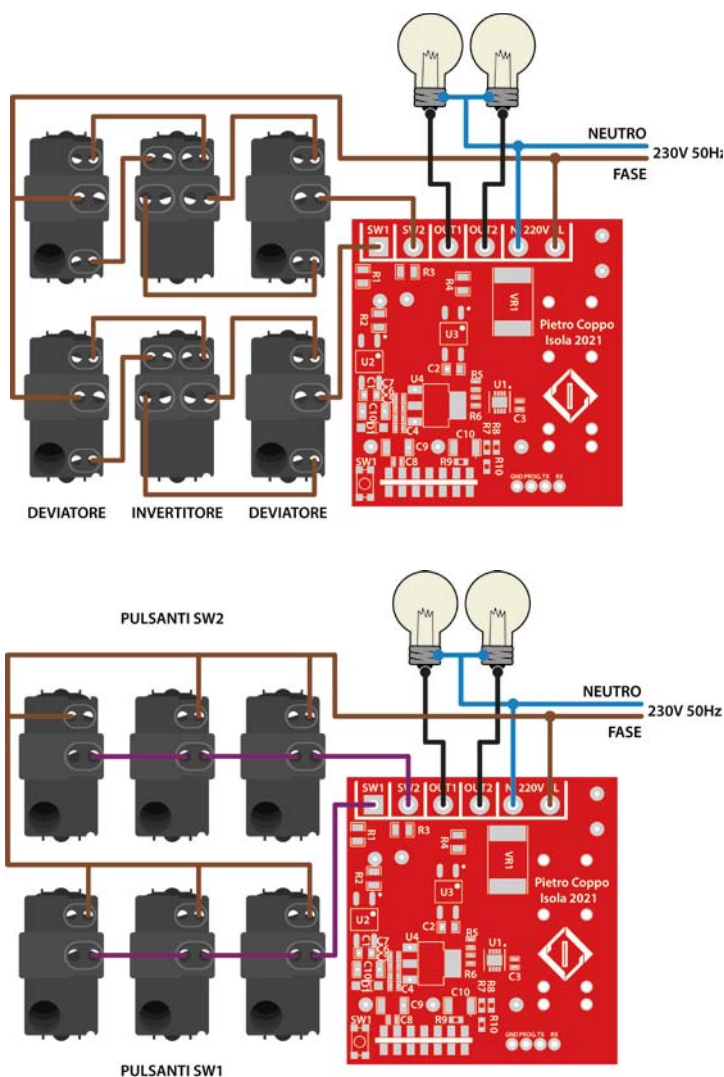
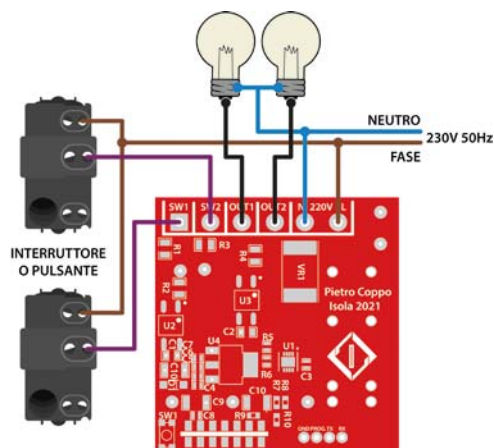


Fig. 9
Utilizzo del Double smart switch con interruttore o pulsante.



de contemporaneamente il ventilatore collegato sull'uscita 2 e fa partire il timer che manterrà in funzione la ventola per il tempo impostato in "Timer – SW2". La modalità 5 invece imposta il dispositivo a funzionare come un comunissimo temporizzatore. Alla pressione del pulsante SW1 o SW2 si accende la luce del corrispettivo canale per il tempo impostato nelle caselle "Timer – SW1" e "Timer – SW2". La casella di spunta "Enable external pushbutton" vi permette di abilitare o disabilitare la regolazione mediante un pulsante esterno.

La casella di spunta "Toggle switch" permette di definire se al Double smart switch collegherete un interruttore oppure un pulsante. 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 della regolazione mediante pagina web.

La casella di spunta "Reboot device on save", abilita il riavvio del dispositivo dopo il salvataggio della configurazione con il tasto "Apply".

In **Fig. 8** e **Fig. 9** trovate alcuni schemi esemplificativi di collegamento del Double smart switch. Di seguito riportiamo alcuni comandi riconosciuti dall'assistente vocale Amazon Alexa e Google Assistance, nell'esempio supponiamo che in <https://sinric.pro> al nostro dispositivo abbiamo assegnato i nomi "Interruttore 1" e "Interruttore 2".

- Per accendere la luce sull'uscita uno pronunciate: "Alexa, accendi Interruttore 1" oppure "Hey Google, accendi Interruttore 1".
- Per accendere la luce sull'uscita due pronunciate: "Alexa, accendi Interruttore 2" oppure "Hey Google, accendi Interruttore 2".
- Per spegnere la luce sull'uscita uno pronunciate: "Alexa, spegni interruttore 1" oppure "Hey Google, spegni Interruttore 1".
- Per spegnere la luce sull'uscita due pronunciate: "Alexa, spegni interruttore 2" oppure "Hey Google, spegni Interruttore 2".

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 il "Double smart switch" sarà ancora più comodo e facile. Per personalizzare l'esperienza e le impostazioni dell'assistente vocale Google Assistant, andate in 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 sulle uscite della versione a relé elettromeccanici potrebbe non spegnersi a causa degli snubber in parallelo ai contatti. Questo non succede nella versione con relé allo stato solido. Per ovviare a questo inconveniente avete tre possibilità: sostituire la lampadina LED con una dimmerabile molto più costosa, eliminare lo snubber, oppure aggiungere un condensatore X2 da 0,22 uF max 0,33 uF in parallelo al carico come mostrato in **Fig. 10**. Con

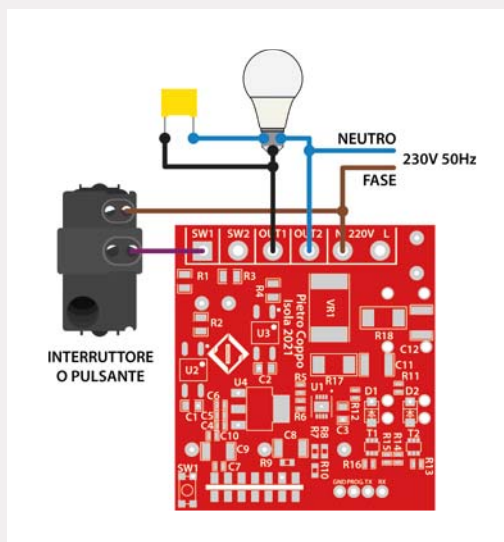


Fig. 10
Cablaggio del condensatore filtro in uscita al Double smart switch EMR in caso di malfunzionamenti con le lampade a LED.

la versione a relé allo stato solido questo difetto non è presente.

CONCLUSIONI

Con i due dispositivi proposti vi rendiamo semplice realizzare da voi un interruttore smart a due canali in due versioni, 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 Double Smart Switch fa parte di una serie di moduli e attuatori domotici progettati e realizzati dagli alunni dell'istituto tecnico Pietro Coppo Isola Slovenia. □



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 (spessore 0,8mm) 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>

Fai il pieno di energia!

€ 12,⁹⁰
Cod. UL45-6



6V-4,5AH

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.

12V-1,3AH



€ 13,⁹⁰
Cod. UL13-12



12V-7AH

€ 24,⁹⁰
Cod. UL7-12

6V-1,3AH



€ 7,⁹⁰
Cod. UL13-6



6V-3,4AH

€ 11,⁹⁰
Cod. UL34-6



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



€ 12,⁰⁰

Cod. HR03AAA800



€ 13,⁰⁰

Cod. HR6AA2100



€ 13,⁸⁰

Cod. HR14C3000

Prezzi IVA inclusa.

RASPBERRY PI PICO

di PIER CALDERAN



Pico è il primo microcontrollore prodotto da Raspberry Pi che monta il chip proprietario RP2040, economico ma estremamente efficiente: scopriamo come utilizzarlo.

A

differenza dei modelli precedenti, Raspberry Pi Pico è una scheda microcontrollore a basso costo e ad alte prestazioni con interfacce digitali I/O, pensata per i principianti. Un po' come la filosofia di Arduino, questo microcontrollore non è equipaggiato come i consueti modelli di Raspberry Pi. Incorpora solo un chip proprietario, siglato RP2040, con un processore Arm Cortex M0+ dual-core che funziona fino a 133 MHz, possiede 264 KB di SRAM e 2 MB di memoria Flash integrata, con 30 pin I/O multifunzione (**Fig. 1**). Grazie al basso costo del chip, oltre al modello Pico commercializzato da Raspberry Pi, esistono già varie versioni prodotte da terze parti che, pur rimanendo compatibili con il chip di base, offrono funzionalità aggiuntive. Ne è un esempio, il modello Arduino Nano RP2040 Connect, dotato di WiFi e Bluetooth.



Fig. 1
Il micro-
controllore
Raspberry Pi
Pico.



Fig. 2
La confezione
e il contenuto
di Raspberry
Pi Pico Starter
Kit.

Oggi però ci concentriamo solo sul capostipite della famiglia e spiegheremo come programmarlo in modo facile, impiegandolo in alcuni semplici progetti.

Per prima cosa diamo uno sguardo alle caratteristiche principali del chip, che sono riepilogate qui di seguito.

- Dual ARM Cortex-M0+ a 133MHz
- 264 kB di SRAM in sei banchi indipendenti
- Supporto per un massimo di 16 MB di memoria Flash off-chip tramite bus QSPI dedicato
- Controllore DMA
- Architettura AHB (AMBA High-performance Bus)
- Periferiche con interpolatore e divisore intero
- LDO (Low Drop Out) programmabile su chip per generare la tensione del core

- 2 PLL (Phase-Locked Loop) su chip per generare il clock USB e core
- 30 pin GPIO, di cui 4 utilizzabili come ingressi analogici
- 2 UART
- 2 controller SPI
- 2 controller I2C
- 16 canali PWM
- Controller USB 1.1 e PHY, con supporto per host e dispositivi
- 8 PIO state machines

Per lo sviluppo del software del chip RP2040, è disponibile l'SDK C/C++ di Raspberry Pi o MicroPython. La programmazione può essere effettuata utilizzando qualsiasi piattaforma, anche se è consigliabile l'uso di una scheda Raspberry Pi.

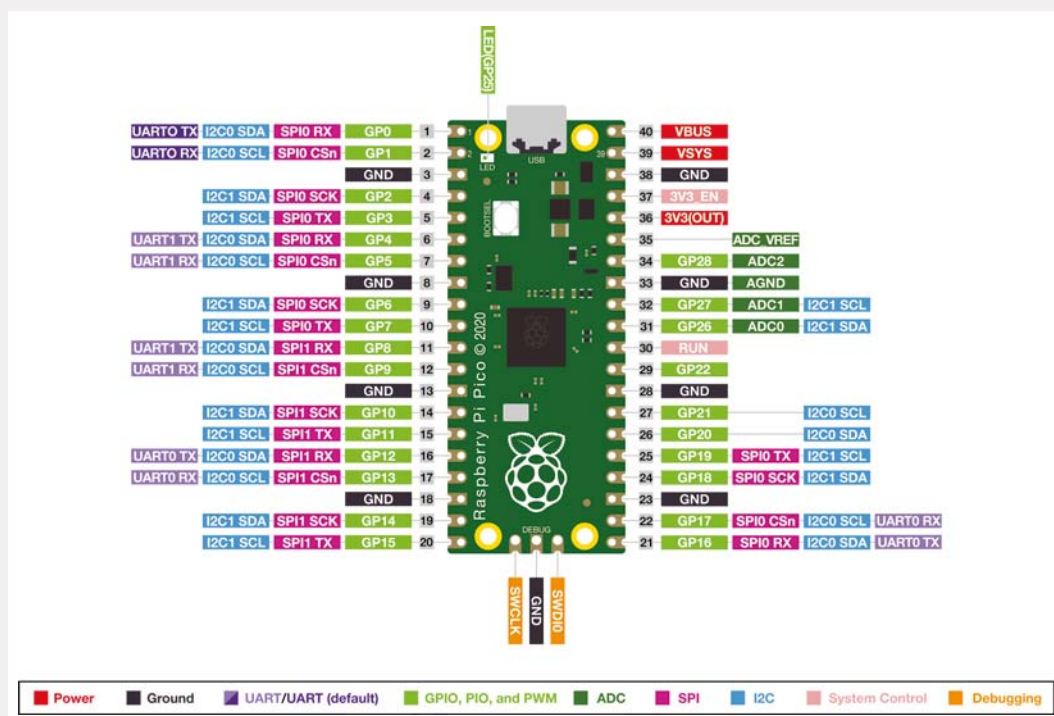


Fig. 3
La piedinatura di
Raspberry Pi Pico.

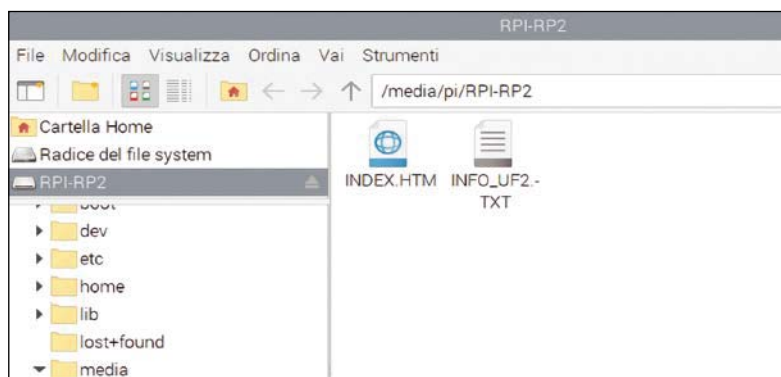


Fig. 4
La cartella con i due file all'interno della cartella dell'unità rimovibile Pico.

RASPBERRY PI PICO STARTER KIT

Per facilitare l'apprendimento ed eseguire subito una serie di esperimenti su Raspberry Pi Pico, è disponibile da Futura Elettronica (cod. RPIKOKIT), uno Starter kit Raspberry Pi Pico (**Fig. 2**).

La confezione in vendita nel nostro store online con il codice 3085-RPIKOKIT comprende una serie di componenti elettronici utili a realizzare alcuni degli esperimenti descritti in questo articolo. Nel kit sono presenti anche i pin header da saldare opzionalmente alla scheda per l'inserimento nella breadboard.

PREPARAZIONE DI RASPBERRY PI PICO

Possiamo collegare via USB il Raspberry Pi Pico (da ora in avanti solo Pico) a un computer qualsiasi. Per comodità abbiamo preferito collegare il nostro piccolo controller a una scheda Raspberry Pi 4 connessa in VNC con un PC.

Questo facilita un po' di cose, soprattutto perché potete lavorare subito con Pico usando Thonny, ovvero l'editor per Python e MicroPython preinstallato nel sistema operativo Raspberry Pi.

La presa micro-USB fornisce l'alimentazione per far funzionare il Pico e consente anche al Pico di parlare con l'ambiente di programmazione di Raspberry Pi. Appena sotto la porta micro-USB c'è un piccolo pulsante contrassegnato con "BOOTSEL" che è l'abbreviazione per "boot selector" o "selettore di avvio". Questo pulsante imposta il Pico in due modalità di avvio all'accensione. Si userà più tardi, non appena avrete scritto il primo programma con MicroPython. Nella parte inferiore del Pico ci sono tre fori più piccoli con sopra la parola "DEBUG". Si tratta di una porta progettata per il debug dei programmi in esecuzione, utilizzando un debugger esterno, di cui non si parlerà in questo articolo.

Capovolgendo il Pico si potrà notare lo strato serigrafico con le etichette di riferimento di tutti i pin, di cui riportiamo lo schema completo delle funzioni (**Fig. 3**).

Questa serigrafia può essere molto utile se, per esempio, volete saldare i pin header con il Pico capovolto. In questo caso, le serigrafie rimarrebbero visibili quando inserite il Pico in una breadboard. Per iniziare la preparazione dell'ambiente operativo, seguite queste semplici operazioni:

1. collegate il Pico alla presa USB tenendo premuto per tre secondi circa il pulsante "BOOTSEL";
2. il Pico verrà montato come un'unità rimovibile;
3. sul desktop di Raspberry Pi apparirà un pop-up che chiederà di aprire l'unità nel File Manager;
4. una volta aperta l'unità nel File Manager apparirà una cartella con due file (**Fig. 4**):

- INDEX.HTM
- INFO_UF2.TXT

Il file INFO_UF2.TXT riporta semplicemente la versione del bootloader e il tipo di scheda, mentre il file INDEX.HTM è un link alla documentazione di Pico. Dobbiamo dire è davvero molto curata e vi consigliamo di leggere le varie sezioni della pagina, al seguente link: <https://www.raspberrypi.com/documentation/microcontrollers>.

INSTALLAZIONE DI MICROPYTHON

Nella pagina della documentazione selezionate la scheda "MicroPython" per aprire la sezione specifica di MicroPython. Se non avete mai usato MicroPython potete visitare il link MicroPython Wiki nella stessa pagina (<https://github.com/micropython/micropython/wiki>).

MicroPython è un'implementazione completa del linguaggio di programmazione Python 3 che viene eseguito direttamente su hardware embedded come Raspberry Pi Pico.

Si ottiene un prompt interattivo REPL (Read Eval Print Loop) ovvero un ambiente shell che permette di eseguire immediatamente i comandi tramite USB Serial e un filesystem integrato. La versione Pico di MicroPython include moduli a basso livello per l'accesso all'hardware specifico del chip.



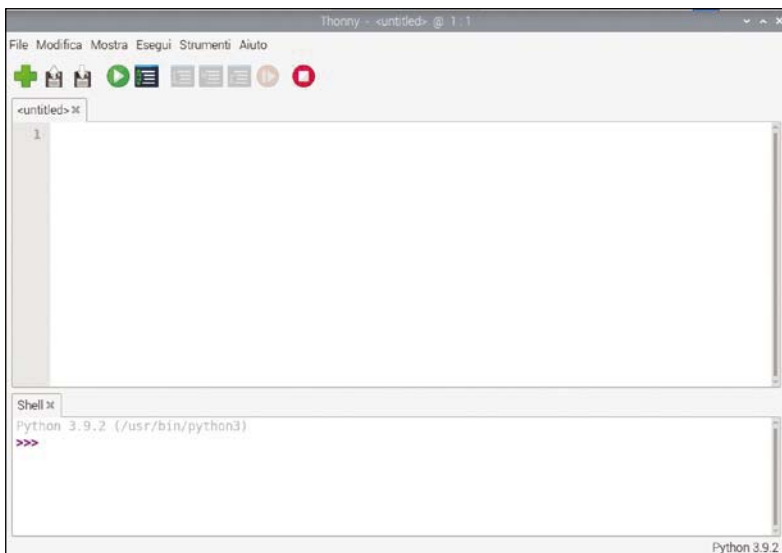


Fig. 5
La finestra
principale di
Thonny.

Nella sezione MicroPython selezionate il link "Download file UF2" per scaricare il firmware MicroPython. Si tratta di un piccolo file che contiene MicroPython per Pico.

Una volta scaricato nella cartella Download, trascinate il file "micropython.uf2" nell'altra finestra aperta (quella dell'unità rimovibile Pico). Una volta rilasciato il file, verrà installato automaticamente MicroPython. Quello che succede in pratica, è il flashing della memoria interna di Pico con il firmware MicroPython. Dopo alcuni secondi la cartella Pico scomparirà dal File Manager.

La pagina web aperta con INDEX.HTM non apre solo le informazioni per scaricare MicroPython, ma include anche una vasta gamma di risorse aggiuntive. Visitate i link per accedere a guide, progetti e la documentazione tecnica dettagliata che copre praticamente tutto, dal funzionamento interno del microcontrollore RP2040 alla programmazione nei linguaggi Python e C/C++.

PROGRAMMAZIONE CON MICROPYTHON

Per semplicità di spiegazione e soprattutto per facilitare l'approccio a Pico ai principianti, oggi parleremo solo della programmazione con MicroPython. Dal suo lancio nel 1991, il linguaggio di programmazione Python è cresciuto fino a diventare uno dei più popolari al mondo. Il linguaggio Python è stato sviluppato per sistemi informatici di qualsiasi tipo e grandezza. Ma i microcontrollori come Pico sono più piccoli, più semplici e con una quantità di memoria notevolmente inferiore, il che significa che non possono eseguire lo stesso codice Python delle loro controparti più grandi.

È qui che entra in gioco MicroPython. Sviluppato originariamente da Damien George e rilasciato per la prima volta nel 2014, MicroPython è un linguaggio di programmazione compatibile con Python, ma sviluppato specificamente per i microcontrollori, da qui il nome. Include molte delle funzionalità di Python, aggiungendone di nuove progettate specificamente per sfruttare le funzionalità disponibili su Pico o altri microcontrollori.

Se avete già programmato con Python, troverete MicroPython immediatamente familiare.

In caso contrario, è un linguaggio da imparare!

THONNY

Prima di iniziare a programmare Pico con MicroPython, dovrete configurare un ambiente di sviluppo integrato (IDE).

Thonny (**Fig. 5**) è un popolare ambiente IDE per Python e MicroPython ed è precaricato nel sistema operativo Raspberry Pi.

Se invece volete utilizzare Pico con una diversa distribuzione Linux, o usare Windows o macOS, potete scaricare dal sito ufficiale "<https://thonny.org>" il programma di installazione del bundle Thonny e Python per il relativo sistema operativo o le istruzioni di installazione per Linux.

Thonny integra tutti gli strumenti necessari per scrivere e sviluppare software in un'unica interfaccia utente. Ovviamente, esistono molti altri ambienti IDE per Python, ma consigliamo vivamente di utilizzare solo Thonny per la programmazione di Pico. Nella parte superiore dell'interfaccia utente di Thonny potete vedere la barra degli strumenti, che offre un sistema di accesso rapido basato su icone e un menu per l'accesso a varie funzionalità del programma. Il menu appare solo in modalità "Regular" ed "Expert", per cui è consigliabile fare clic su "Switch to regular mode" in alto a destra alla prima apertura di Thonny. In questo modo potrete accedere al menu Tools e cambiare altre opzioni come, per esempio, la lingua dell'editor. Fra le tante c'è anche l'italiano. Da qui in avanti i riferimenti e le figure sono relativi all'interfaccia utente in italiano e la modalità dell'interfaccia utente "Regular". La scelta della modalità si trova sotto Strumenti > Opzioni > Generali > Modalità UI.

L'area centrale dell'editor è dedicata agli script, dove vengono praticamente scritti tutti i programmi in linguaggio Python e/o MicroPython.

La parte inferiore è la cosiddetta Python Shell che consente di digitare singole istruzioni che vengono eseguite non appena si preme il tasto INVIO e fornisce anche informazioni sugli script in esecuzione,



Fig. 6
La scelta dell'interprete MicroPython per Pico.

per esempio, le istruzioni *print*, gli errori di esecuzione eccetera.

Thonny viene avviato dal menu Programmazione di Raspberry Pi.

Per impostare Thonny come interprete per Pico, fate clic nella parte inferiore destra della finestra in corrispondenza della scritta con il nome dell'interprete e della versione, per esempio "Python 3.9.2". Nell'elenco che appare, selezionate "MicroPython (Raspberry Pi Pico)" (**Fig. 6**). Questa opzione è presente anche in Strumenti > Opzioni > Interprete. Fatto questo, si aprirà una finestra (**Fig. 7**) dalla quale installare o aggiornare MicroPython nel Pico.

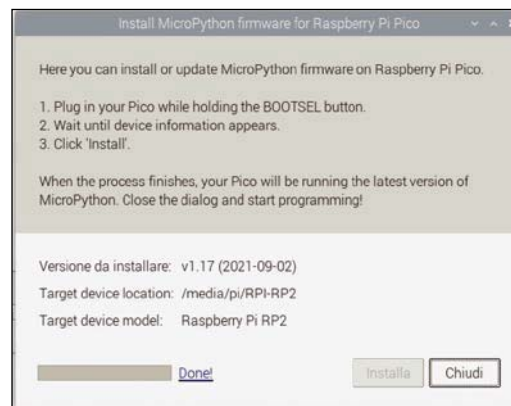


Fig. 7
L'installazione dell'interprete MicroPython per Pico.

IL PRIMO PROGETTO CON PICO

Raspberry Pi Pico, con il suo piccolo microcontrollore RP2040, è stato progettato pensando al calcolo fisico. I suoi numerosi pin di input/output (GPIO) per uso generico consentono di comunicare con una vasta gamma di componenti, consentendo di creare progetti anche molto complessi. Non vogliamo soffermarci sui vari *print* ("Hello World!") e altre istruzioni di Python che già dovreste conoscere. Concentriamoci piuttosto sul primo progetto "fisico" con Pico.

Attenzione: vi raccomandiamo di tenere sempre d'occhio la serigrafia o la figura con la piedinatura. A tal proposito vi ricordiamo che nel nostro Starter Kit ci sono due cartoncini che riportano la piedinatura completa a grandezza naturale.

BLINK DEL LED INTERNO

Per capire come funziona il tutto, vi consigliamo di scrivere nell'editor di Thonny queste poche righe che servono a far lampeggiare a intervalli di un secondo il LED interno della scheda Pico (**Fig. 8**):

```
import machine
import utime
led_onboard = machine.Pin (25, machine.Pin.OUT)
while True:
    led_onboard.value (1)
    utime.sleep (1)
    led_onboard.value (0)
    utime.sleep (1)
```

Come potete vedere, la prima riga è:

```
import machine
```

Questa riga di codice è la chiave per lavorare con MicroPython su Pico. La libreria "machine" contiene tutte le istruzioni necessarie a MicroPython per comunicare con Pico e/o altri dispositivi compatibili, estendendo il linguaggio per il calcolo fisico. Senza questa linea, non sarete in grado di controllare nessuno dei pin GPIO di Pico e, nel nostro caso, non potremo accendere il LED interno. La riga successiva del programma fornisce un esempio dell'API della libreria machine:

```
led_onboard = machine.Pin (25, machine.Pin.OUT)
```

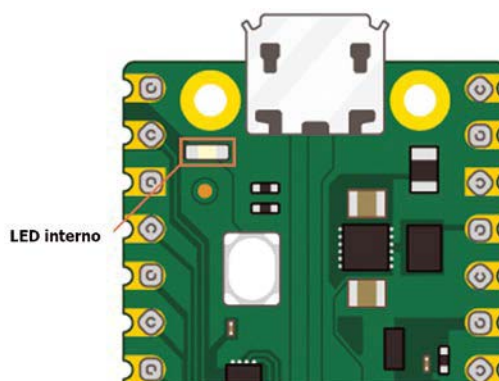


Fig. 8
Il LED interno di Pico.

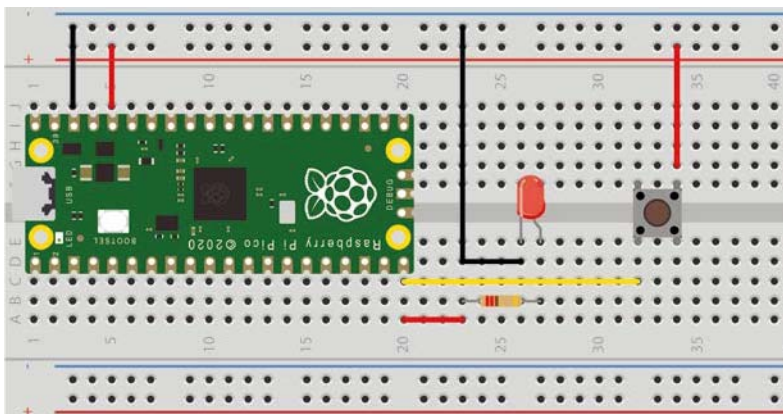


Fig. 9
Collegamento
di un LED e di
un pulsante.

Questa riga definisce un oggetto chiamato `led_onboard`, che è un nome descrittivo per fare riferimento al LED interno. È ovviamente possibile utilizzare qualsiasi nome, ma è meglio attenersi a nomi che descrivono lo scopo della variabile, per rendere il programma più leggibile e comprensibile. L'istruzione `machine.Pin(25, machine.Pin.OUT)` chiama la funzione della libreria per gestire i pin GPIO di Pico:

- il primo argomento, 25, è il numero del pin che si sta configurando;
- il secondo, `machine.Pin.OUT`, dice a Pico che il pin deve essere usato come output.

All'interno del ciclo `while`, l'istruzione `led_onboard.value(1)` accenderà il LED interno (denominato appunto `led_onboard`) mentre l'istruzione `led_onboard.value(0)` lo spegnerà. L'istruzione `utime.sleep(1)` imposta il ritardo a 1 secondo fra i due stati. Attenzione che quando si parla dei pin GPIO sul Pico, di solito vengono indicati con i loro nomi completi: per esempio, GP25 sta per il pin collegato al LED interno. In MicroPython, le lettere GP devono essere eliminate, quindi bisogna scrivere "25" anziché "GP25" nel programma, altrimenti non funzionerà.

LIBRERIA UTIME INVECE DI TIME

Invece di importare la classica libreria `time` di Python

per creare temporizzazioni e ritardi, qui viene usata la libreria `utime` (u sta per la lettera μ = micro). Si fa notare che la libreria `utime` è una versione progettata per microcontrollori come Pico e anche se si importa la libreria `time`, MicroPython utilizzerà automaticamente la libreria `utime`.

ESECUZIONE DEL PROGRAMMA

Per eseguire il programma basta fare clic sul pulsante Esegui o F5 da tastiera o dal menu Esegui. Vi verrà chiesto di salvare il programma sul Pico. Salvate il file con il nome "Blink.py" per esempio. Vedrete il LED lampeggiare finché non si preme l'icona Stop per annullare il programma e ripristinare Pico.

BLINK CON TOGGLE

C'è anche un altro modo per gestire il lampeggio del LED. Se si usa la funzione `toggle` della libreria `machine` potete ottenere lo stesso risultato con le seguenti righe di codice:

```
import machine
import utime
led_onboard = machine.Pin(25, machine.Pin.OUT)
while True:
    led_onboard.toggle()
    utime.sleep(1)
```

La funzione, `toggle()` è disponibile su tutti i pin di uscita digitale e commuta semplicemente tra on e off: se il pin è alto, `toggle()` lo disattiva; se è basso, `toggle()` lo attiva. Notare che gli stessi programmi descritti qui sopra funzionano anche con LED esterni. Basta solo collegare nella breadboard un LED a un pin GPIO qualsiasi. In questo caso, ricordatevi di mettere una resistenza da 330 Ω in serie al LED.

Per esempio, potete modificare l'istruzione precedente con la seguente per controllare un LED collegato a GP15.

Listato 1

```
import machine
import utime
led_external = machine.Pin(15, machine.Pin.OUT)
button = machine.Pin(14, machine.Pin.IN, machine.Pin.PULL_DOWN)
while True:
    if button.value() == 1:
        led_external.value(1)
    if button.value() == 0:
        led_external.value(0)
```



```
led_external = machine.Pin (15, machine.Pin.OUT)
```

USARE INSIEME INGRESSI E USCITE

Una volta capito come impostare i pin GPIO in uscita potete assemblare un semplice circuito che accende e spegne un LED con la pressione di un pulsante. Facendo riferimento alla **Fig. 9**, eseguite questi collegamenti:

1. collegate nella breadboard l'anodo del LED al pin GP15 con una resistenza da 330 Ω in serie.
2. collegate nella breadboard il catodo del LED al negativo della breadboard.
3. collegate un terminale del pulsante al pin GP14.
4. collegate l'altro terminale del pulsante al positivo della breadboard.

Il codice (le istruzioni) per svolgere il compito dell'esempio è composto dalle righe proposte nel

Listato 1.

Come potete vedere, con l'istruzione *machine.Pin(15, machine.Pin.OUT)* si imposta il pin GP15 come uscita, mentre con l'istruzione *machine.Pin(14, machine.Pin.IN, machine.Pin.PULL_DOWN)* si imposta il pin GP14 come ingresso in PULL_DOWN, ovvero sempre basso. Una volta avviato il programma, tenendo premuto il pulsante, il LED si accenderà e, viceversa, si spegnerà.

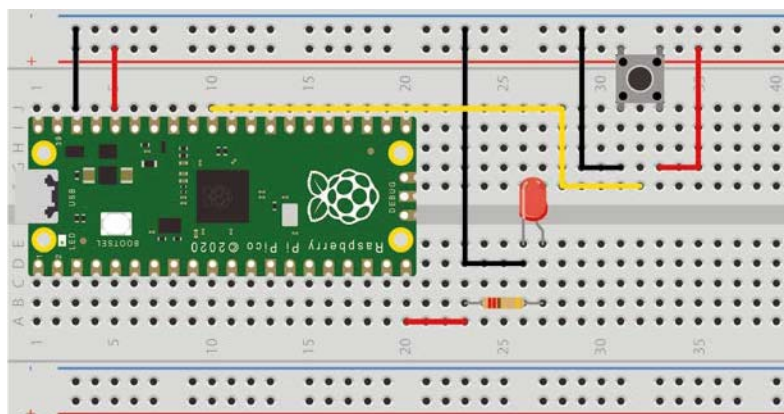
DISSOLVENZA DI UN LED CON PWM

Il convertitore analogico-digitale (ADC, Analog Digital Converter) di Pico prende un segnale analogico e lo converte in un segnale digitale così che il microcontrollore lo possa elaborare. Al contrario, se il microcontrollore digitale dovesse creare un'uscita analogica dovrebbe essere dotato di un convertitore digitale-analogico (DAC, Digital Analog Converter). Dal momento che il Pico non possiede un DAC ma solo uscite digitali, possiamo usare la tecnologia PWM (Pulse Width Modulation o modulazione di larghezza di impulso) per "simulare" un segnale analogico. In questo modo, potrete collegare un potenziometro a un ingresso ADC e un LED a un'uscita PWM. Lo schema di collegamento è illustrato nella **Fig. 10**. Il codice MicroPython è davvero molto semplice:

```
import machine
import utime

potentiometer = machine.ADC(26)
led = machine.PWM(machine.Pin(15))
led.freq(1000)

while True:
    reading = potentiometer.read_u16()
```



```
print(reading)
led.duty_u16(reading)
utime.sleep(.5)
```

Fig. 10
Lo schema di collegamento di un LED e di un potenziometro.

Come potete vedere, con l'istruzione *machine.PWM(machine.Pin(15))* il LED viene attivato con la PWM sull'uscita GP15, mentre l'ingresso del potenziometro viene impostato dall'istruzione *machine.ADC(26)* che abilita uno dei quattro ingressi analogici (GP28, GP25, GP26 e GP35).

Una volta avviato il programma, ruotando il potenziometro vedrete cambiare la luminosità del LED. Notare che la risoluzione di lettura dell'ADC è a 16 bit per cui si vedranno stampati nella shell i valori da 0 a 65535.

SENSORE DI TEMPERATURA

Il microcontrollore RP2040 del Pico ha un sensore di temperatura interno, che è in grado di leggere sul quarto canale del convertitore analogico-digitale. Come il potenziometro, l'uscita del sensore è un segnale a tensione variabile per cui al variare della temperatura, cambia anche la tensione.

Il codice da usare è quello contenuto nel **Listato 2**. Notate che è necessario impostare un fattore di conversione che corrisponde al quoziente fra la tensione massima (3.3 volt) e il valore in bit (65535).

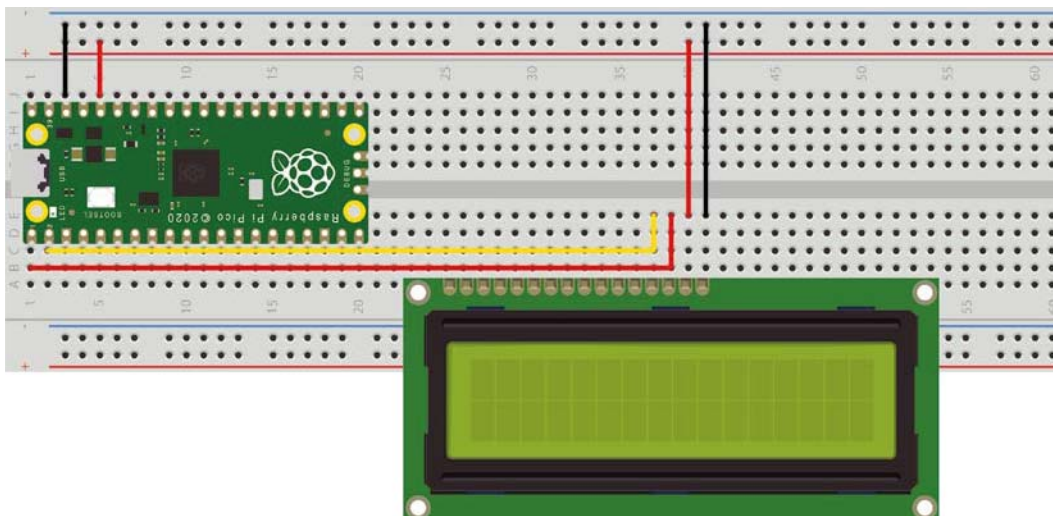
Listato 2

```
import machine
import utime

sensor_temp = machine.ADC(4)
conv_fac = 3.3 / (65535)

while True:
    reading = sensor_temp.read_u16() * conv_fac
    temperature = 27 - (reading - 0.706) / 0.001721
    print(temperature)
    utime.sleep(2)
```

Fig. 11
Il display LCD a
una delle porte I²C
della scheda Pico.



Prima di ottenere la lettura della temperatura, è necessario eseguire una seconda conversione, per prendere la tensione riportata dal convertitore analogico-digitale e convertirla in gradi Celsius:

$$\text{temperatura} = 27 - (\text{lettura} - 0,706)/0,001721$$

Questa è un'operazione matematica specifica per il sensore di temperatura del chip RP2040. I valori sono presi da un documento tecnico disponibile al seguente link: <https://datasheets.raspberrypi.com>.

DISPLAY LCD I²C

Nello Starter Kit di Futura Elettronica potete trovare anche un display LCD compatibile con l'interfaccia I²C-Bus della scheda Pico.

Fra le varie interfacce disponibili, questo display consente un collegamento I²C-Bus con il Pico davvero molto semplice.

Collegando il display a una delle porte I²C di Pico e l'alimentazione (**Fig. 11**), potete aggiungere poche righe allo script precedente per vedere l'output della temperatura sul display: le vedete nel **Listato 3**. L'istruzione `machine.I2C(0,sda=sda, scl=scl, freq=400000)` imposta i pin GP0 e GP1 come interfaccia I²C-Bus. Nel ciclo while sono state aggiunte le istruzioni `i2c.writeto()` per la scrittura nei registri del display e della stringa di output. Una volta avviato il programma, potrete vedere la temperatura stampata sul display.

CONCLUSIONI

Ci sarebbe ancora molto da dire ma per oggi ci fermiamo qui. Con la speranza di essere stati sufficientemente chiari, vi invitiamo a sperimentare i progetti presentati in queste pagine, sfruttando il materiale del nostro Starter Kit. Per vostra comodità, tutto il codice degli esempi è disponibile nella nostra area download. □

▼ Listato 3

```
import machine
import utime
sda = machine.Pin(0)
scl = machine.Pin(1)
i2c = machine.I2C(0,sda=sda, scl=scl, freq=400000)
adc = machine.ADC(4)
conv_fac = 3.3 / (65535)
while True:
    reading = adc.read_u16() * conv_fac
    temperature = 25 - (reading - 0.706)/0.001721
    i2c.writeto(114, '\x7C')
    i2c.writeto(114, '\x2D')
    out_string = "Temp: " + str(temperature)
    i2c.writeto(114, out_string)
    utime.sleep(2)
```



Cosa occorre?

I componenti utilizzati in questo progetto sono disponibili presso Futura Elettronica. In particolare lo starter kit per Raspberry Pi Pico (cod. RPICOKIT) è in vendita a Euro 49,90. Raspberry Pi Pico è disponibile anche separatamente (cod. RPI-PICO) a Euro 6,90. 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>

OSCILLOSCOPIO 2 CANALI 70 MHZ

- ✓ Frequenza di campionamento fino a 1 GS/s
- ✓ Funzione FFT integrata
- ✓ Software per analisi in tempo reale con il PC
- ✓ Porta USB Host

cod. DS072

€ 329,00

Disponibile anche
nella versione a 200MHz

cod. DS0202

€ 459,00



OSCILLOSCOPIO 2 CANALI 100 MHZ CON GENERATORE DI FORME D'ONDA



cod. DS0FG102

€ 469,00

- ✓ Frequenza di campionamento fino a 1 GS/s
- ✓ Funzione FFT integrata
- ✓ Software per analisi e controllo in tempo reale con il PC
- ✓ Porta USB Host / USB Device

Disponibile anche
nella versione a 200MHz

cod. DS0FG202

€ 549,00

GENERATORE DI FUNZIONI E FORME D'ONDA ARBITRARIE 2 CANALI 30 MHZ

- ✓ Frequenza onda DC 30 MHz
- ✓ 2 canali
- ✓ Frequenza di campionamento 250 MSa/s
- ✓ Risoluzione verticale 16 Bit

cod. HDG32B

€ 439,00



Disponibile anche
nella versione a 60MHz

cod. HDG62B

€ 510,00

OSCILLOSCOPIO PORTATILE 2 CANALI - 40 MHZ CON MULTIMETRO

Dispone di ampio display LCD TFT a colori da 2,8. La larghezza di banda è di 40 MHz, 250 MSa/s di velocità di campionamento in tempo reale a canale singolo. Il multimetro fornisce misurazioni di tensione continua e alternata, corrente continua e alternata, resistenze, capacità, test diodi e continuità. È interfacciabile a PC tramite software scaricabile gratuitamente. Supporta Windows 7 / 8 / 10 a 32 e 64 bit.



cod. MH02C42

€ 178,00

Disponibile anche nella versione
a 2 canali 70 MHz con generatore
di forme d'onda e multimetro

cod. MH02D72GEN

€ 268,00

Costruisci un vero robot!

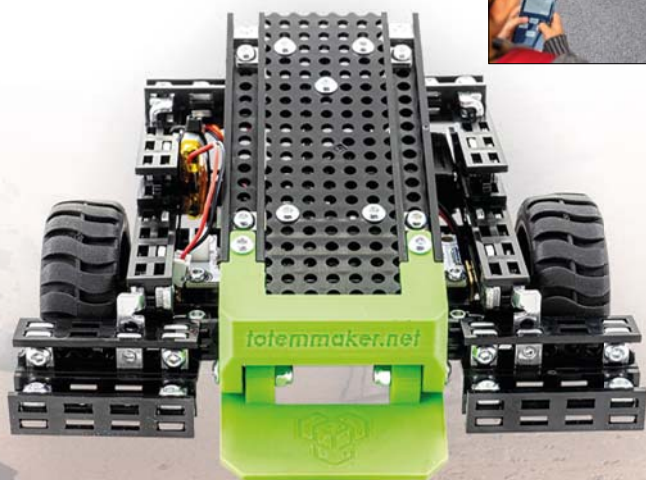


Mini robot da combattimento controllato da App

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®**

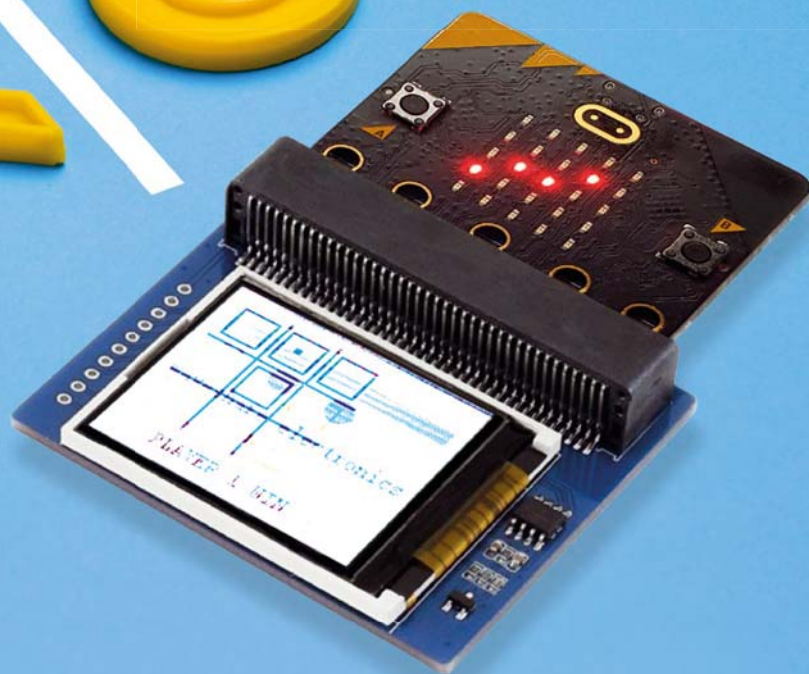
Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche
e vendita on-line su:

www.futurashop.it

di MARCONATALE PARISE
e dell'Ing. VINCENZO
GERMANO

GIOCO A TRIS CON micro:bit



Riedizione di un classico gioco elettronico realizzato in chiave moderna con la piccola scheda che rende facile imparare a programmare.

P

er quanto oggi siamo letteralmente "bombardati" da ogni tipo di videogame, su console, PC e smartphone, esistono giochi comunque suggestivi anche se semplici; ecco perché abbiamo deciso di proporre in chiave moderna uno dei passatempi più famosi al mondo: il Tris.

Un popolarissimo gioco carta e matita, o gessetto e lavagna o ancora meglio giocabile con un bastoncino sulla sabbia, fatto di poche e semplici regole, ma caratterizzato da tanti aspetti curiosi che vogliamo esporvi prima di passare al progetto. Il gioco del Tris è anche noto in Italia come crocetta e pallino, filetto, cerchi e croci, oppure con simboli del tipo OXO o XOXO che riprendono direttamente i simboli utilizzati nel gioco; considerando che è universale, in altre parti del mondo è noto con altri nomi, per esempio negli Stati Uniti si chiama tic-tac-toe (oppure tick-tat-toe e tit-tat-toe); in Inghilterra è "noughts



Fig. 1
La scheda
micro:bit con
applicato il
display LCD.

and crosses" (zero e croci); in Francia si chiama Morpion (piattola); in Spagna ta-te-ti, tres en raya, gato, equis cero. Alcune ricerche sostengono che giochi simili fossero già diffusi nell'antico Egitto e al tempo dei Romani in cui esisteva un gioco chiamato "terni lapilli" - utilizzato tra i soldati - in cui però, non venivano usati gli stessi simboli di oggi (X e O). Il Tris deve sicuramente la sua notorietà e longevità negli anni alla semplicità delle sue regole, infatti nella sua accezione più generale si gioca su una griglia quadrata di 3x3 caselle e all'inizio del gioco un giocatore sceglie come simbolo una "X",

mentre l'avversario un "O" (cerchio); dopodiché a turno, i giocatori scelgono una casella vuota e vi disegnano il proprio simbolo. Vince il giocatore che riesce a disporre tre dei propri simboli in linea retta orizzontale, verticale od obliqua; per quanto, molto spesso le partite terminano con un pareggio (soprattutto se i giocatori non sono giovanissimi), infatti se la griglia viene riempita senza che nessuno dei giocatori sia riuscito a completare una linea retta di tre simboli, il gioco finisce in parità e come accade in altri giochi, tanto per citarne uno quello degli scacchi, nel caso in cui il gioco finisse in parità, la partita è detta "patta".

Avendo un numero ridotto di situazioni in cui ci si può trovare a giocare e una griglia 3x3, è possibile implementarlo in un programma per microprocessore o microcontrollore, come abbiamo fatto.

Vedremo a breve l'implementazione su Microbit che tuttavia non sarà fatta con un giocatore "umano" contro Microbit ma per semplificare ulteriormente l'implementazione ed evitare di confrontarsi con un dispositivo che non perderà mai una partita, abbiamo pensato di realizzare il gioco del Tris con una doppia partecipazione umana, ovvero come in origine con carta e matita.

IL NOSTRO HARDWARE

Prima di passare all'implementazione software, vediamo il necessario hardware per la corretta gestione di questo progetto, dove utilizziamo micro:bit e un display LCD da 1.8 pollici della

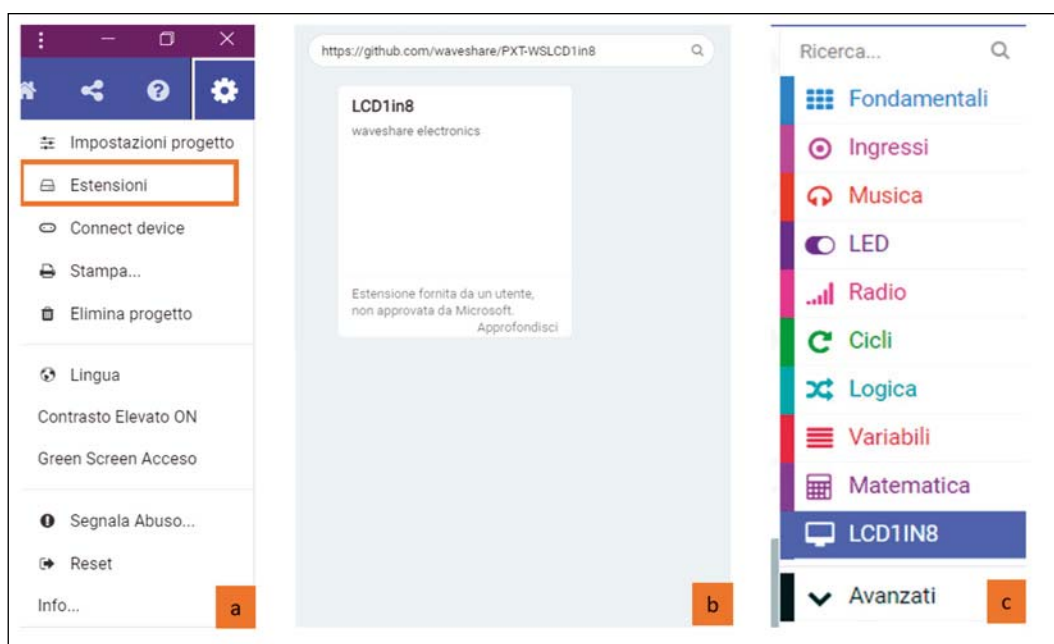


Fig. 2
Aggiunta
Librerie.

Waveshare Electronics mostrato in **Fig. 1**. Questo display è facilmente assemblabile perché ha solo ed esclusivamente il connettore d'interfaccia da collegare al micro:bit e null'altro, in più offre un'ampio display se confrontato con altre soluzioni simili e il driver per la sua gestione è direttamente su scheda. Sulla scheda del display LCD è incorporato il controller a interfaccia SPI (Serial Peripheral Interface): si tratta di un chip Sitronix ST7735S che permette di pilotare 162 e 132 canali RGB al fine di abilitare o disabilitare qualsiasi pixel del display LCD. Tramite il protocollo di comunicazione SPI o con un'interfaccia parallela da 8bit a 18bit è possibile comunicare con un microprocessore esterno e salvare i dati nella memoria RAM disponibile di 132x168x18 bit. Il controller consente anche di minimizzare i consumi energetici realizzando operazioni di scrittura/lettura senza alcun clock (segnale di temporizzazione) esterno, ma per maggiori dettagli si faccia riferimento al sul datasheet scaricabile on-line.

Per quanto l'architettura del sistema sembri semplice, le tecnologie di micro:bit e del display LCD mascherano la complessità astruendo mediante varie librerie software la complessità dell'hardware

DESCRIZIONE LIBRERIE

A questo punto possiamo direttamente all'implementazione software del sistema. Tra i software utilizzabili, abbiamo deciso di utilizzare la piattaforma Make Code Block e di importare le librerie messe a disposizione dal produttore del display, per avere tutte le sue funzionalità immediatamente utilizzabili.

Procedendo per ordine, vediamo come aggiungere tali librerie al programma, con riferimento alla **Fig. 2**: innanzitutto dall'interfaccia utente bisogna cliccare sul pulsante a forma d'ingranaggio e poi, dal menu cui dà accesso, cliccare su "Estensioni" (a); nella finestra cui accederete digitate il path <https://github.com/waveshare/PXT-WSLCD1in8> (b) ed infine aggiungete la libreria *LCD1in8* dedicata a micro:bit v2. Fatto ciò, la nuova libreria LCD1IN8 comparirà nel pannello principale (c).

Prima di procedere oltre è opportuno conoscere le funzioni della libreria che utilizzeremo, per interagire correttamente con il display e che è possibile visionare dalla piattaforma di sviluppo tramite un click sul nome della libreria LCD1IN8. Vediamo quali sono le varie funzioni e i parametri da definire:

- **LCD1IN8 Init**: consente di inizializzare il display;
- **Clear Drawing cache**: cancella tutti i dati in RAM;

- **Filling Color**: consente di definire il colore di background del display;
- **LCD Clear**: cancella i dati in RAM mostrati su display;
- **Show Full Screen**: aggiorna il display con i dati disponibili in memoria RAM;
- **Set back light level**: definisce il livello di intensità luminosa della retroilluminazione;
- **Draw Point**: consente di scrivere in memoria RAM le informazioni relative ad un "punto" del quale devono essere definiti i seguenti parametri: coordinata x, y, colore, dimensioni del punto tramite un menù a tendina;
- **Draw Line**: consente di scrivere in memoria RAM le informazioni relative ad una linea attraverso la definizione di due punti: coordinata 'x' iniziale, 'y' iniziale, 'x' finale, 'y' finale, (si aggiungono anche: colore, dimensioni e spessore della linea tramite un menù a tendina);
- **Draw Rectangle**: consente di scrivere in memoria RAM le informazioni relative ad un rettangolo del quale devono essere forniti i vertici estremi in termini di coordinate x e y: coordinata 'x' iniziale, 'y' iniziale, 'x' finale, 'y' finale, (altre informazioni che è necessario definire sono: colore, il tipo di riempimento del rettangolo, spessore della linea tramite un menù a tendina);
- **Draw Circle**: consente di scrivere in memoria RAM le informazioni relative ad un cerchio del quale devono essere forniti il centro in termine di coordinate x, y ed il raggio: coordinata 'x', 'y', raggio, (in più: colore, il tipo di riempimento del cerchio, spessore della linea tramite un menù a tendina);
- **Show Number**: consente di scrivere in memoria RAM le informazioni relative ad un numero del quale deve essere fornita la posizione in termine di coordinate x, y ed il colore: coordinata 'x', 'y', colore;
- **Show String**: consente di scrivere in memoria RAM le informazioni relative ad un carattere del quale deve essere fornita la posizione in termine di coordinate x, y ed il colore: coordinata 'x', 'y', colore.

LE LOGICHE

Il progetto, per quanto semplificato a livello hardware richiede un'importante implementazione dal punto di vista software, per cui risulta fondamentale descrivere le logiche di implementazione; per facilitarne la comprensione è buona norma realizzare un flow-chart (diagramma di flusso) che nel caso specifico è mostrato in **Fig. 3**.



Fig. 3
Diagramma di
flusso Gioco Tris.

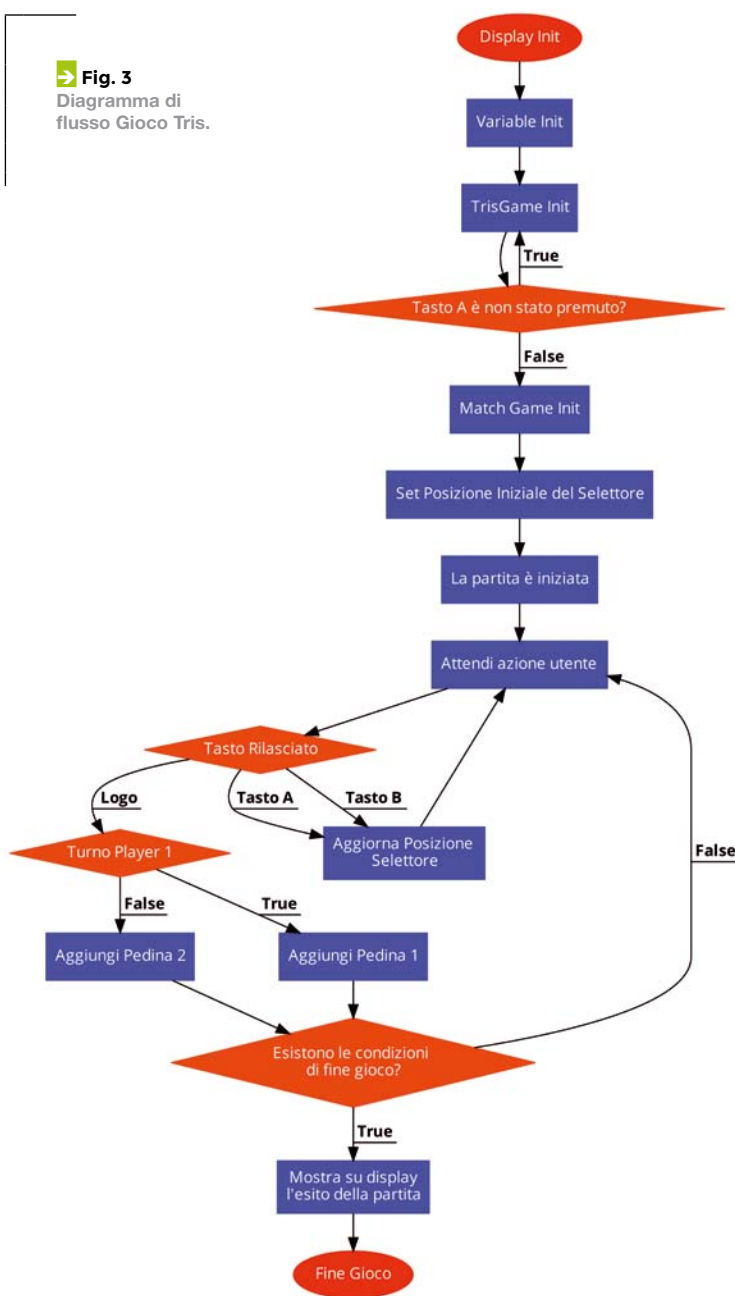


Fig. 4
Inizializzazione
del gioco
Tris.

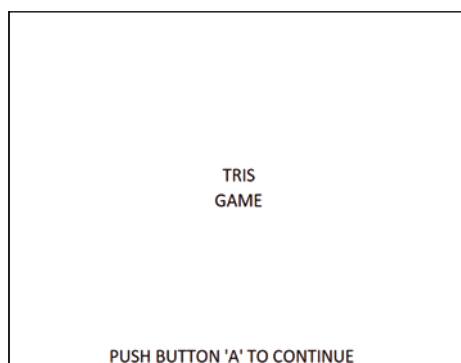
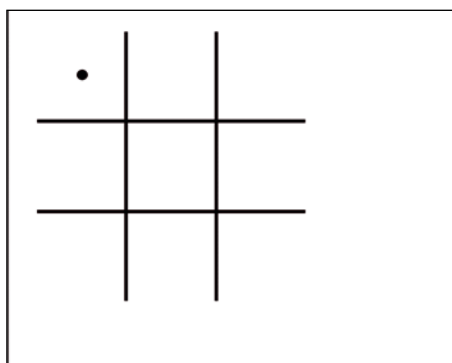
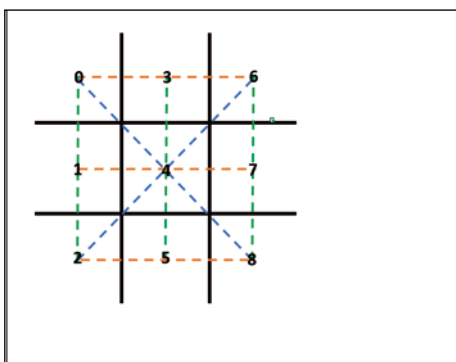


Fig. 5
Inizializ-
zazione
della
partita.

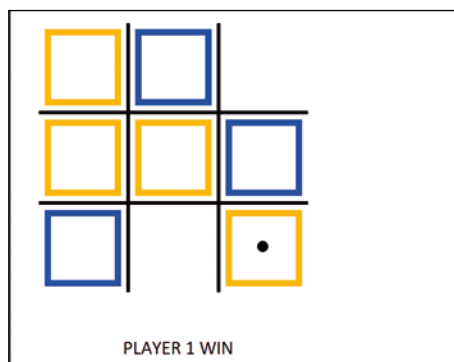


La prima fase del programma consiste nell'inizializzazione del display e delle variabili che verranno descritte nel successivo capitolo così da migliorare la lettura del programma. Il successivo blocco di "TrisGame Init" riguarda le prime istruzioni che guideranno l'utente nella gestione della partita e a tale scopo utilizziamo come riferimento la **Fig. 4**, che esemplifica ciò che verrà mostrato dal display. Il display è composto da 160x128 pixel pertanto qualsiasi dato può essere interpretato come la successione di 'n' pixel con il fine di mostrare una parola oppure una figura. Quindi la prima fase del gioco consiste nel mostrare la stringa "Tris GAME" e nel guidare l'utente nella successiva azione, ovvero la pressione del "tasto A" presente su Microbit. Alla pressione di tale tasto seguono i vari blocchi di inizializzazione partita che consistono nella definizione e scrittura su display della griglia del Tris, del selettore e della prima pedina. La griglia sarà composta da nove riquadri (come vedevamo prima una griglia 3x3) all'interno dei quali il selettore (identificato da un pallino nero) potrà navigare per definire la successiva posizione della pedina. Con la pressione del tasto "A" si avrà quanto riportato in **Fig. 5**. A questo punto il programma attende la successiva azione utente che consiste nello spostare il selettore e definire la successiva posizione. La micro:bit rende disponibili due tasti che utilizzeremo per la movimentazione del selettore e il logo touch per confermare l'inserimento della pedina. Il tasto A verrà utilizzato per il movimento verticale mentre il tasto B per il movimento orizzontale che saranno di tipologia endless (senza fine) così da garantire la completa movimentazione all'interno della griglia. Come visto nell'introduzione, il gioco del Tris si basa sullo scambio di azioni utente che posizionano le loro pedine al fine di realizzare il Tris ovvero tre pedine consecutive in orizzontale, verticale o diagonale. Pertanto il turno è inizializzato dal player 1 che sposta il selettore tramite i tasti A e

→ **Fig. 6**
Possibili
combinazioni
di vittoria
partita.



← **Fig. 7**
Fine
Partita.



B nella posizione desiderata e conferma l'inserimento della pedina tramite il logo touch. Terminata questa operazione, il programma deve controllare se esistono le condizioni di vittoria altrimenti passa il turno al secondo player che potrà inserire la sua pedina e così via. Le condizioni di vittoria sono identificate nella **Fig. 6** e vengono verificate ogni qual volta una nuova pedina è inserita. I numeri associati ad ogni casella vengono utilizzati per rendere più intuitiva e semplice la lettura delle combinazioni e torneranno molto utili in fase di implementazione software. Le linee tratteggiate indicano la sequenzialità delle caselle e quindi di una condizione di vittoria che può essere interpretata come combinazione di numeri, ad esempio la linea arancio orizzontale rappresenta la sequenza numerica 0-3-6. Quando le condizioni di vittoria sono verificate, con una delle 8 possibili soluzioni vittoria, la partita può ritenersi conclusa e sul display verrà mostrato il testo con il vincitore (**Fig. 7**) altrimenti la partita terminerà con un pareggio.

IMPLEMENTAZIONE SOFTWARE

Passiamo alla scrittura del codice tramite la piattaforma di sviluppo Make Code Block analizzando passo-passo le funzioni realizzate in linea con il diagramma di flusso mostrato precedentemente in **Fig. 3**. Le prime funzioni realizzate sono eseguite all'interno della funzione avvio e riguardano i blocchi di inizializzazione display (Display Init), variabili (Variables Init) e gioco del Tris (TrisGame Init). Per comprendere al meglio le funzioni implementate, utilizziamo come riferimento la **Fig. 8** che mostra alcuni parametri molto importanti per la comprensione delle logiche che andremo a descrivere. La creazione della griglia e della pedina si basa sulla definizione delle coordinate x, y che verranno utilizzate come parametri delle funzioni "display":

- x_p, y_p – indicano le coordinate del pixel dal quale verrà creata la griglia;

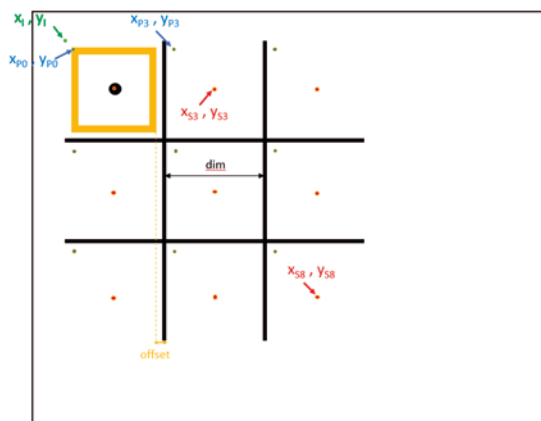
- x_{px}, y_{py} – indicano le coordinate del pixel dal quale verrà creata la pedina;
- x_{sx}, y_{sy} – indicano le coordinate del selettore;
- dim – rappresenta la dimensione in termini di pixel di ogni quadrante della griglia;
- $offset$ – indica il numero di pixel che distanziano la pedina dalla griglia;

Per facilitare la comprensione del testo utilizzeremo il carattere corsivo per le variabili e il grassetto corsivo per le funzioni.

INIZIALIZZAZIONE PROGETTO

Riportiamo in **Fig. 9** la lista di variabili che andremo ad utilizzare successivamente e le funzioni adottate:

- Variabili *black, blue, white, red e green* verranno utilizzate per definire il colore delle pedine e del selettore;
- Variabili *_dim* e *offset_pedina* le abbiamo già descritte;
- Variabili *S_base_x* e *S_base_y* sono le coordinate del primo pixel dal quale verrà creata la griglia del Tris;



← **Fig. 8**
I parametri
relativi alle
logiche.

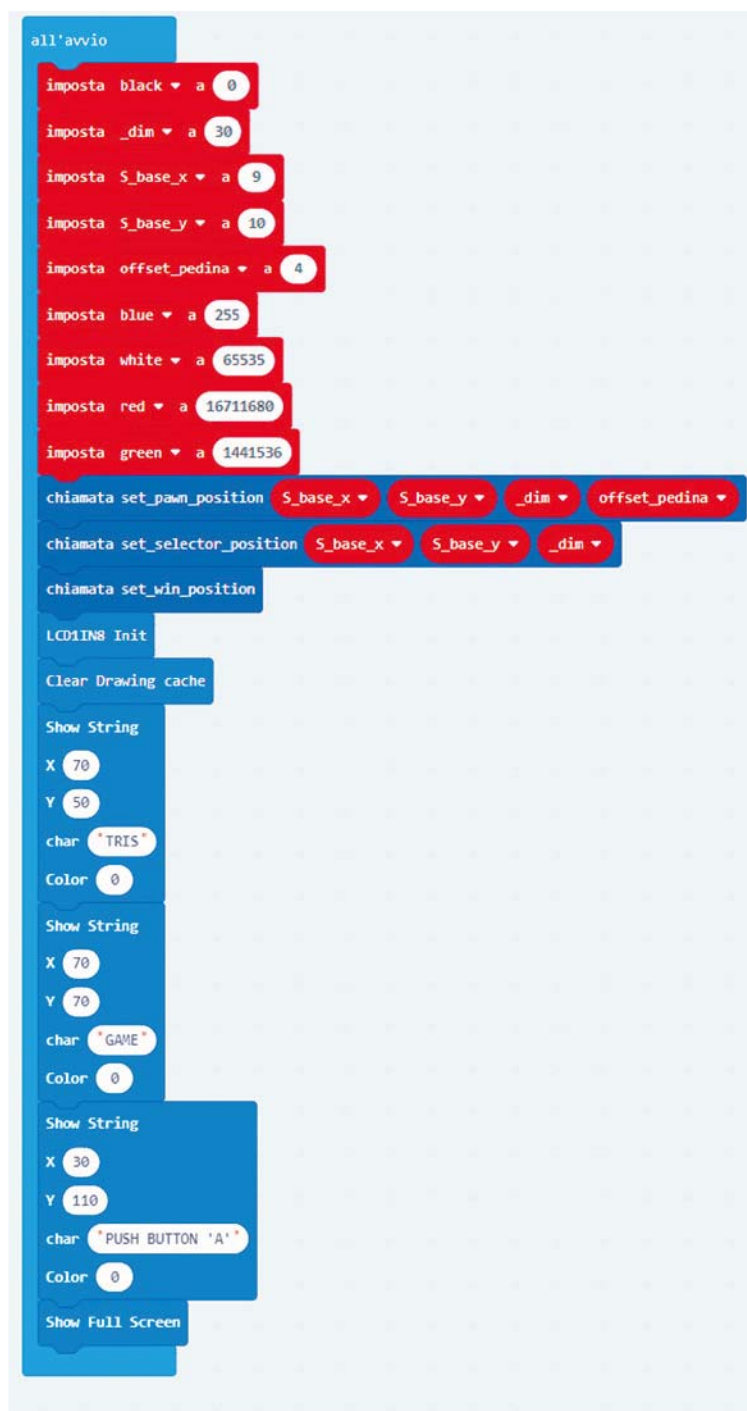


Fig. 9
Inizializzazione del gioco.

Le funzioni **set_pawn_position**, **set_selector_position**, **set_win_position** verranno analizzate nel dettaglio;
LCD118 Init e **Clear Drawing Cache** sono funzioni della libreria del display descritte precedentemente, che consentono di configurare i registri interni del dispositivo e di cancellare ogni possibile dato nella memoria volatile del display;

Infine utilizziamo la funzione **Show String** per visualizzare i dati come mostrato nella **Fig. 4**.
Le funzioni hanno lo scopo di salvare in memoria i dati relativi alle possibili 9 posizioni di pedina e selettore, ma anche le possibili combinazioni di vittoria. Cominciamo dalla funzione **set_pawn_position**, in riferimento alla **Fig10a**, che richiede i parametri *start_x*, *start_y*, *dim* e *offset*; questi in fase di chiamata verranno identificati rispettivamente con le coordinate della griglia (*S_base_x* e *S_base_y*), con la dimensione del quadrante griglia (*_dim*) ed infine con l'*offset_pedina*. Perciò definiamo la funzione e quindi la creazione delle variabili:

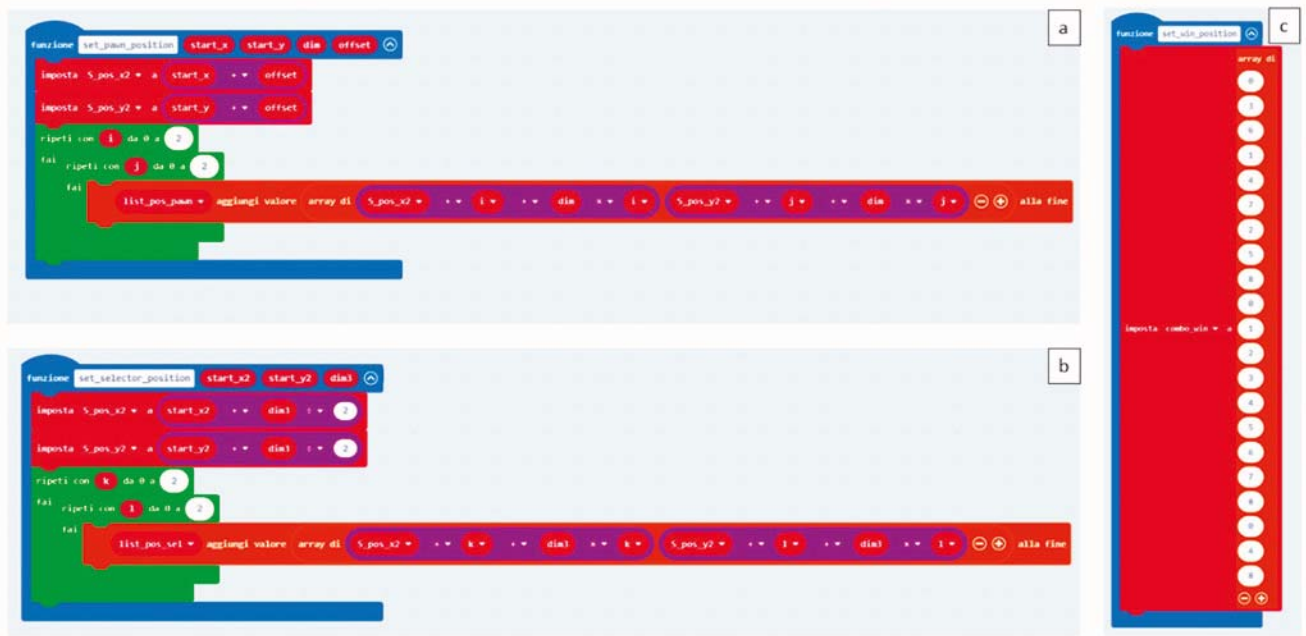
- *S_pos_x2*, *S_pos_y2* tengono conto della posizione della griglia e dell'offset della pedina;
- la lista *list_pos_pawn* sarà riempita con le coordinate della pedina, le quali nella **Fig. 8** sono scritte in blu relativamente al pixel per la creazione della pedina per ogni quadrante.

La funzione **set_selector_position** di **Fig. 10** (è evidenziata dalla lettera b) utilizza lo stesso metodo della funzione **set_pawn_position** (a) con la differenza che le coordinate in questione saranno identificate al centro di ogni quadrante come indicato dalle frecce rosse di **Fig. 8**. L'ultima funzione **set_win_position** (la c in **Fig. 10**) ha lo scopo di raccogliere in un unico array *combo_win* le possibili combinazioni vittoria con sequenze di tre elementi consecutivi ed in linea con la descrizione definita precedentemente.

INIZIO DEL GIOCO

In linea con il diagramma di flusso descritto, attendiamo l'azione utente e quindi la pressione del tasto A che dal punto di vista implementativo si traduce nel trascinare la funzione **quando premi pulsante A** presente nella sezione *Fondamentali* e definire una macchina a stati; quest'ultima risulta necessaria in quanto, come dicevamo, il tasto A verrà utilizzato anche per spostare il selettore. Con riferimento alla **Fig. 11** eseguiamo quanto segue.

- Aggiungiamo all'interno della sezione relativa alla pressione tasto un **if-else** con argomento la variabile *btn_A_sts*, che in corrispondenza della pressione del tasto A avrà valore 0 e pertanto porterà all'incremento della medesima variabile e della variabile *system_sts* che rappresenta lo stato del sistema come suggerisce il nome. Da questo momento in poi, le successive pressioni del tasto andranno ad identificare lo spostamento verticale del selettore che andremo ad analizzare in seguito.



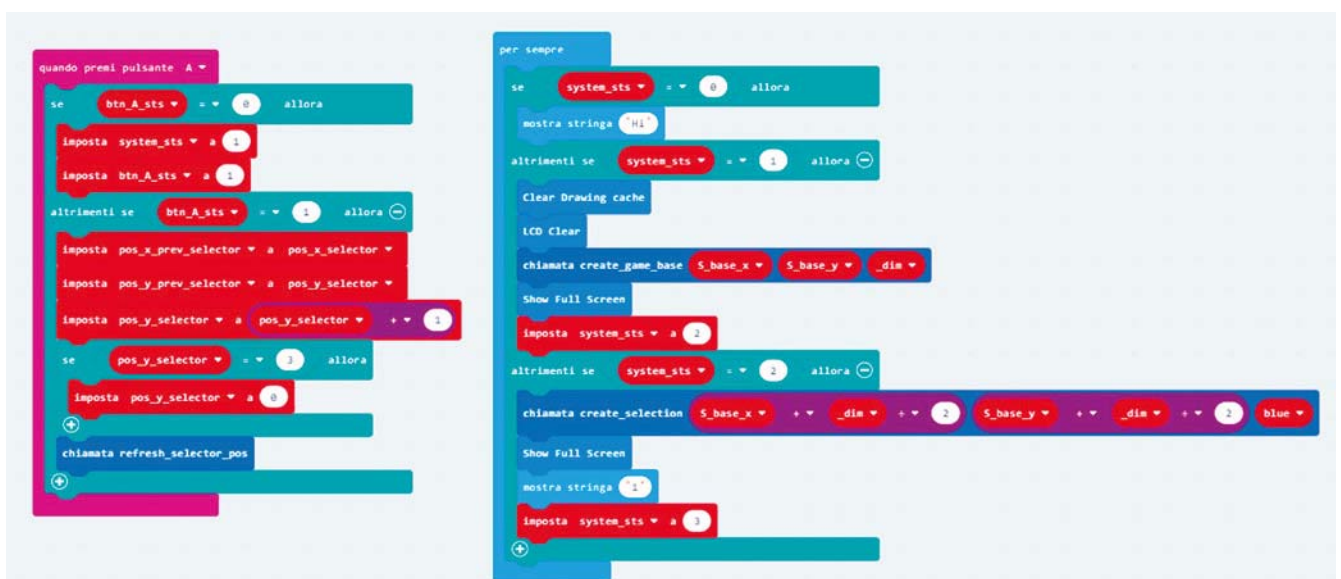
- In parallelo la funzione **per sempre** risulta sempre attiva pertanto risulta necessario, tramite un blocco di if-else con variabile `system_sts`, assegnare una priorità alle operazioni da effettuare.
- Alla pressione del tasto A, la variabile `system_sts` assume valore 1 con conseguente creazione della griglia (**create_game_base**), dopo aver ripulito il display dai dati visualizzati.
- Al successivo incremento della variabile `system_sts` viene creato il selettore e mostrato sulla

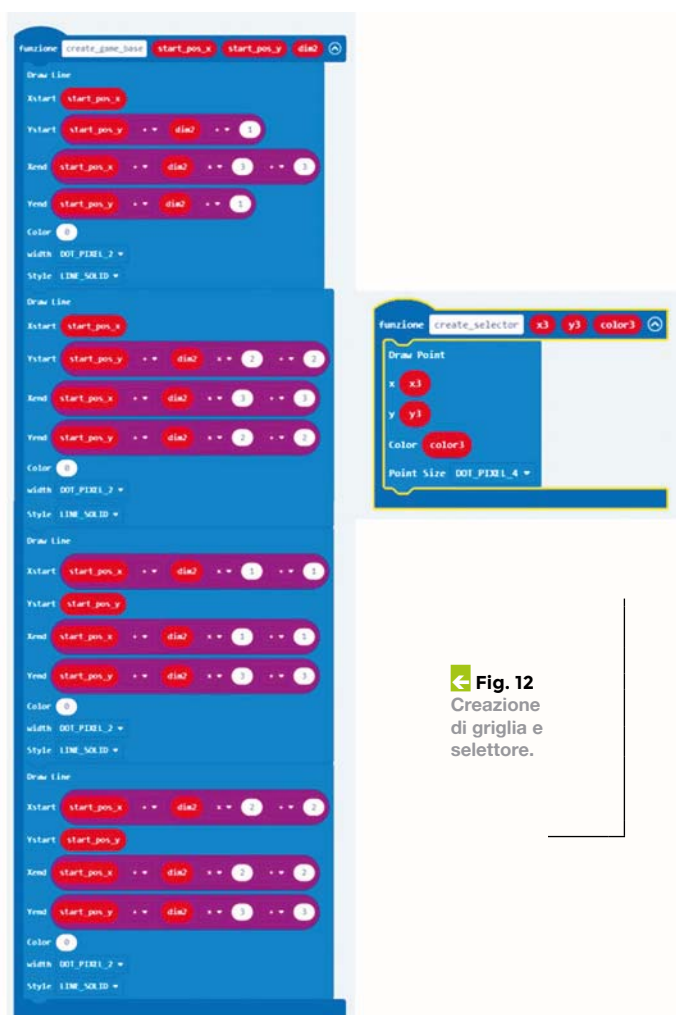
matrice led del Microbit il numero 1 per indicare il turno del giocatore.

Per la creazione della griglia la funzione **create_game_base** richiede tre argomenti identificati dalle coordinate e dalla dimensione del quadrante. Come in **Fig. 5**, decidiamo di utilizzare la libreria del display **Draw Line** e quindi nella definizione di due punti (quattro coordinate) per ogni linea con lo scopo di rappresentare la griglia attraverso quattro linee. Tra i parametri occorre:

Fig. 10
Funzioni di Init.

Fig. 11
Avvio del gioco.





← Fig. 12
Creazione
di griglia e
selettore.

- impostare *Color* a 0 (indicativo del colore nero);
- impostare *Width* al valore *DOT_PIXEL_2* così da definire una linea più spessa;
- impostare *Style* a *LINE_SOLID* al fine di definire una linea continua.

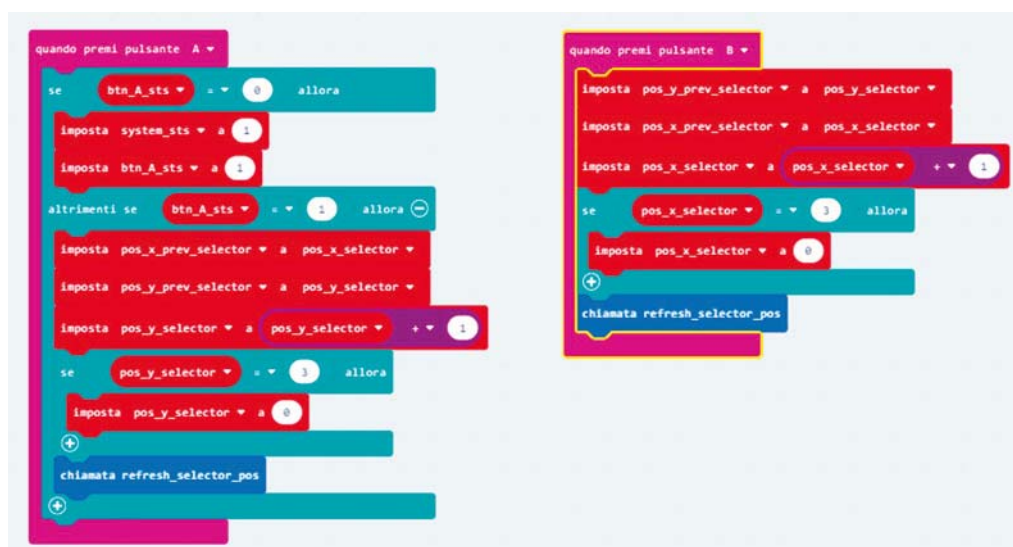
Procediamo nella realizzazione del selettore con la funzione **create_selector** e quindi nell'utilizzo della funzione **Draw Point** (Fig. 12) che non richiede particolari processi implementativi ma l'assegnazione delle corrispondenti variabili e la dimensione del punto *Point Size* al valore *DOT_PIXEL_4*. In fase di chiamata funzione è importante esplicitare le coordinate del punto che dovrà posizionarsi al centro del riquadro ($_dim/2$) ed il colore che dovrà essere assegnato alla variabile *blu* inizialmente dichiarata.

SPOSTAMENTO DEL SELETORE

Con l'implementazione e la creazione del piano di gioco, il sistema rimane in attesa della successiva azione utente che spetterà al giocatore 1 il quale potrà spostare il selettore tramite i tasti A e B della micro:bit.

La nuova posizione del selettore viene definita attraverso quattro variabili che hanno la funzione di identificare le nuove coordinate x e y (*pos_x_selector*, *pos_y_selector*) ed avere memoria anche di quelle precedenti tramite *pos_x_prev_selector* e *pos_y_prev_selector*. In corrispondenza della pressione tasto A andremo a salvare le attuali coordinate e incrementare (+1) la coordinata y per la movimentazione verticale; la pressione del tasto B implementa le stesse funzionalità con la

→ Fig. 13
Posizionamento
selettore.



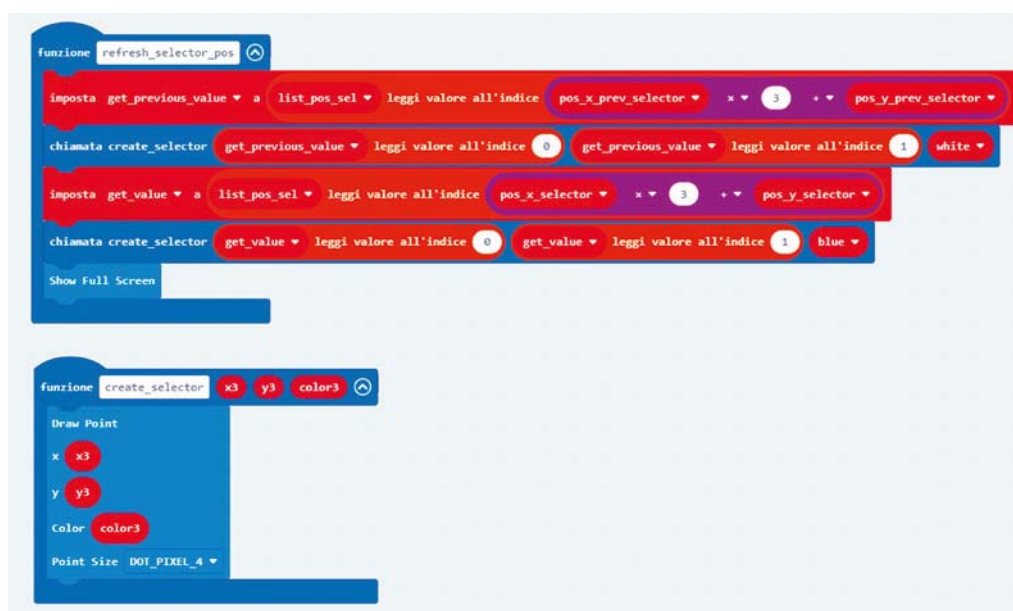


Fig. 14
Aggiornamento
Posizione
Selettore.

differenza che la variabile da aggiornare (+1) sarà la coordinata x (**Fig. 13**). Se l'incremento eccede del valore 2 per entrambe le coordinate allora verrà assegnato il valore 0 così da simulare la movimentazione endless.

L'aggiornamento delle variabili appena descritte determina l'aggiornamento della posizione del selettore, come mostrato in **Fig. 14** attraverso la funzione **refresh_selector_pos**.

Se ricordate, in **Fig. 10** (punto b) abbiamo descritto la funzione che consente di creare e riempire la variabile globale *list_pos_sel* con le possibili coordinate del selettore definendo una lista di 9 elementi contenenti le coordinate x,y che a questo

punto andiamo ad utilizzare all'interno della nostra funzione, tuttavia per comprendere al meglio la logica da implementare consideriamo la **Fig. 15**. *List_pos_sel* può essere schematizzato come mostrato, mentre le variabili precedentemente definite *pos_x_prev_selector*, *pos_y_prev_selector*, *pos_x_selector* e *pos_y_selector* possono assumere i valori da 0 a 2 come evidenziato per la griglia base del Tris, pertanto è possibile risalire agli elementi della lista sfruttando la relazione $coordinata_x * 3 + coordinata_y$ che lega un oggetto monodimensionale con un altro bidimensionale quale la matrice. Supponendo che *pos_x_selector* e *pos_y_selector* abbiano entrambi valore 1



Fig. 15
Ricerca
posizione.

Fig. 16
Conferma
Inserimento pedina.



(coordinata x_{S4} , y_{S4}), il risultato della relazione sarà 4 che trova corrispondenza nella lista *list_pos_sel*. Quindi ritornando all'implementazione di **Fig. 14** e facciamo quanto segue.

- Creiamo l'array *get_previous_value* che dovrà contenere gli elementi x,y della lista *list_pos_sel* in corrispondenza del risultato della relazione definita tramite le variabili *pos_x_prev_selector* e *pos_y_prev_selector*. La funzione **leggi valore all'indice** è disponibile all'interno della sezione Array.
- Utilizziamo la funzione **create_selector** (precedentemente descritta) che riceve le coordinate contenute in posizione 0 e 1 del vettore *get_previous_value* ed il colore associato alla variabile *white* con lo scopo di andare a rimuovere quella che era la posizione precedente del selettore.
- Creiamo l'array *get_value* che dovrà contenere gli elementi (x,y) della lista *list_pos_sel* in corrispondenza del risultato della relazione definita tramite le variabili *pos_x_selector* e *pos_y_selector*.
- Utilizziamo la funzione **create_selector** che riceve le coordinate contenute in posizione 0 e 1 del vettore *get_value* ed il colore associato alla variabile *blu* con l'intento di aggiornare e

quindi scrivere su display la nuova posizione del selettore.

CONFERMA DELLA POSIZIONE PEDINA

Quando il giocatore di turno decide, tramite il selettore, la successiva posizione che la pedina dovrà assumere, dovrà confermare tale posizione tramite la pressione del logo touch che illustreremo a display attraverso l'implementazione proposta nella **Fig. 16**.

Sempre nella sezione *Fondamentali* è possibile trascinare il blocco relativo alla funzione del logo e così procedere con l'operazione di scrittura su display della corrispondente pedina e nella valutazione delle possibili condizioni vittoria.

Per valutare le condizioni di fine partita dopo l'inserimento di una pedina, è necessario avere memoria delle posizioni occupate dalle pedine di entrambi i giocatori.

Dal punto di vista implementativo, procediamo con la creazione della variabile *index_sel* che è indicativa della successiva posizione x,y della pedina tramite la relazione **coordinata_x * 3 + coordinata_y** e della variabile *status_pawn_refresh* che assume il valore di ritorno della funzione **refresh_pawn_pos** che consente di verificare se è possibile aggiungere la pedina in corrispondenza della posizione del

selettore come evidenziato in **Fig. 17**. La funzione consiste nella creazione di due vettori che indicheremo con `pos_pawn1_busy` e `pos_pawn2_busy` e che hanno lo scopo di tenere traccia delle posizioni (in termini di `index_sel`) per la pedina del giocatore 1 e del giocatore 2 e quindi:

- aggiungiamo il blocco di **if-else** per valutare il turno del giocatore;
- verifichiamo se il vettore `pos_pawn1_busy` contiene elementi tramite la funzione **lunghezza array** disponibile nella sezione Array e successivamente tramite un ulteriore "if" verifichiamo che neanche il secondo vettore `pos_pawn2_busy` contenga la posizione `index_sel`; la funzione restituisce il valore 0 se `index_sel` è presente nel vettore `pos_pawn2_busy` altrimenti aggiunge `index_sel` in coda al vettore `pos_pawn1_busy` e restituisce (*return*) il valore 1;
- nel caso in cui il vettore `pos_pawn1_busy` contenga già elementi al suo interno, andiamo

a valutare se `index_sel` è già presente nei due vettori tramite la funzione **trova indice di** disponibile nella sezione Array e restituisce il valore 1 se l'elemento non è presente aggiungendolo `index_sel` in coda al vettore `pos_pawn1_busy` altrimenti restituisce il valore 0;

- medesime considerazioni possono essere implementate anche per il secondo giocatore utilizzando il vettore `pos_pawn2_busy` come mostrato in **Fig. 17**.

In base al valore restituito e al turno del giocatore riconsideriamo l'implementazione di **Fig. 9**, dove attendiamo che una nuova pedina venga inserita (`status_pawn_refresh == 1`) così da estrarre le coordinate dalla lista `list_pos_pawn` sulla base dell'ultimo `index_sel` aggiunto nella lista `pos_pawn1_busy` per il giocatore 1 oppure `pos_pawn2_busy` per il giocatore 2.

Le coordinate vengono assegnate alla variabile `get_`

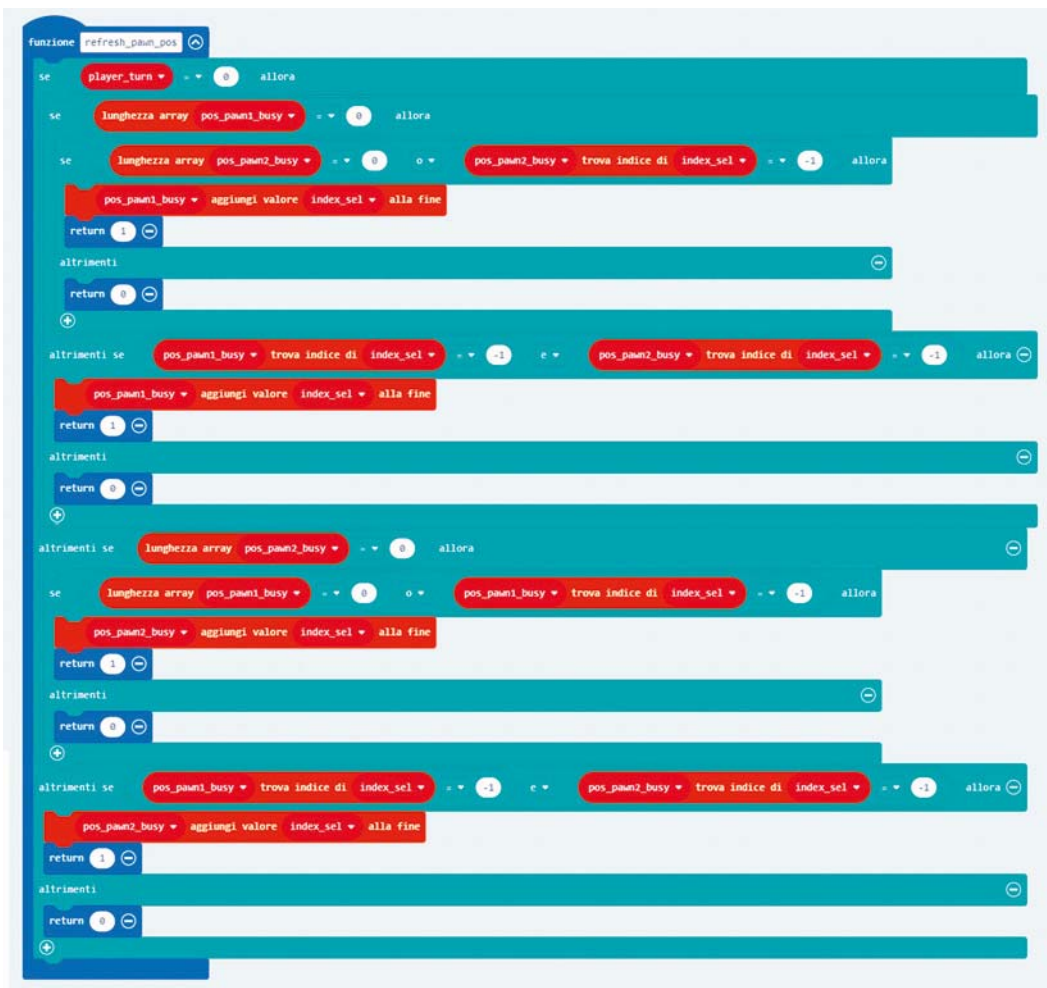


Fig. 17
Conferma
inserimento
pedina.



Fig. 18
Verifica
condizioni
fine partita.

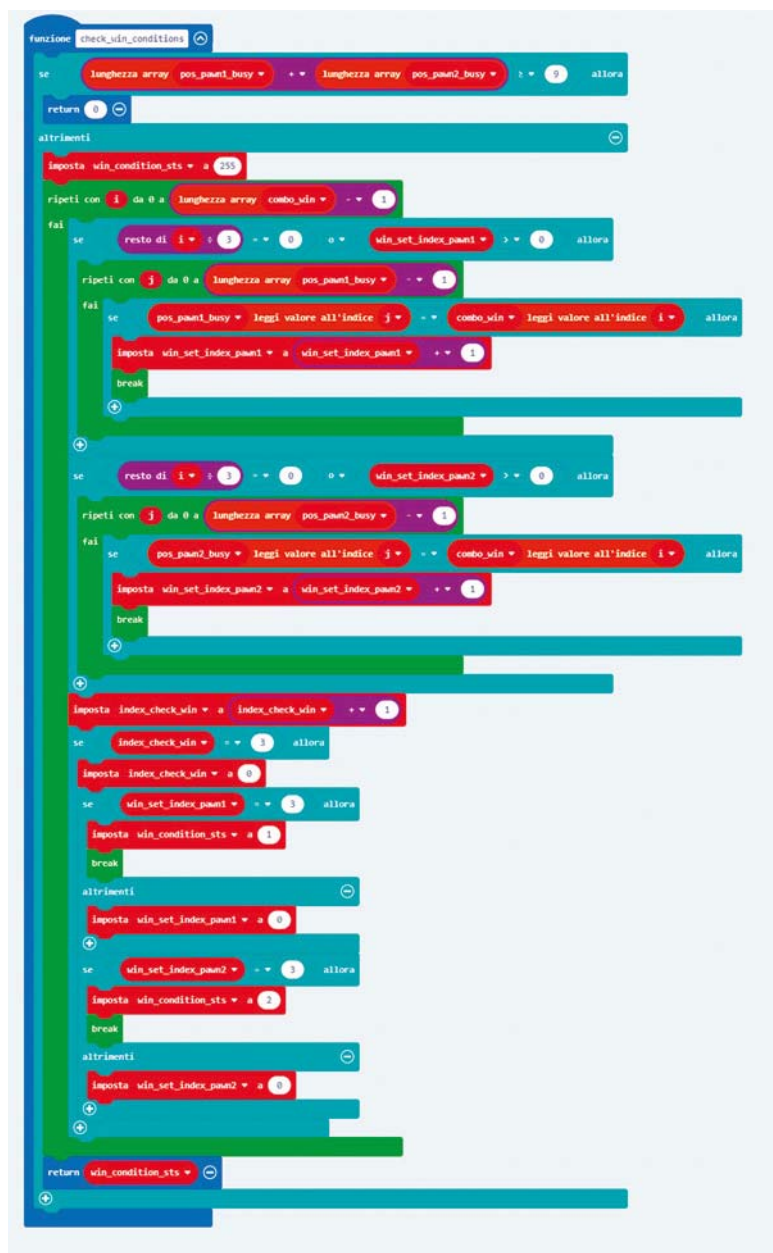


Fig. 19
Verifica condizioni
di vittoria.

value e sulla base dei valori x e y presenti all'indirizzo 0 ed 1 viene creata la pedina tramite la funzione **create_pawn** ed associato il colore rosso nel caso del giocatore 1 ed il colore verde per il giocatore 2. Dopo la creazione della pedina, cambia il turno del giocatore tramite la variabile *player_turn* ed il numero mostrato sulla matrice di LED.

VERIFICA DELLE CONDIZIONI DI FINE GIOCO

Dopo aver ultimato la valutazione della posizione *index_sel* all'interno dei vettori *pos_pawn1_busy* e *pos_pawn2_busy* ed aggiunto la nuova pedina in

base al turno del particolare giocatore, è necessario procedere con la valutazione delle condizioni di gioco definite all'interno della funzione **verify_end_conditions** definita in Fig. 18; tramite essa andremo a valutare se sussistono le condizioni di pareggio o vittoria giocatore 1 o giocatore 2 sulla base delle possibili combinazioni di vittoria. Il primo passo è la creazione della variabile *game_sts* alla quale verrà assegnato il valore di ritorno della funzione **check_win_conditions**, che avrà il compito di verificare le condizioni di vittoria in base all'implementazione definita in Fig. 19.

La prima verifica della funzione **check_win_conditions** è analizzare i vettori *pos_pawn1_busy* e *pos_pawn2_busy* ed in particolare verificare che il numero di elementi sia pari a 9 come il numero di quadranti della griglia; in tal caso verrà restituito il valore 0. Se la prima condizione non è rispettata allora è necessario verificare che gli elementi dei vettori *pos_pawn1_busy* e *pos_pawn2_busy* non contengano una delle combinazioni vittoria definiti (in **Fig. 10c**) tramite il vettore *combo_win* (con elementi *index_sel*), che raggruppati in sottoinsiemi di 3 elementi definiscono le combinazioni di vittoria.

Creiamo la variabile *win_condition_sts* che dovrà essere inizializzata al valore 255 ed aggiungiamo la funzione **ripeti** disponibile nella sezione Cicli per esaminare ogni elemento del vettore *combo_win*. In linea con la struttura del vettore, aggiungiamo un blocco **if-else** con due seguenti condizioni: la prima valuta che l'indice *i* del vettore *combo_win* sia multiplo di 3, indicando l'inizio di un nuovo insieme di elementi; aggiungiamo la funzione **resto di** disponibile nella sezione Matematica; la seconda si basa sulla valutazione della variabile *win_set_index_pawn1*, che consente di proseguire con la valutazione del sotto-insieme se una precedente corrispondenza è stata riscontrata. Lo scopo principale della funzione è confrontare ogni elemento di una possibile combinazione vittoria con gli elementi del vettore che tengono memoria delle posizioni occupate in termini di *index_sel*, per ogni giocatore. Aggiungiamo un blocco **ripeti** con indice *j* da 0 alla lunghezza del vettore *pos_pawn1_busy*, per il giocatore 1 e tra-

mite un blocco **if-else**, verifichiamo che l'*i*-esimo elemento del vettore *combo_win* sia uguale al *j*-esimo elemento del vettore *pos_pawn1_busy*. Se non esiste alcuna occorrenza allora si attende il successivo indice del vettore *combo_win* multiplo di 3, altrimenti la variabile *win_set_index_pawn1* viene aggiornata (+1), interrotto il ciclo "for" tramite un **break** e si prosegue con la valutazione del successivo elemento all'interno del sotto-gruppo di 3 elementi. Analoga implementazione deve essere fatta per il giocatore 2 e le variabili interessate come *win_set_index_pawn2*. Sempre all'interno del ciclo "for" del vettore *combo_win*, creiamo la variabile *index_check_win* che verrà incrementata di 1 per ogni *i*-esimo elemento di *combo_win* ed aggiungiamo un blocco **if-else** che controllerà il valore di *index_check_win* uguale a 3, indicando la fine di un sottoinsieme di combinazione vittoria. All'interno del blocco, azzeriamo la variabile *index_check_win* e verifichiamo il contenuto delle variabili *win_set_index_pawn1* e *win_set_index_pawn2*. Se il valore è inferiore a 3, allora si procederà con il reset al valore zero, altrimenti verrà aggiornata la variabile *win_condition_sts* al valore 1 per il giocatore 1 oppure al valore 2 per il giocatore 2, con conseguente interruzione del ciclo "for" (**break**) e ritorno (**return**) della variabile *win_condition_sts*.

Ritornando all'implementazione di **Fig. 18**, in base al valore della variabile *game_sts* e quindi al valore restituito dalla funzione **check_win_conditions**, andiamo a visualizzare su display l'esito della partita. Il primo blocco **if-else** verifica che il valore della variabile sia diverso da 255 mentre il

Fig. 20
Presentazione
gioco del Tris.

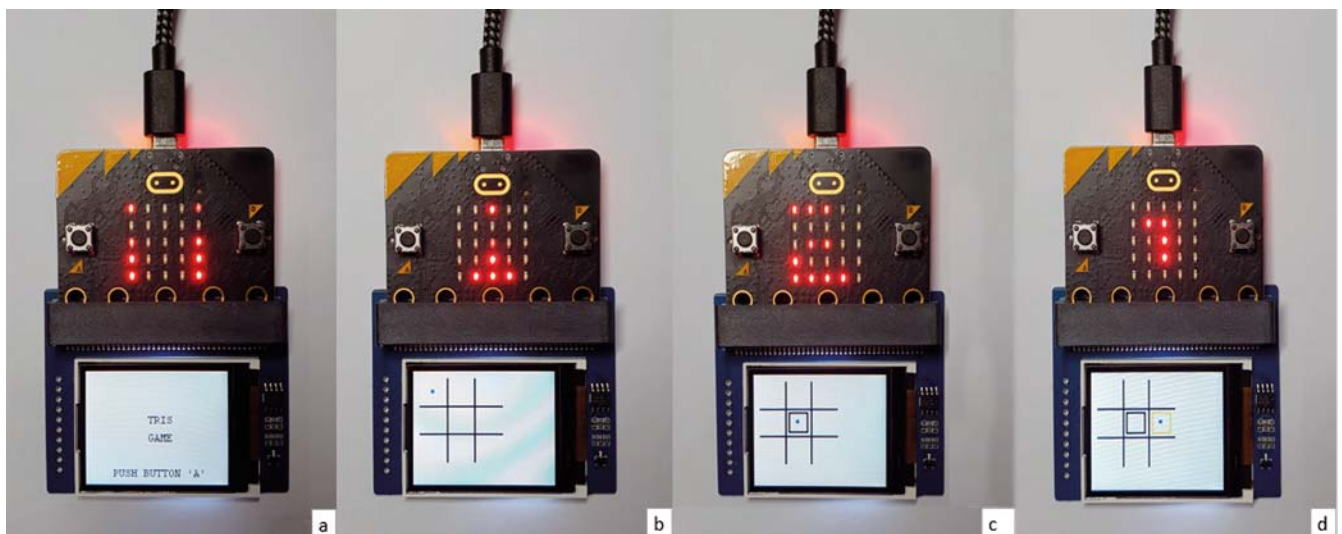
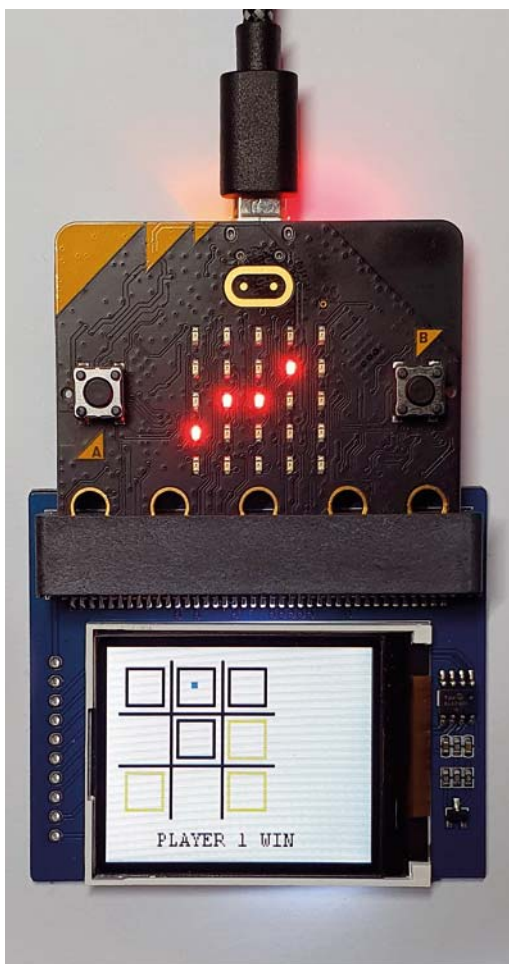


Fig. 21
Fine partita gioco
del Tris.



successivo **if-else** analizza il risultato della partita ed in particolare:

- se il valore di `game_sts` è 0 allora saranno presenti le condizioni di pareggio e la variabile `str_sts_game` verrà impostata al valore "TIE" e mostrato sulla matrice led del Microbit la lettera T;
- se il valore di `game_sts` è 1 allora il giocatore 1 avrà vinto la partita e la variabile `str_sts_game` verrà impostata al valore "GIOCATORE 1 WIN" e mostrato sulla matrice led del Microbit la lettera W;
- se il valore di `game_sts` è 2 allora il giocatore 2 avrà vinto la partita e la variabile `str_sts_game` verrà impostata al valore "GIOCATORE 2 WIN" e mostrato sulla matrice led del Microbit la lettera W.

Al termine deve essere utilizzata la funzione display **Show String** con argomento la variabile `str_sts_game` seguita da **Show Full Screen** per mostrare su display l'esito della partita.

... E ORA SI GIOCA!

Completata la definizione dei blocchi e quindi l'implementazione software, è il momento di giocare. Alcune fasi di gioco sono riportate nella **Fig. 20**. Qui vedete le fasi più importanti dell'implementazione descritte nei precedenti paragrafi:

- in (a) sono mostrate le prime informazioni mostrate su display all'avvio con le indicazioni che l'utente dovrà seguire (pressione del tasto A) per iniziare la partita;
- in (b) è proposto il risultato dopo la pressione del tasto con la creazione della griglia del gioco Tris ed il posizionamento del selettore;
- in (c) e in (d) vedete le pedine per il giocatore 1 e 2 dopo la pressione del logo touch.

Infine la **Fig. 21** mostra una situazione di fine partita con la vittoria del giocatore 1.

CONCLUSIONI

Vi abbiamo guidato nella realizzazione della riedizione "digitale" dello storico gioco del Tris, partendo dalla presentazione del display LCD lato hardware e tutta l'implementazione lato software, spiegando nei minimi dettagli tutte le fasi, le funzioni e le variabili utilizzate, per dare l'idea della facilità di realizzazione e di utilizzo della scheda micro:bit; infatti permette, attraverso le sue librerie, di astrarre l'utente dalla complessità hardware, agevolando soprattutto l'informatizzazione dei giovani/giovanissimi, avvicinandoli così al mondo dell'elettronica.

Buon divertimento con questo gioco e con micro:bit.



Cosa occorre?

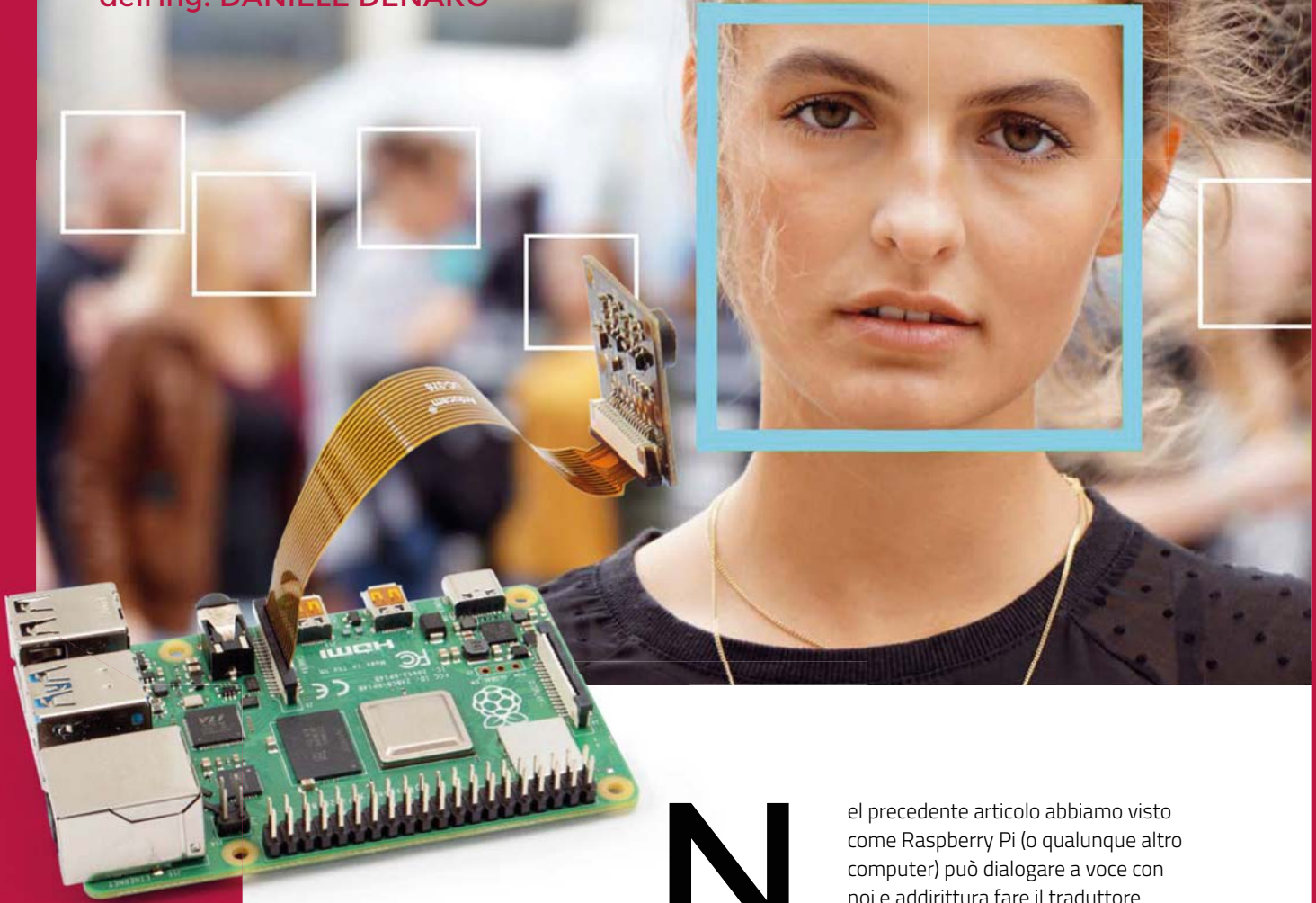
I componenti utilizzati in questo progetto sono disponibili da Futura Elettronica. La board micro:bit V2 è disponibile con lo starterkit (cod. MICROBITQSKIT) a Euro 39,90, il display per micro:bit (cod. 1.8INCH-MBIT) è in vendita a Euro 18,50. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

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

RASPERRY PI CI RICONOSCE

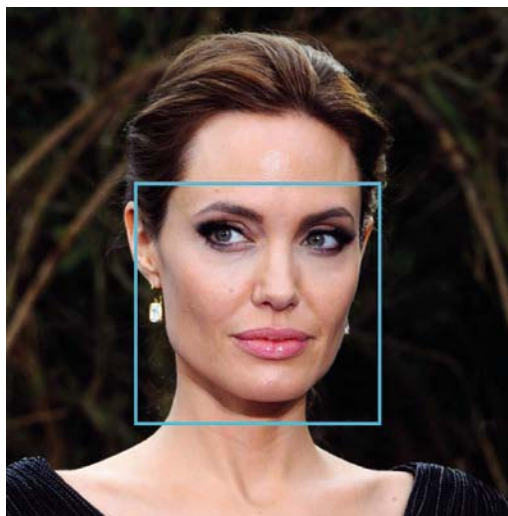
dell'ing. DANIELE DENARO



Riconosciamo i volti umani utilizzando la nostra scheda Raspberry Pi e una libreria di riconoscimento d'immagine.

Nel precedente articolo abbiamo visto come Raspberry Pi (o qualunque altro computer) può dialogare a voce con noi e addirittura fare il traduttore simultaneo. L'unico limite è che per fare ciò deve disporre di una connessione di rete (cablata o wireless) che le permetta l'accesso ad alcuni servizi Internet di supporto. Infatti il motore software vero e proprio che permette di svolgere i compiti anzidetti non risiede, né viene eseguito sulla board, bensì si trova su potenti server di Google, non tanto perché il compito sarebbe particolarmente impegnativo, quanto piuttosto perché, sia il riconoscimento del parlato che la sintesi vocale, hanno dei ben definiti ed importanti campi di applicazione commerciali, con conseguenti interessi economici che spingono verso il mantenimento nei Data Center. Non è strano, quindi, che dati i notevoli inve-

Fig. 1
Ritaglio del viso.



stimenti nella ricerca, le grosse società operanti in tali settori siano restie a rendere disponibile il loro software. L'alternativa open source esiste, grazie a ricercatori accademici e appassionati, ma rimane confinato, spesso, alle lingue più diffuse come quella inglese. In ogni caso abbiamo visto che il funzionamento non risente di questa sponda remotizzata ed appare come se fosse completamente contenuto nell'hardware locale. Nel caso delle immagini, invece, la lingua dell'utente non ha importanza e per questa ragione, è possibile realizzare sistemi di riconoscimento video facendo affidamento sul mondo dell'open source. È possibile e lo abbiamo fatto per sviluppare il progetto proposto in questo articolo, nel quale infatti affronteremo il problema del riconoscimento del viso nei fotogrammi catturati da una fotocamera abbinata all'immane scheda Raspberry Pi. Questa funzionalità la potremo sfruttare per farci riconoscere e poter azionare, per esempio a voce, qualche attuttore. Oppure la potremo utilizzare per riconoscere dei visi tra quelli presenti in un'apposita lista.

Prima di entrare nel vivo del progetto e quindi dell'hardware e del software da cui è composto, riteniamo utile un accenno alla problematica del riconoscimento facciale, facendo una carrellata propedeutica sulle soluzioni più comuni esistenti.

RICONOSCIMENTO DEI VOLTI

Sembra che già appena nati, i bambini siano predisposti a riconoscere l'orientazione e i tratti caratteristici di un viso, tanto da distinguerlo da qualunque altro oggetto nonostante la grezza capacità visiva e la scarsa messa a fuoco. È comprensibile,

quindi, che imparino rapidamente a riconoscere i visi familiari e probabilmente le loro espressioni. Anche da adulti, gli esseri umani hanno una grande capacità di riconoscere e discriminare i volti conosciuti e familiari da tutti gli altri. Ben altra difficoltà è quella che il computer incontra nell'espletare questo compito.

I primi approcci al riconoscimento automatico dei volti sono stati rivolti a identificare tratti caratteristici del volto umano, come sembra fare un neonato, ovvero cercare i pattern stilizzati di occhi, naso e bocca e le loro relazioni geometriche (dati biometrici). Ma questo approccio soffriva molto dell'orientazione del viso rispetto all'elemento di ripresa (telecamera o fotocamera).

In ogni caso prima di passare all'identificazione vera e propria, bisogna isolare il viso all'interno della foto o del frame video, in genere ridotta a immagine monocromatica; questa fase è conosciuta, più propriamente, come "face detection" e porta al riquadro minimo che contiene il viso. Un efficiente algoritmo per l'inquadratura del viso è quello chiamato "HaarCascade" ed è inserito come funzionalità nel software (gratuito) di image processing "OpenCv" (per C o Python). Questo algoritmo assomiglia molto ad un filtro convolutivo come quelli che si trovano nei software per deep-learning. Ma è di tipo binario e molto elementare; in pratica è una maschera per evidenziare pattern del tipo: tratti orizzontali, tratti obliqui e tratti verticali. Questa maschera scandisce l'immagine a diversa scala, in cascata.

Il modello di queste maschere è relativo ad un oggetto specifico che viene determinato da una fase di addestramento. OpenCv contiene alcuni di questi modelli, tra cui il viso umano, e permette di caricarlo ed utilizzarlo per inquadrare i visi anche in tempo reale.

HaarCascade è veloce ma non è l'unico strumento, perché possono essere utilizzati altri sistemi: per esempio, sempre con OpenCv, si può utilizzare il modello LbpCascade che è ancora più veloce di HaarCascade. Questo sistema applica un tipo di maschera diversa (vedi più avanti).

Sempre usando OpenCV come framework è possibile caricare delle reti convolutive sviluppate con "Caffe" o "Tensorflow" (che sono altri due framework specializzati nel deep-learning).

Queste reti convolutive di circa 3/5MB sono più impegnative in termini di memoria occupata (per modelli haarcascade si tratta di circa 1MB) ma più efficaci ed anche veloci, se vengono utilizzati con acceleratori hardware come i sistemi CUDA.

Un altro framework utilizzabile è la libreria "Dlib" (per C o Python). Questa libreria è simile a OpenCv ed è specializzata per i calcoli di machine learning. L'utilizzo di Dlib con l'ultima versione di "cnn_face_detection_model_v1" carica una rete convolutiva semplificata (MMOD ; Max-Margin Object Detection) (circa 0,8MB) che ha una buona velocità se utilizzata con un acceleratore hardware. Un altro metodo più veloce e leggero è quello HOG (circa 200 kB), anche se meno preciso, soprattutto se il viso non è completamente frontale. Il metodo HOG (histogram of oriented gradients) costruisce una mappa delle variazioni di intensità dei pixel che va confrontata con i modelli addestrati a riconoscere i visi. Se la velocità non è il problema principale o se si è dotati di hardware particolarmente potente, il metodo più efficace è il metodo MTCNN (Multi-Task Cascaded Convolutional Neural Network) formato da tre tipi di reti convolutive in cascata: P-Net, R-Net, e O-Net (proposal network che seleziona le aree candidate, refine network che rifinisce i candidati ed infine la output network che estrae i punti caratteristici del viso).

Una volta che il riquadro del viso è stato selezionato ed estratto si può procedere al riconoscimento. Riconoscere un viso significa confrontarlo con i visi contenuti in un database. La cosa più semplice da fare è quella di trasformare l'immagine del viso in un insieme di caratteristiche proprie di quel viso; in tal modo si tratta semplicemente di confrontare questo vettore di valori con quelli contenuti nel database delle facce conosciute. Uno dei primi metodi è stato quello chiamato "Eigenfaces" (auto-facce). Consiste nel ridurre i visi di addestramento in vettori di uno spazio multidimensionale (un valore per ogni pixel dell'immagine) e poi da questo estrarre, tramite metodi statistici come il PCA (Principal Component Analysis), un vettore di uno spazio di dimensione minore, che meglio rappresenta il viso medio. Allora il database delle facce

riconosciute sarà formato da un vettore rapportato a quello medio. Infine un viso da verificare sarà confrontato, dopo la sua elaborazione, con questi vettori del database, e se la distanza (euclidea) sarà inferiore ad una certa soglia il viso sarà riconosciuto.

Poiché "Eigenfaces" è sensibile all'illuminazione è stato superato dal metodo "Fisherfaces", dove lo spazio dei pixel viene ridotto a poche dimensioni e partizionato in classi tramite algoritmi di LDA (Linear Discriminant Analysis).

Un altro sistema che si basa sull'analisi dell'intensità dei pixel è quello LPBH (Local Binary Pattern Histogram). In questo sistema viene applicato una specie di filtro come quelli convolutivi ma basato su una soglia determinata dal pixel centrale. Poi viene preso in considerazione un istogramma dei valori come vettore da confrontare.

Tutti questi sistemi sono apparsi negli anni '90 del secolo scorso, mentre invece negli anni 2000 ha cominciato ad affermarsi l'approccio machine learning nella versione deep-learning. DeepFace (2014) e FaceNet (2015), il primo sistema creato da Facebook ed il secondo creato da Google, hanno portato il riconoscimento dei visi a percentuali di successo pari a quasi il 100% (98/99%), mentre i sistemi del ventennio precedente si fermavano a circa 90%. Questi sistemi basati su reti convolutive tentano di estrarre un certo numero di punti che possano essere presi come invarianti per tutti i visi, e quindi atti a formare un vettore di valori caratteristici per il confronto. Per fare questo sono addestrate su milioni di visi.

Per esempio DeepFace è addestrato con coppie foto-nome. Poiché i visi possono essere anche versioni diverse dello stesso personaggio, l'addestramento tramite back-propagation, finisce per costruire in uno strato interno (il penultimo) un vettore di valori caratteristico di un certo personaggio a parte la foto diversa. La struttura della

Fig. 2
Da foto a
vettore di valori
rappresentativi.



Fig. 3
Struttura DeepFace.

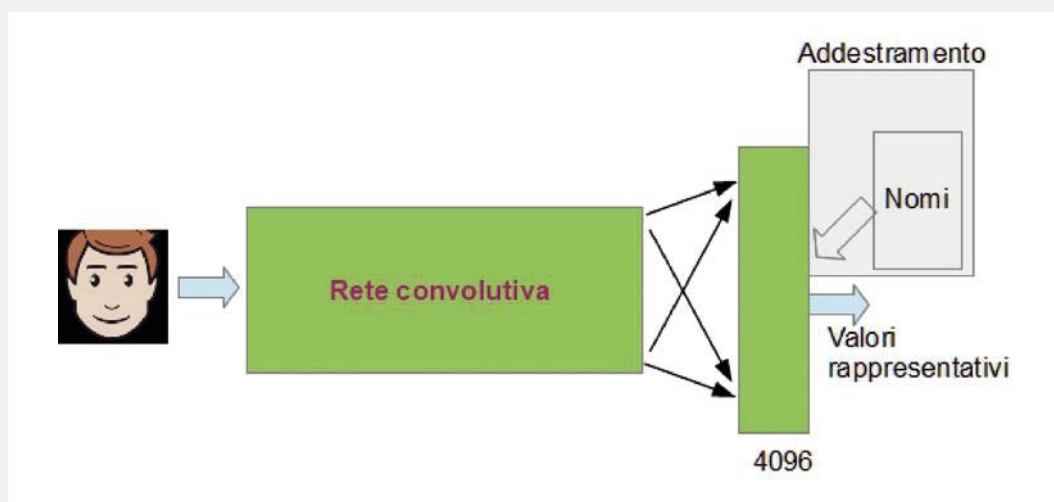
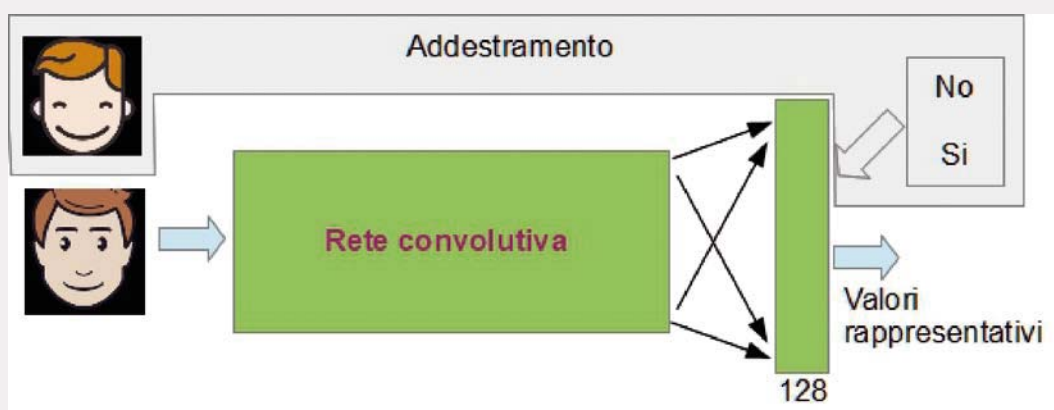


Fig. 4
Struttura FaceNet.



rete è composta da alcuni strati convolutivi che culminano in due strati classici completamente connessi.

L'ultimo strato è quello della classificazione in nomi e viene usato solo per l'addestramento. La rete addestrata (una tantum) viene usata propagando l'immagine fino al penultimo strato di 4096 nodi, che in realtà è ora l'ultimo strato. Questo vettore di 4096 valori rappresenta la codifica dell'immagine e può essere confrontato con gli altri dell'eventuale database per verificare la corrispondenza, per esempio utilizzando la distanza euclidea in uno spazio a 4096 dimensioni.

FaceNet invece di nominare le foto in modo da discriminare quelle dello stesso personaggio dalle altre, presenta le foto simili e quelle diverse alla rete, sotto addestramento, utilizzando una funzione di errore chiamata "triplet". Questo tipo di errore è applicato al vettore di uscita in modo da penalizzare le foto diverse rispetto ad una foto di

riferimento del personaggio (chiamata "anchor") e premiare quelle simili. Anche questa volta la rete convolutiva finisce per capire come estrarre le caratteristiche principali e discriminanti di una foto di un viso. Questa volta il vettore di output (chiamato anche di "embedding"), che sarà utilizzato per identificare le foto in un database, sarà di soli 128 valori. La comparazione è quindi più semplice, ma FaceNet ha dimostrato di essere ancora più affidabile di DeepFace.

Altre soluzioni per il riconoscimento dei visi con l'utilizzo delle reti convolute sono state proposte successivamente a DeepFace e FaceNet; rispetto a queste due, le differenze sostanziali sono sulla profondità della rete convolutiva o sull'estrazione del vettore caratteristico.

VGG-Face (2018) utilizza una rete più profonda rispetto a DeepFace, utilizza l'addestramento tramite nomi come questa ma presenta un vettore di uscita di 2622 valori.

ArcFace, CosFace e SphereFace (2019) differiscono per il tipo di addestramento utilizzato.

Se DeepFace discrimina utilizzando la funzione SoftMax per classificare le immagini e FaceNet utilizza la "triplet", queste ultime proposte suggeriscono metodi più sofisticati.

OpenFace (2017) nasce per essere un software open-source e ricalca la struttura di FaceNet.

OpenFace è implementato con Python, PyTorch, OpenCV o Dlib. L'output della rete addestrata è quindi di 128 valori. La libreria Python "face_recognition" contiene ambedue le fasi per il riconoscimento del viso: la localizzazione dei visi (face detection) realizzata a scelta con il metodo "HOG" (più veloce e predefinito) oppure con il più preciso metodo "MMOD" basato su una rete convolutiva semplificata; ed il vero riconoscimento basato su una versione di OpenFace. La libreria comunque si basa sostanzialmente sulla libreria Dlib che infatti contiene queste funzionalità.

Nel caso di utilizzo del riconoscimento facciale per l'identificazione di sicurezza, esso non basta, infatti potrebbe essere facilmente ingannato da una foto del soggetto. Si passa allora a visioni tridimensionali della faccia, oppure scansioni all'infrarosso dei fotogrammi per la determinazione della temperatura corporea. Oppure, ancora, si utilizzano sistemi compositi che combinano le varie tecnologie di ripresa.

Infine dobbiamo far presente alcune criticità espresse da parte della società sull'uso esteso ed incontrollato del riconoscimento personale e del suo tracciamento: è innegabile che rischi di controlli "Orwelliani" sono sempre dietro l'angolo, perché come ci ha insegnato la Storia, la scienza e la tecnologia possono essere usati nel bene o nel male.

IL PROGETTO

Bene, entriamo ora nel vivo della nostra applicazione: prima di passare al riconoscimento del viso vero e proprio, bisogna isolare il riquadro del viso dalla foto o dal fotogramma (frame) video. In caso contrario, gli altri elementi presenti renderebbero il compito più problematico. In diversi articoli precedenti è stato affrontato il problema del "tracking" (inseguimento) del viso, per cui qui lo affronteremo superficialmente. Negli articoli dedicati al riconoscimento facciale che abbiamo pubblicato in passato, il viso è stato isolato tramite l'algoritmo "HaarCascade" presente nel software open-source OpenCv. Continueremo ad utilizzarlo ma lo affiancheremo anche ad altri algoritmi presenti nella

libreria "Dlib". La libreria Dlib è una libreria predisposta per il *machine learning*. È stata sviluppata in C++ ma esiste anche in versione Python (in realtà semplice *binding*, interfacciamento, della versione originale) su *dlib.net*.

Per installare la versione Python che useremo, il comando è il seguente:

```
pip3 install dlib
```

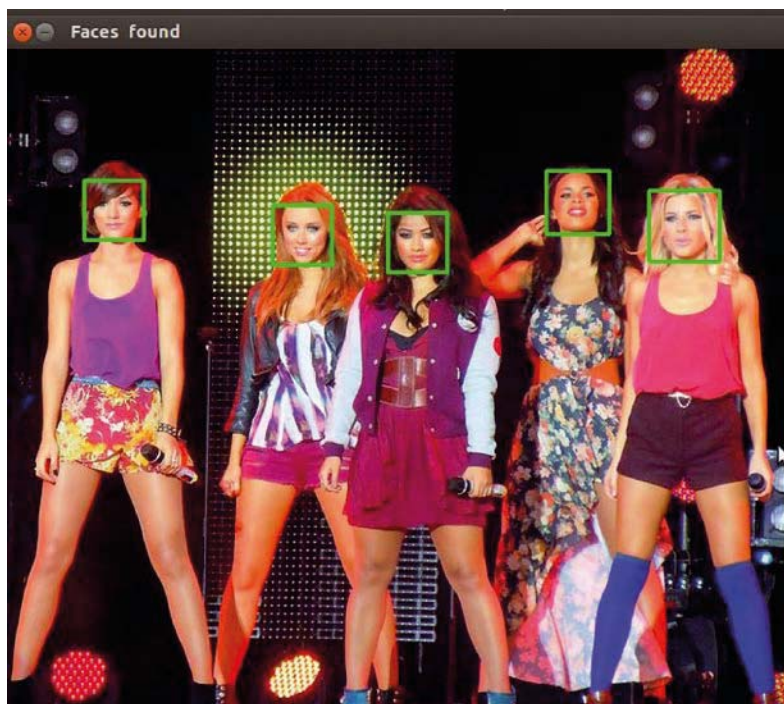
Ma poiché in realtà è una interfaccia della libreria C++, il processo di installazione richiede, in genere, di compilare il sorgente e quindi ha bisogno di trovare nel path dei comandi il programma cmake. Per cui va preventivamente installato se non lo avete già sul computer. Su Raspberry Pi, invece, il comando scaricherà la versione compilata ma ha bisogno di trovare installata la libreria "libatlas" (algebra lineare), ovvero bisogna usare prima il comando:

```
sudo apt-get install libatlas-base-dev
```

Come detto utilizzeremo anche OpenCv. In ogni caso la libreria Dlib prevede di utilizzare la libreria Python PIL che, se non già presente va installata con il solito comando:

```
pip install pillow
```

Fig. 5
Tracciamento
del viso.



↓ Output 1

```
usage: TestCameraDetect.py [-h] [--model {0,1,2,3}] [--saveface] [--camera CAMERA] [--name NAME]
Test face detection
optional arguments:
  -h, --help            show this help message and exit
  --model {0,1,2,3}    0 : HAAR | 1 : LBP | 2 : HOG | 3 : CNN (def: 0)
  --saveface            Save face photo (def: False)
  --camera CAMERA      camera number (def: 0)
  --name NAME          file photo name (def: FacePhoto.jpg)
```

Abbiamo già parlato negli articoli sull'allarme intelligente apparsi nei numeri 257 e 258, delle problematiche di compilazione e installazione di OpenCv. Qui ricordiamo solo che per la massima efficienza è spesso il caso di scaricare i sorgenti e ricompilarli tenendo conto dei molti parametri di compilazione in particolare per abilitare l'utilizzo delle CPU multi-core o delle GPU CUDA.

In ogni caso nel repository https://github.com/open-electronics/Artificial_Intelligence potete trovare indicazioni utili. Sul sito trovate anche il software che riguarda questo articolo, nella sotto cartella "FaceRecognition".

In realtà Dlib non sarà utilizzata direttamente in quanto è la base operativa di una libreria Python specializzata nel riconoscimento dei visi. Questa libreria si chiama appunto "Face Recognition" e la potete installare con il comando:

```
pip3 install face_recognition
```

Mentre la documentazione la potete trovare in: <https://face-recognition.readthedocs.io/en/latest/readme.html#>

Affrontiamo ora la prima parte del problema: l'isolamento del viso. Nel sito www.elettronica.in.it, alla pagina di presentazione di questa rivista, potete trovare il software da scaricare, tra cui lo script "TestCameraDetect.py" in cui vengono testati 4 metodi per il tracciamento del viso. Come hardware sono stati usati un Raspberry Pi 4 da 8Gb e un PC (i7).

- HaarCascade (con OpenCv) : metodo abbastanza veloce ed efficace circa: 28F/s (PC), 6F/s (Raspberry Pi)
 - LbpCascade (con OpenCv) : metodo più veloce circa 38F/s (PC), 10F/s (Raspberry Pi)
 - HOG (con face_recognition-dlib) : metodo efficace ma più lento 7F/s (PC), 1F/s (Raspberry Pi)
 - CNN (con face_recognition-dlib) : impraticabile senza acceleratori hardware < 0.05F/s
- Questo programma (**Output 1**) può scegliere

uno di questi quattro metodi con cui rilevare un eventuale viso frontale dal video della telecamera.

Il viso viene inquadrato in una cornice ed estratto dall'immagine stessa.

La porzione dell'immagine che racchiude il viso può anche essere salvata in un file jpg, in modo che possa essere successivamente usata come immagine da riconoscere.

Per la fase di selezione della porzione di viso, nei successivi script, si è preferito scegliere il modello HaarCascade sia per continuità con gli articoli precedenti, sia soprattutto per il buon compromesso efficacia-velocità.

Veniamo al riconoscimento vero e proprio.

Possiamo immaginare uno scenario in cui la webcam rileva la presenza di un viso nell'inquadratura e lo confronta con un viso di riferimento. Prepariamo, quindi, una foto del nostro viso inquadrato frontalmente. Possiamo usare qualunque metodo e anche il programma "TestCameraDetect.py". E' opportuno che scattiamo la foto nelle condizioni in cui la webcam opera abitualmente. Salviamo il file con la nostra foto con il nome "FacePhoto.jpg" posizionandolo nella directory di lavoro. Per il riconoscimento si può usare il programma "FaceRecon.py".

Questo programma Python, legge e codifica la nostra foto di riferimento, attiva la webcam in un ciclo continuo di rilevamento, in cui estrae le eventuali facce presenti, le codifica e le confronta con quella di riferimento. Nel momento in cui ci riconosce attiva un messaggio vocale ed esce. Chiaramente si è lasciato all'esigenza dell'utente la scelta di implementare una routine da attivare nel momento del riconoscimento.

Infatti potrebbe attivare un computer principale, oppure sbloccare lo stesso computer, o attivare un qualche carico magari mediante comando vocale ecc. ecc. Ricordiamo che nel caso di uso su Raspberry Pi con sessione remota, il messaggio vocale non sarà udito perché attualmente la ses-



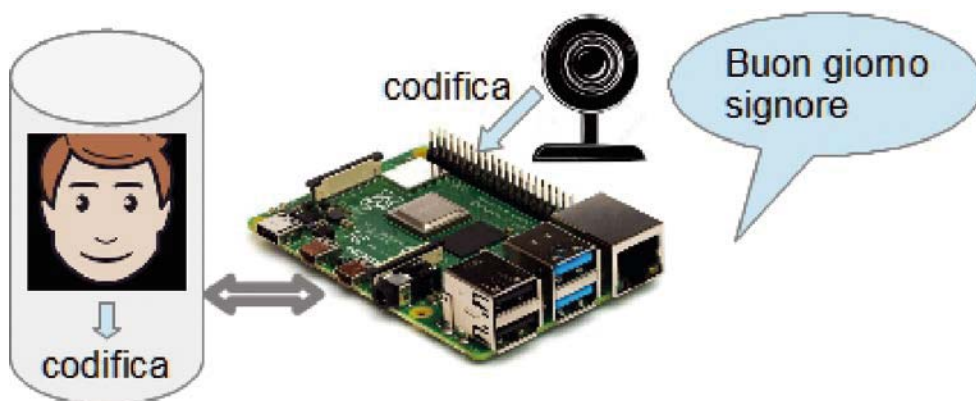


Fig. 6
Riconoscimento con Raspberry Pi.

sione SSH o il desktop remoto VNC non ripetono il sonoro sul computer connesso.

Come detto, l'estrazione dei visi dalle frame video viene fatta tramite l'algoritmo HaarCascade che gira su OpenCv, mentre il riconoscimento viene fatto tramite la codifica dei visi realizzata tramite la pre-addestrata rete convolutiva OpenFace (vedere paragrafo precedente) che è simile a FaceNet di Google. In sostanza OpenFace codifica l'immagine del viso in 128 valori caratteristici. Questi valori identificano un punto (vettore) in uno spazio a 128 dimensioni. Più le foto sono simili più i punti corrispondenti sono vicini. La somiglianza è quindi rappresentata dalla distanza euclidea (normalizza-

ta) di questi punti. Esattamente:

somiglianza = 1 - distanza.

I tempi del riconoscimento (tra immagini codificate) è veramente minimo: inferiore a 0.1ms. Mentre il tempo di codifica della foto del viso, ovvero l'identificazione dei 128 punti caratteristici e rappresentativi tramite la rete CNN è di circa 0.5s. Per cui è questo il tempo da tenere in considerazione. Ne consegue che un confronto multiplo su un data-base di facce codificate prende sostanzialmente lo stesso tempo di un confronto singolo. Infatti il confronto fra due immagini codificate corrisponde alla misura di distanza tra due vettori di uno spazio a 128 dimensioni. Misura sicuramente

↓ Listato 1 - FaceRecon.py.

```
# Routines principali fr sta per "face_recognition"
:
:
def knownencode(knownface):      # carica e codifica la faccia di riferimento
    known_image = fr.load_image_file(knownface)
    knowncode = fr.face_encodings(known_image)[0]
    return knowncode
def facerecogn(wface, known):     # riconoscimento con minima soglia pari a 0.7
    result = fr.compare_faces([known], wface, tolerance=0.7)
    return result[0]              # ritorna vero o falso
def cycle(webcam, known):
    while True:
        okf, frame = webcam.read() # legge frame da webcam
        if not okf:
            print("No frame!")
            continue
        fvis = facedetect(frame)    # rileva l'area della faccia
        if fvis is None:
            continue
        fvis = cv2.cvtColor(fvis, cv2.COLOR_BGR2RGB) # da BGR(opencv) a RGB
        face = fr.face_encodings(fvis)              # codifica l'area del viso
        if len(face) == 0:
            continue
        ok = facerecogn(face[0], known) # tentativo di riconoscimento
        if ok:
            break                          # se positivo esce
```

Output 2

```
usage: FaceRecon.py [-h]
Face recognition by webcam. This program uses webcam 0 and file FacePhoto.jpg (in the same directory) as
reference face. Program exit when recognition has positive ending.
optional arguments:
  -h, --help show this help message and exit
```

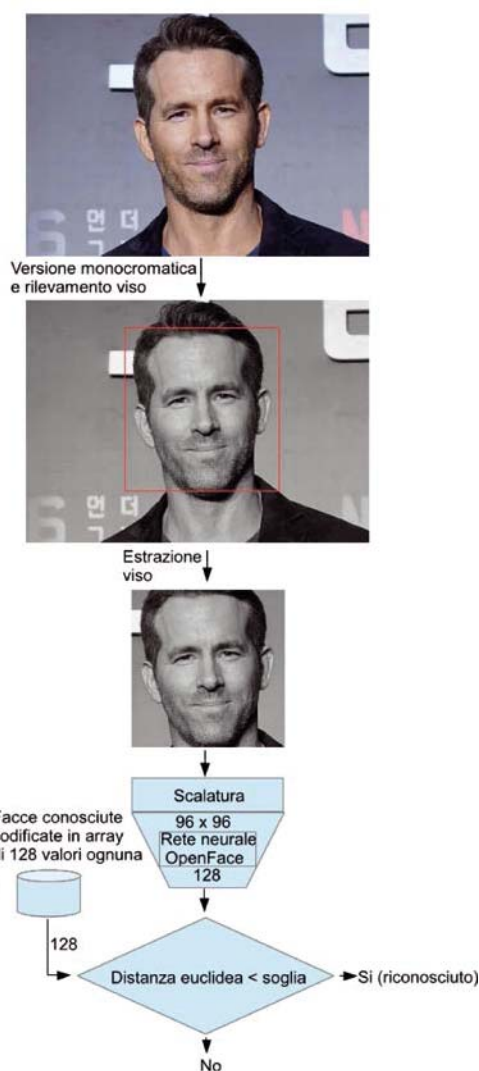


Fig. 7
Flusso operativo di FaceRecon.py.

molto più veloce della propagazione dell'immagine di input attraverso la complessa rete convolutiva OpenFace. Nel **Listato 1** sono descritte le funzioni principali, ma sul nostro sito www.elettronica.in.it trovate il programma completo tra i file di questo numero della rivista.

Nel programma "FaceRecon.py" (**Output 2**) i visi vengono confrontati con un risultato vero o falso... I visi sono considerati uguali se il grado di vicinanza dei due vettori caratteristici è maggiore di una soglia di tolleranza definita nella funzione di comparazione.

Un altro programma presente nel sito: "VerifyPhotos.py" può invece verificare la distanza, ovvero la somiglianza fra due foto di visi, in modo più preciso. Viene lanciata dapprima una finestra di sistema per selezionare il file della foto di riferimento e poi viene lanciata una nuova finestra di selezione per la foto da comparare. Questa volta il risultato non è vero/falso ma il grado di somiglianza tra 1 (identica) e 0 (completamente diversa). Viene anche visualizzato il tempo di codifica e quello del confronto.

CONCLUSIONI

Bene, termina qui la descrizione della nostra applicazione di riconoscimento dei volti basata sulla popolare Raspberry Pi, che ancora una volta si dimostra capace, pur disponendo di un hardware tutto-sommato essenziale, di svolgere compiti impegnativi di cui spesso crediamo capaci solo i supercomputer. □

Cosa occorre?

Raspberry Pi 4 Tipo B con 4GB di memoria (cod. RPI4-4GB) è disponibile da Futura Elettronica a Euro 72,00, 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>

Tutto sotto controllo!

Vuoi stare tranquillo anche quando sei fuori casa? Niente di più facile con le telecamere WiFi motorizzate!

Istruzioni in italiano!



TELECAMERA MOTORIZZATA IP WiFi - 2MPX - ONVIF

Telecamera IP a colori motorizzata con sensore d'immagine CMOS da 2MPX. È dotata di ottica fissa da 3,6 mm e di 11 LED IR che permettono di riprendere immagini perfette anche in condizioni di scarsa luminosità. Supporta la visione da remoto tramite smartphone con sistema Android e iOS. I video possono essere registrati sulla memoria dello smartphone tramite apposita APP, oppure su micro SD Card (max. 128GB - acquistabile separatamente). Dispone di microfono e di altoparlante che permettono di parlare e di essere ascoltati da remoto. Formato di compressione video H.264/MJPEG. Push di allarme su motion detection, possibilità di visualizzare dall'APP fino a 9 telecamere contemporaneamente.



Prezzi IVA inclusa.

TELECAMERA Wi-Fi MOTORIZZATA CON LED DUAL LIGHT - AUDIO BIDIREZIONALE / SD CARD / ETHERNET

Telecamera motorizzata Wi-Fi a colori basata sulla tecnologia ONVIF. Può essere gestita da remoto, tramite APP gratuita, disponibile sia per sistemi Android che iOS. Grazie al sistema PTZ (PAN/TILT/ZOOM) di cui la telecamera è dotata, è possibile orientare, da remoto, l'inquadratura delle immagini in funzione delle proprie esigenze di ripresa. Funzione motion detection con notifica di push sul display dello smartphone. È dotata di 8 LED, dei quali 4 IR e 4 a luce bianca. Integra un microfono e uno speaker, che consentono alla telecamera di parlare e di essere ascoltati da remoto. Offre la possibilità di registrare le immagini riprese, sia sulla memoria interna dello smartphone sul quale è installata l'APP di gestione, sia sulla micro SD card inserita nell'apposito slot della telecamera. Viene fornita completa di staffa di fissaggio.



TELECAMERA IP WiFi MOTORIZZATA CON FUNZIONE SMART TRACKING MOTION 2MPX ONVIF

Mini telecamera IP a colori motorizzata dotata di ottica fissa da 3,6 mm, 6 LED IR, con funzione Smart Tracking Motion (monitoraggio del movimento). Il modulo Wi-Fi integrato consente di collegare la telecamera in remoto permettendo di visualizzare su PC e Smartphone (iOS, Android), quanto ripreso dal suo obiettivo che può ruotare sia orizzontalmente sia verticalmente. Consente inoltre di registrare video su PC oppure sulla micro SD card integrata (max 128 GB, acquistabile separatamente) e visualizzare dall'APP fino a 9 telecamere contemporaneamente. Grazie al microfono e all'altoparlante integrato è possibile comunicare con chi è vicino alla telecamera.



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**



Prezzi IVA inclusa.

DVBT/T2
TERRESTRIAL

**H.265
HEVC**

**FULL HD
1080**

USB

HDMI

mpeg
VIDEO

XVID
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.



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



CONOSCERE E USARE

di FRANCESCO FICILI

4

Node-RED

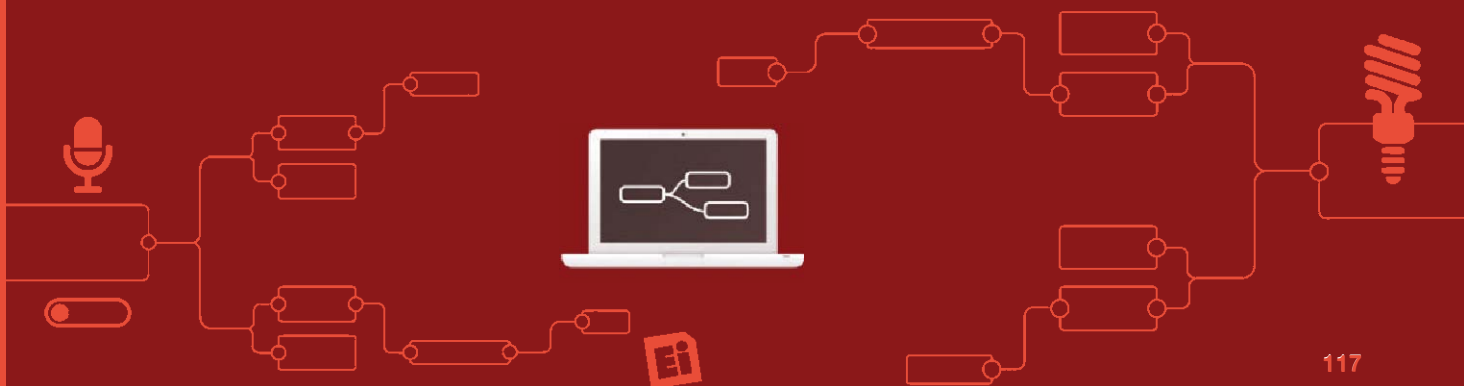
Iniziamo a scoprire Node-RED, un tool di flow-based programming, orientato all'IoT ed alla connettività, originariamente sviluppato dall'IBM Emerging Technology Services team e adesso parte della JS Foundation. In questa puntata ci occupiamo dell'implementazione della parte grafica delle nostre applicazioni, sfruttando la Node-RED dashboard, uno dei pacchetti di nodi aggiuntivi maggiormente utilizzati. Quarta Puntata.

Nella precedente puntata abbiamo iniziato a prendere confidenza con la gestione delle periferiche hardware tramite Node-RED, utilizzando, come di consueto per questo corso, la Raspberry Pi come piattaforma hardware. Nello sviluppo di applicazioni IoT, o comunque orientate alla connettività, l'utilizzo delle periferiche è sicuramente una faccia della medaglia, in quanto è fondamentale per l'interfacciamento dello specifico sensore o attuatore che si desidera rendere "smart" o comunque remotizzare, in modo che sia gestibile attraverso una opportuna interfaccia. La realizzazione della suddetta interfaccia è l'altro lato della medaglia e può essere declinata in diverse forme. Sicuramente una delle forme più comunemente utilizzate è quella grafica, nella modalità in cui l'istanza di Node-RED sulla nostra Raspberry Pi agisce come webserver, fornendo l'interfaccia grafica quando ci colleghiamo tramite

un browser ad uno specifico indirizzo IP. Node-RED ci viene incontro in questa specifica modalità di utilizzo con un componente estremamente interessante: la dashboard. Tuttavia, prima di vedere di cosa si tratta nel dettaglio, dobbiamo affrontare il tema della configurazione, in particolare perché, come vedremo in seguito, alcune modalità di utilizzo dei nostri dispositivi ci espongono a seri rischi di Cybersecurity.

CONFIGURARE NODE-RED

Come abbiamo detto nella parte introduttiva, in questa puntata porremo molto l'accento sulla creazione di interfacce utente remote, tramite Node-RED, utilizzando un componente (la dashboard) che è in grado di esporre parti grafiche interattive su una specifica pagina web, creando di fatto delle webapp. Questa modalità presenta notevoli vantaggi, tra cui la semplicità di utilizzo



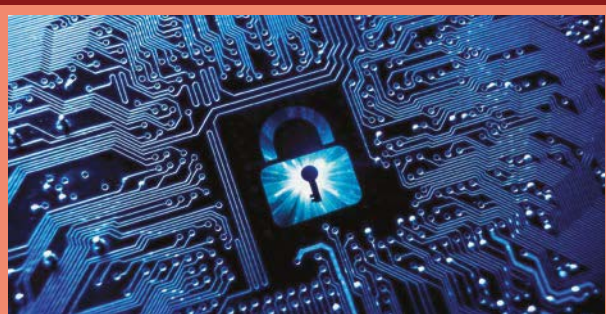


 Fig. 1 - Cybersecurity.

e l'estrema compatibilità. Infatti per gestire uno specifico dispositivo remoto sarà sufficiente collegarci all'indirizzo IP in cui Node-RED espone il webserver e non è importante con quale dispositivo (la webapp è compatibile con i vari smartphone in circolazione, sia Android che iOS, che anche con i vari browser utilizzati in ambiente desktop). Tuttavia se vogliamo remotizzare un'applicazione di questo tipo, uscendo dai confini della nostra rete locale, non abbiamo altra scelta se non quella di aprire una porta del nostro router. Questa azione, inevitabilmente, ci sottopone ad un rischio: se non adeguatamente protetta, la nostra interfaccia web, come anche l'editor di Node-RED, potrebbero essere sfruttati da una terza parte a scopi malevoli. Dobbiamo quindi affrontare un problema di cybersecurity. Fortunatamente Node-RED dispone di tutta una serie di predisposizioni che, se opportunamente attivate, ci consentiranno di esporre i nostri pannelli di controllo al di fuori della nostra rete locale con la ragionevole sicurezza di poter resistere ad attacchi esterni (in cybersecurity non si parla mai di sicurezza assoluta, in quanto avendo un determinato quantitativo di tempo a disposizione, anche le interfacce più sicure possono essere forzate). Prima di addentrarci però nei dettagli relativi alla protezione della nostra istanza di Node-RED è opportuno analizzare come Node-RED affronta il problema della configurazione, che consente anche di attivare le varie features di Cybersecurity integrate. Node-RED, al suo avvio, carica una serie di opzioni di configurazione presenti sul file `settings.js`, che si trova di default nella sua directory principale. Di conseguenza, per configurare il comportamento della nostra istanza di Node-RED possiamo semplicemente intervenire su questo file, attivando o modificando le varie opzioni di configurazione che sono presenti. Le opzioni disponibili sono diverse ed impattano su innumerevoli aspetti relativi all'editor, alla gestione dei files ed al comportamento in generale di Node-RED. La maggior parte delle opzioni sono commentate e corredate di una breve descrizione, per attivarle è quindi sufficiente decommentare la sezione specifica e fare le modifiche del caso. Ad esempio, se volessimo visualizzare l'editor all'interno di uno specifico path, come `"/editor"`, basterebbe

attivare l'opzione `httpAdminRoot`, e modificarla in:

```
httpAdminRoot: '/editor',
```

Riavviando Node-RED (può essere consigliabile riavviare del tutto la Raspberry Pi, tramite il comando `sudo reboot`), non visualizzeremo più il nostro editor su `<localhost>:1880`, ma su `<localhost>:1880/editor`.

Il dettaglio delle varie opzioni di configurazione esula dagli obiettivi di questo corso, ma una descrizione dettagliata delle varie configurazioni possibili può essere reperita dal sito web di Node-RED al seguente link:

<https://nodered.org/docs/user-guide/runtime/configuration>.

OPZIONI DI SECURITY

Ora che abbiamo capito come fare per configurare opportunamente Node-RED, entriamo nel dettaglio delle configurazioni di Cybersecurity che vogliamo mettere in atto per proteggere la nostra istanza di Node-RED nel momento in cui volessimo renderla accessibile dall'esterno della nostra rete locale.

La prima cosa che vogliamo fare è proteggere l'editor, per fare in modo che solo gli utenti autorizzati possano accedere e fare modifiche ai flows. Per garantire questo tipo di protezione di accesso, Node-RED fornisce due sistemi:

- Autenticazione basta su username/password
- Autenticazione basata su OAuth/OpenID providers (come ad esempio Twitter o GitHub)

Noi ci concentreremo solo sulla prima modalità in questa puntata. Per creare utenti con relative credenziali di accesso dobbiamo decommentare e modificare la proprietà `adminAuth` nel file `settings.js` dell'istanza di Node-RED che vogliamo proteggere. La proprietà `adminAuth` dispone di due campi:

- `Type`: rappresenta il tipo di protezione che vogliamo implementare. La modalità username/password è definita "credentials"
- `Users`: è un array di javascript objects che rappresenta i vari utenti che vogliamo creare, con relative password e diritti di accesso.

Quindi, se vogliamo creare due utenti, un utente `admin`, con diritti di lettura e scrittura e un utente `guest` con diritti di sola lettura, dovremo modificare la proprietà come illustrato nel **Listato 1**.

Come si può vedere dal listato le password sono opportunamente crittografate tramite l'algoritmo di hashing `bcrypt`, in modo da non essere conservate in chiaro (nell'esempio gli hash delle due password sono stati ridotti per ragioni di spazio). Per generare gli hash delle password dei nostri utenti possiamo utilizzare direttamente la funzione di hash `bcrypt` integrata in Node-RED (`node-red admin hash-pw`), oppure utilizzare uno dei tanti servizi presenti online, come



Listato 1 – Creazione degli utenti admin e guest

```
adminAuth: {
  type: "credentials",
  users: [
    {
      username: "admin",
      password: "$2b$10$uf4IHPnu01B04pR...",
      permissions: "*"
    },
    {
      username: "guest",
      password: "$2b$10$D4vDDId7yb42kme1...",
      permissions: "read"
    }
  ]
},
```

ad esempio <https://passwordhashing.com/BCrypt>. Una volta creati gli utenti ed assegnati gli hash delle password non ci resta che assegnare i privilegi di accesso ad ogni utente. Nell'esempio di **Listato 1**, all'utente admin sono stati assegnati privilegi illimitati (*), mentre all'utente guest sono stati assegnati privilegi di sola lettura (read). Una volta fatta questa modifica e riavviato Node-RED, se proviamo ad accedere all'editor, verremo re-indirizzati su una pagina di login come quella riportata in **Fig. 2**, e solo inserendo una coppia di credenziali valide potremo effettivamente accedere.

Il meccanismo di autenticazione di Node-RED prevede la creazione di token di accesso, che hanno validità per un certo periodo. Di conseguenza non saremo costretti a re-inserire la password di accesso finché il token rimane valido, e il periodo di validità dei token può essere modificato agendo sulla proprietà *sessionExpiryTime* (il valore di default è di 7 giorni). Si faccia attenzione però a questo settaggio: periodi troppo lunghi potrebbero esporre a rischi legati alla sicurezza, mentre periodi troppo corti comportano la necessità di dover reinserire la password di accesso ogni volta che l'editor lo richiede (anche a runtime, mentre stiamo modificando i nostri flows).

Il meccanismo di protezione che abbiamo creato consente di bloccare l'accesso al nostro editor da parte di soggetti non autorizzati, ma è ancora vulnerabile, perché lo scambio di informazioni avviene su un canale HTTP non protetto. Di conseguenza le nostre credenziali di accesso sono trasmesse in chiaro dal client (il browser che stiamo utilizzando) verso il nostro server Node-RED, e di conseguenza potrebbero essere intercettate. Per eliminare questo rischio dob-

Listato 2 – Configurazione del canale HTTPS

```
https: {
  key: require("fs").readFileSync('/home/pi/.node-red/privatekey.pem'),
  cert: require("fs").readFileSync('/home/pi/.node-red/certificate.pem')
},
```

biamo passare su un canale sicuro di tipo HTTPS. Per farlo è sufficiente decommentare la proprietà https presente sempre sul nostro settings file. Questa proprietà contiene due campi, key e cert, che rappresentano la chiave privata e la catena di certificati necessari per la creazione del canale HTTPS. I campi key e cert non sono altro che i path della chiave privata e del certificato, in formato .pem. Quindi, supponendo di avere una coppia chiave/certificato valida, per re-indirizzare la connessione al nostro server Node-RED su HTTPS dobbiamo inserire una configurazione simile a quella riportata nel **Listato 2**.

Per la creazione della chiave privata e del certificato, è possibile seguire questa guida:

<http://stevesnoderedguide.com/securing-node-red-ssl>.

A questo punto, per raggiungere il nostro editor, non potremo più utilizzare una connessione HTTP, ma dovremo collegarci in HTTPS. Può essere comodo attivare anche la seguente opzione, sempre sul nostro file settings.js:

```
requireHttps: true,
```

che reindirizza le connessioni HTTP su HTTPS automaticamente.

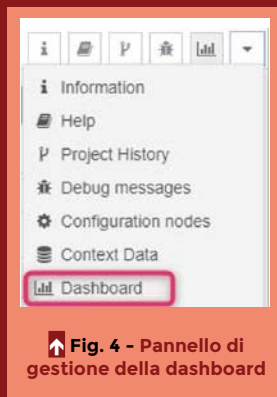
A questo punto il nostro editor è ben protetto, e possiamo quindi decidere di esporlo verso l'esterno. Si faccia comunque sempre molta attenzione a questa operazione, dal momento che aprendo una porta del nostro router verso l'esterno ci esponiamo comunque ad attacchi, quindi si raccomanda sempre di evitare di inserire informazioni sensibili nei dispositivi a cui viene dato accesso e di proteggere sempre le varie interfacce che si espongono (come ad esempio SSH o simili).

NODE-RED DASHBOARD

Ora che la nostra istanza di Node-RED è protetta possiamo passare all'argomento principale di questa puntata, ossia, come accennato nella parte introduttiva, come realizzare interfacce grafiche per i nostri flows.



Fig. 2 - Protezione dell'accesso all'editor di Node-RED con username e password.



Dal momento che Node-RED nasce come un ambiente di sviluppo con forti connotazioni IoT, è stato sin dall'inizio pensato un componente che permetta agli utilizzatori di realizzare semplicemente interfacce grafiche: la dashboard. La dashboard non è altro che un set di nodi (si installa esattamente come gli altri nodi e, una volta installato, compare in una sezione dedicata sulla palette) che permettono di inserire elementi grafici come pulsanti, gauges, grafici e molti altri, che ci permettono di visualizzare dati in real-time e di interagire con altri nodi presenti nei nostri flows. Per installare la dashboard è sufficiente cercare tramite l'opzione "Manage Palette" il componente node-red-dashboard. Una volta installato vedremo nella palette il

gruppo di nodi riportato in Fig. 3.

Oltre ai vari nodi viene installato anche un particolare pannello di gestione, denominato dashboard ed accessibile dalla sidebar (vedi Fig. 4), che ci permette di gestire layout, temi, font ed altre opzioni della parte grafica.

La dashboard è semplicissima da utilizzare, infatti, una volta installata, l'interfaccia grafica della nostra istanza di Node-RED risulta accessibile all'indirizzo:

<Node-RED server URL/IP>:1880\ui

Ossia dobbiamo semplicemente aggiungere "\ui" alla locazione a cui solitamente ci colleghiamo per accedere all'editor. Questo specifico path può anche essere modificato dall'utente, andando semplicemente a modificare la properties "ui" del file *setting.js*, in maniera del tutto simile a come abbiamo fatto per il path dell'editor nella parte iniziale di questa puntata (ricordiamoci sempre di riavviare Node-RED per rendere effettive le modifiche che facciamo al file di configurazione).

DASHBOARD NODES

Come abbiamo accennato in precedenza, la dashboard è composta da una serie di nodi, ed alla quasi totalità di questi nodi corrisponde un elemento grafico, denominato widget, con il quale sia il flow che l'utente possono interagire. E' possibile quindi visualizzare dati provenienti dal flow (ad esempio visualizzare la lettura di un sensore di temperatura su un grafico) oppure recepire input dall'utente ed eseguire azioni (ad esempio, attivare un relè alla pressione di un pulsante).

Facciamo quindi un'analisi di dettaglio dei vari widget che compongono la dashboard:

- **Audio Out:** questo widget permette di riprodurre audio (wav o mp3) o di sintetizzare vocalmente un testo passato come input.
- **Button:** questo widget crea un pulsante virtuale con il quale è possibile interagire. E' possibile impostare lo specifico *msg.payload* da inviare alla pressione del pulsante.
- **Chart:** questo widget crea un grafico, che può essere di vari tipi differenti (grafico a linee, a barre o a torta). Nel grafico vengono rappresentati i dati inviati alla porta di input del nodo, come ad esempio la lettura di un sensore.
- **Colour Picker:** questo widget crea un picker che permette di selezionare un colore.
- **Date Picker:** questo widget crea un picker che permette di selezionare una data.
- **Dropdown:** questo widget crea un dropdown menu e permette di selezionare una delle opzioni. Gli elementi del dropdown possono essere definiti dalla finestra di configurazione del nodo o anche tramite uno specifico campo (*msg.options*) inviato tramite un messaggio in ingresso.

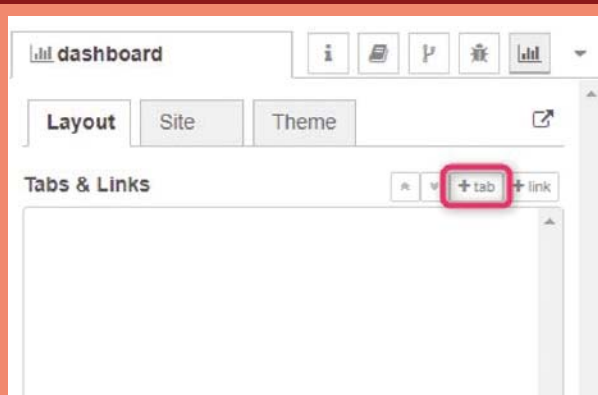


Fig. 5 - Aggiunta di un Tab.

- **Form:** questo widget crea un form i cui campi possono essere definiti tramite la finestra di configurazione del nodo.
- **Gauge:** questo widget crea un gauge che rappresenta il valore istantaneo passato alla porta di ingresso tramite *msg.payload*. Il nodo ha 4 modalità differenti per la gauge: standard, donut, compass e level.
- **Notification:** questo widget è in grado di creare notifiche per l'utente. La notifica può essere di tipo toast (popup che scompare dopo un tempo predefinito) oppure nella forma popup con pulsante di conferma (ed in questo caso può anche essere richiesto un input all'utente).
- **Numeric:** questo widget crea un input di tipo numerico.
- **Slider:** questo widget crea una slider orizzontale controllabile dall'utente.
- **Switch:** questo widget crea uno switch, ossia un pulsante che ha due posizioni stabili, a cui corrispondono due differenti valori di *msg.payload* inviati come output.
- **Text:** questo widget crea un elemento grafico testuale di output, e può quindi essere utilizzato come label. Il contenuto della label viene impostato tramite il valore di *msg.payload* inviato sulla porta in ingresso.
- **Text Input:** questo widget crea un elemento grafico testuale di input. Può supportare diverse modalità, come password, email, telephone e vari tipi di picker.
- **Template:** la dashboard ha anche una sua specifica tipologia di template node, che permette di creare widget personalizzati, tramite HTML e Javascript.
- **Ui control:** questo nodo permette di controllare dinamicamente la dashboard.

Una volta posizionato un nodo della dashboard sul nostro flow, dovremo occuparci di come il relativo widget viene posizionato nel layout visualizzato sulla UI. La dashboard è organizzata in elementi gerarchici che prendono il nome di Tabs e Groups. I Tabs sono gli elementi posizionati gerarchicamente più in alto e rappresentano appunto i vari tab che possono essere visualizzati dalla UI (ne può essere

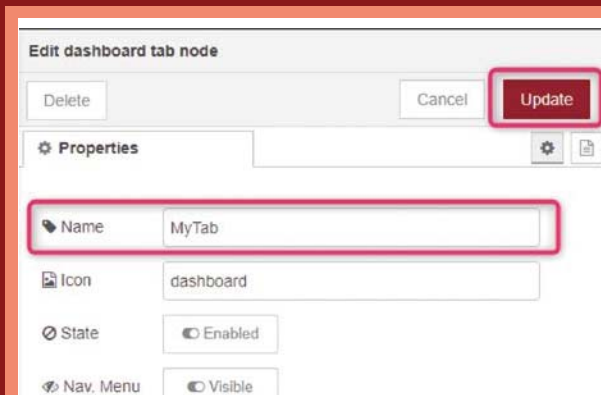


Fig. 6 - Rinominare il TAB.

visualizzato uno per volta). I groups invece sono contenuti nei tab e contengono al loro interno una serie di widgets. Di conseguenza, se vogliamo posizionare dei widget sulla nostra UI, dobbiamo creare almeno un tab ed un group. Questa operazione può essere eseguita sia dalla pagina di gestione della dashboard, che anche dalle singole pagine di configurazione dei vari nodi, nella sezione che permette di associare un nodo/widget ad un group. Nella pagina di gestione della dashboard è possibile anche gestire il layout dei gruppi all'interno dei tab e dei widget all'interno dei gruppi.

ESEMPIO

Vediamo a questo punto un piccolo esempio di come funziona la dashboard, in maniera da comprenderne meglio caratteristiche e funzionalità. Ipotizziamo di voler creare un semplice pannello con un controllo di tipo slider ed uno di tipo gauge, posizionati all'interno di un gruppo su un tab, e di voler controllare il valore visualizzato dal gauge tramite lo slider. Per prima cosa creiamo gruppo e tab, andando a selezionare la sezione dashboard della sidebar e posizionandoci sul tab Layout. Per aggiungere un tab clicchiamo sul pulsante "+ tab", come illustrato in **Fig. 5**. Una volta aggiunto il tab quest'ultimo comparirà nella sezione Tabs & Links; passandoci sopra con il mouse compariranno alla destra dell'elemento stesso tre opzioni: "+ group", "edit" e "link". Selezioniamo l'opzione edit per cambiare il nome al tab. Una volta selezionata l'opzione si aprirà una finestra delle proprietà dalla quale è possibile eseguire alcune operazioni, come rinominare il tab, aggiungere un'icona e altro. Cambiamo il nome da quello di default in MyTab e clicchiamo sul pulsante rosso Update per proseguire, come illustrato in **Fig. 6**.

A questo punto clicchiamo su "+ group" per aggiungere un nuovo gruppo.

Similmente a quanto visto nel caso del tab, anche i gruppi hanno una opzione edit, che possiamo usare per rinominare il gruppo, cambiandone il nome in "MyGroup".

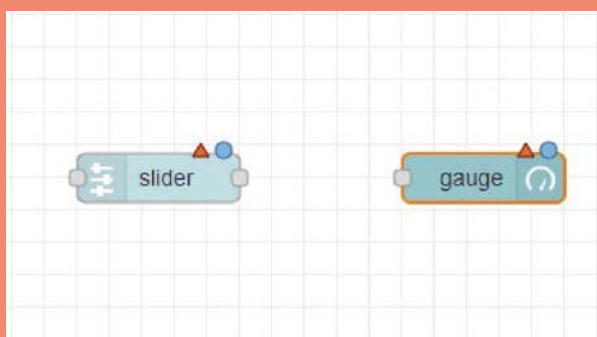


Fig. 7 - Posizionamento di un nodo slider e un nodo gauge sul workspace.

Anche in questo caso abbiamo varie opzioni che per il momento possiamo ignorare. A questo punto la nostra UI dispone di un tab e di un gruppo, quindi possiamo inserire i widget. Selezioniamo, tra i nodi della sezione dashboard, un nodo slider ed un nodo gauge e trasciniamoli sul workspace, come illustrato in **Fig. 7**.

Come si può vedere i nodi presentano le icone a forma di triangoli rossi, ad evidenziare un errore di configurazione (oltre al pallino blu ad indicare che ancora non è stato

eseguito il deploy). Questo perché i nodi non sono ancora stati esplicitamente assegnati a nessun gruppo. Questa assegnazione viene fatta dalla finestra di proprietà del nodo. Iniziamo con il nodo slider, cliccando 2 volte per accedere alla finestra properties. Da questa finestra assegniamo il gruppo tramite il controllo Group (primo dall'alto) e modifichiamo anche i seguenti 2 campi (come illustrato anche in **Fig. 8**):

Name: assegniamo come nome al controllo "MySlider".

Tooltip: aggiungiamo il seguente tooltip, "The slider will control the gauge".

Lasciamo i valori di default per il resto dei controlli e clicchiamo sul pulsante Done per confermare.

Continuiamo modificando le proprietà del nodo gauge, assegnando il medesimo gruppo ed assegnando "MyGauge" come nome al nodo, come illustrato in **Fig. 9**.

A questo punto, dato che vogliamo che lo slider controlli il gauge, colleghiamo con un wire l'output del nodo slider all'input del nodo gauge, ed eseguiamo infine il deploy. Per visualizzare l'interfaccia grafica creata, possiamo collegarci al path interno in cui è visualizzata la ui, ossia, come detto in precedenza:

```
<Node-RED server URL/IP>:1880/ui
```

Dove <Node-RED server URL/IP> rappresenta l'indirizzo della nostra raspberry. In **Fig. 10** è riportato un esempio di output di questo semplice flow con interfaccia grafica.

SECURITY DELLA UI

Ora che abbiamo visto come creare una interfaccia grafica con Node-RED è anche opportuno ritornare nuovamente sul tema della security, in quanto l'opzione di protezione vista all'inizio della puntata è valida solo per l'editor, ma non per la UI. Node-RED ha un secondo set di opzioni per proteggere il pannello della dashboard ed i contenuti statici (come le immagini), se ce ne sono.

Le due opzioni, che potete trovare sempre sul file *settings.js*, sono riportate nel **Listato 3**.

Anche in questo caso si tratta di una coppia username-password, e la password è da inserire esattamente nelle stesse modalità della password per l'accesso all'editor, ossia crittografata con algoritmo bcrypt.

ESEMPIO PRATICO: STAZIONE METEO CON BME280
Passiamo a questo punto all'esempio pratico di questa

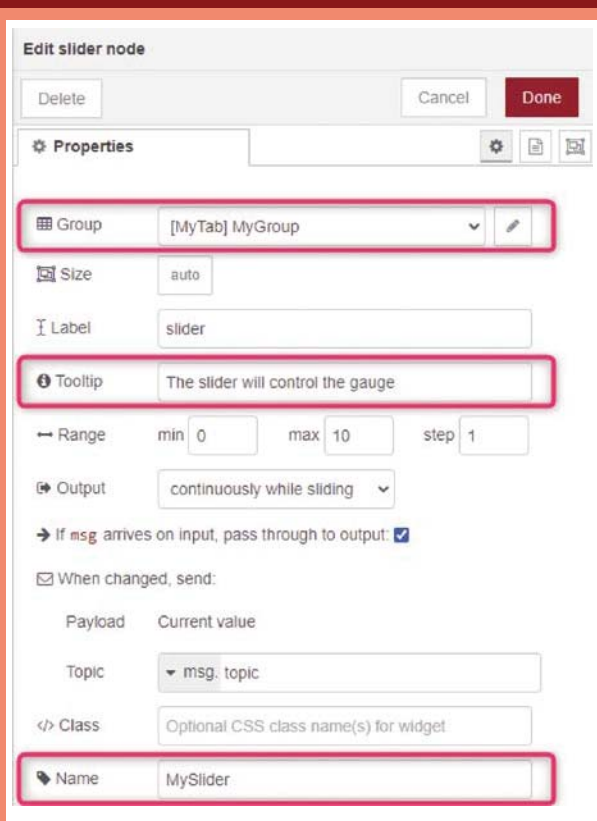


Fig. 8 - Proprietà del nodo slider.

Listato 3 - Opzioni di protezione della dashboard e dei contenuti statici

```
httpNodeAuth: {user:"admin",pass:"$2b$10$uf4IHPnu01B04pR..."},
httpStaticAuth: {user:"admin",pass:"$2b$10$uf4IHPnu01B04pR..."},
```

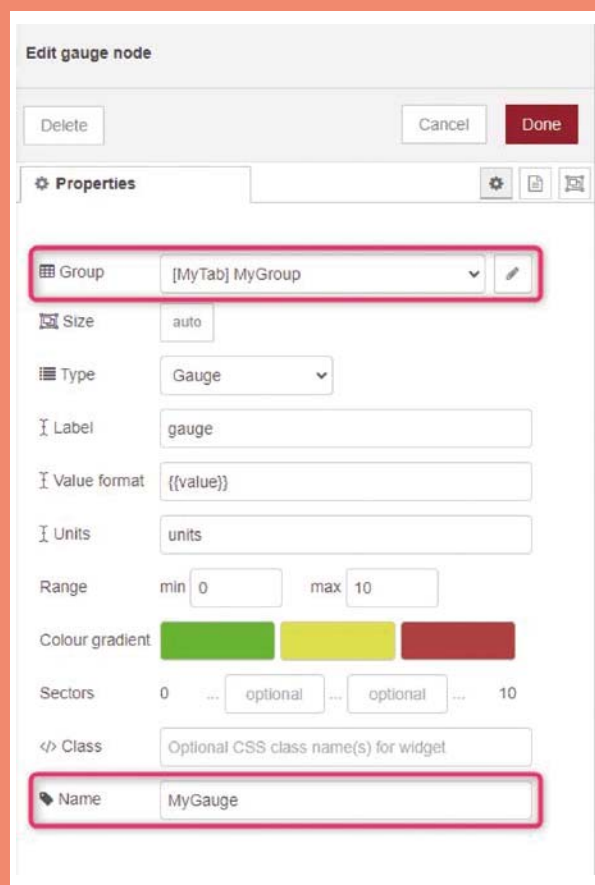


Fig. 9 - Proprietà del nodo gauge.

puntata: quello che ci proponiamo di ottenere è di realizzare una semplice stazione meteo che prenda dati di un sensore di temperatura, umidità e pressione e li visualizzi su una serie di gauge su una pagina web. Come hardware utilizzeremo ovviamente una Raspberry Pi ed il sensore BME280. Tale sensore può essere acquistato presso lo shop di futura elettronica ed ha come codice prodotto la sigla 1606-GY-BME280 (una immagine del sensore è

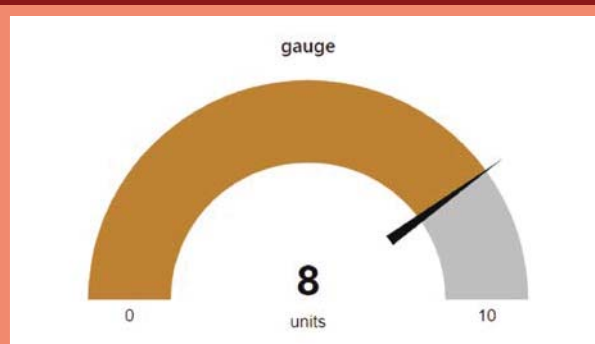


Fig. 10 - Dashboard ottenuta dall'esempio.

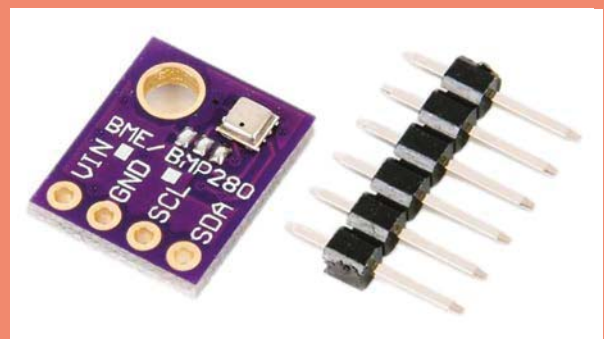


Fig. 11 - Sensore BME280.

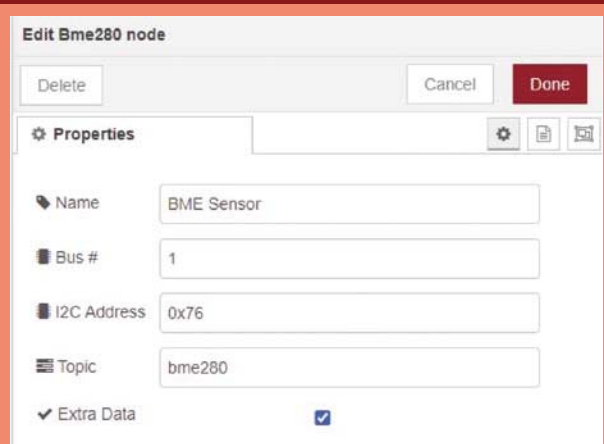


Fig. 12 - Configurazione nodo BME280.

riportata in Fig. 11).

La realizzazione è semplicissima: partiamo installando il nodo specifico che ci permette di comunicare con il sensore. Come di consueto usiamo la pagina di ricerca (manage palette, accessibile dal menu con il simbolo delle tre linee orizzontali in alto a destra), cerchiamo il nodo *node-red-contrib-bme280* ed installiamolo.

Una volta installato il nodo, posizioniamolo sul workspace ed apriamo la sua pagina delle proprietà. Dobbiamo:

- Rinominarlo in BME Sensor
- Assegnare come bus 1
- Assegnare come I2C Address 0x76
- Assegnare come topic bme280
- Spuntare l'opzione extra data (ci permetterà di leggere anche la temperatura apparente) come illustrato in Fig. 12.

Il nodo ha un funzionamento semplicissimo, ogni volta che riceve un messaggio in ingresso, ritorna un object con 4 campi:

- Temperatura (temperature_C)
- Umidità (humidity)
- Pressione barometrica (pressure_hPa)

- Temperatura apparente (heatIndex)

Aggiungiamo adesso gli altri nodi necessari.
Ci occorre:

- un nodo inject per triggerare la lettura periodicamente (decidiamo di leggere 1 volta al secondo)
- 4 nodi change (ci serviranno per estrarre i campi dal messaggio in output dal nodo BME280)
- 3 nodi function (ci serviranno per formattare l'output di umidità, pressione e temperatura apparente, che hanno molte cifre decimali)
- 4 nodi Gauges (per la visualizzazione)

Creiamo anche l'infrastruttura della dashboard, composta da un Tab e da un Group, ed associamo i 4 gauges allo stesso gruppo, ed eseguiamo le seguenti configurazioni.

- Gauge 1:
 - Nome: Temperature
 - Range: -20 – 80
 - Units: °C
- Gauge 2:
 - Nome: Humidity
 - Range: 0 – 100
 - Units: %
- Gauge 3:
 - Nome: Pressure

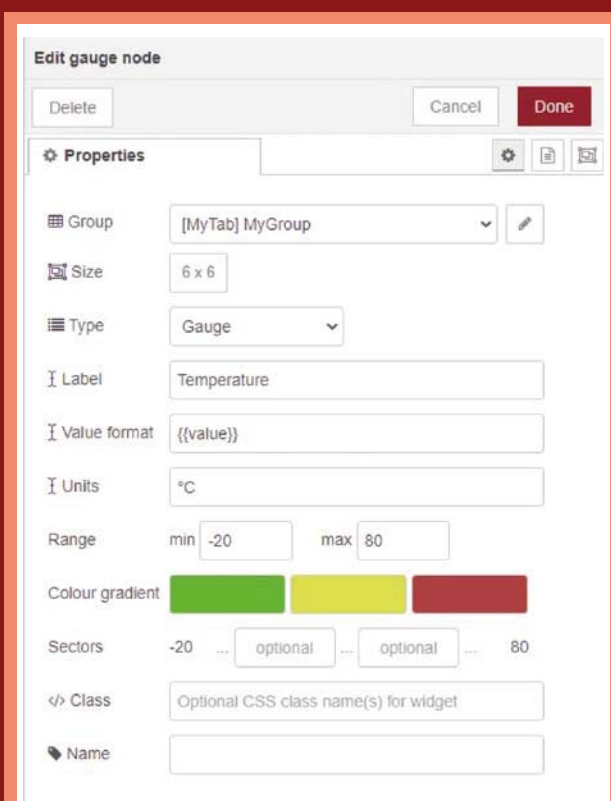


Fig. 13 - Configurazione del gauge Temperature.

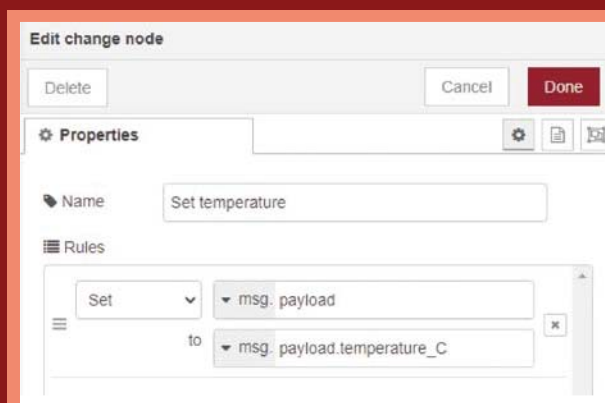


Fig. 14 - Configurazione del nodo Change che estrae il dato di temperatura.

- Range: 500 – 1500
- Units: hPa
- Gauge 4:
 - Nome: Heat Index
 - Range: -20 – 80
 - Units: °C

In Fig.13 è rappresentata la configurazione del gauge che fornisce l'indicazione relativa alla temperatura.

A questo punto configuriamo i nodi change. Lo scopo di questi nodi è quello di estrarre un campo specifico dal messaggio contenente l'object complessivo e passarlo al nodo function che lo formatterà correttamente per la visualizzazione (o direttamente al nodo gauge nel caso del dato di temperatura).

Per farlo ci serviamo della regola set e impostiamo nel campo "to" il valore `msg.payload.<campo specifico>`. Dove il valore <campo specifico> individua il campo che vogliamo estrarre dal messaggio. In Fig.14 è rappresentata la configurazione del nodo change che estrae il dato di temperatura.

A questo punto non ci rimane che configurare i 3 nodi function. Questi nodi hanno la funzione di formattare cor-

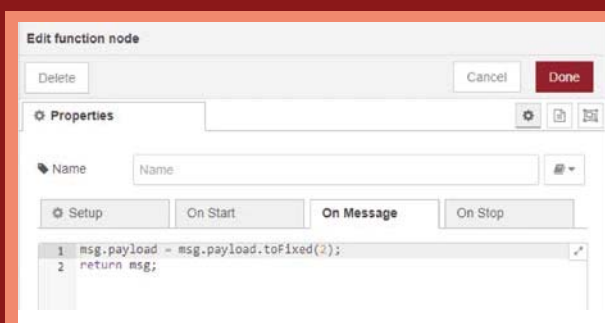


Fig. 15 - Uno dei nodi function usati per correggere la precisione.

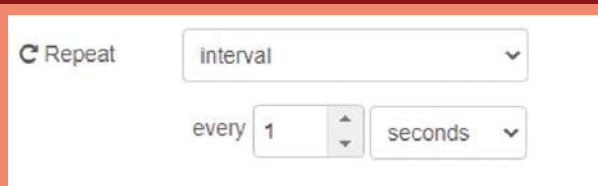


Fig. 16 - Configurazione del nodo inject.

rettamente i dati di alcuni campi (umidità, pressione, heat index), che hanno una precisione eccessiva (svariate cifre decimali) per la tipologia di dato da visualizzare. Tramite i nodi *funcion*, servendoci del metodo *toFixed()* portiamo la precisione a 2 cifre decimali. Lo snippet di codice Javascript che esegue questa operazione è riportato in Fig.15 ed è lo stesso per tutti i 3 i dati da manipolare (la temperatura viene già riportata dal nodo BME280 con una precisione di 2 cifre decimali). Ora non ci resta che configurare anche il nodo *inject* per fornire la periodicità. Come abbiamo visto anche nelle puntate precedenti, è sufficiente settare l'opzione *interval* ed indicare 1 secondo come periodicità (Fig.16). A questo punto la configurazione è conclusa, e possiamo collegare i nodi del nostro flow come riportato in Fig.17. Ora possiamo eseguire il deploy del nostro flow ed effettuare i collegamenti tra la nostra Raspberry Pi ed il sensore. La breakout BME280 ha 4 pin: Alimentazione,

GND, SDA ed SCL. Colleghiamo il pin di alimentazione ad una delle uscite a 5V della Raspberry Pi e il GND ad uno dei GND. I pin SDA ed SCL invece vanno collegati ai rispettivi pin della porta I2C1 della Raspberry Pi (pin 3 e 5 del connettore GPIO). A questo punto, se tutto è stato eseguito correttamente, collegandoci alla pagina:

<Node-RED server URL/IP>:1880/ui

Dove <Node-RED server URL/IP> rappresenta l'indirizzo della nostra raspberry, dovremmo visualizzare la dashboard rappresentata in Fig.18.

CONCLUSIONI

Siamo arrivati alla conclusione di questa quarta puntata in cui abbiamo visto come gestire la security e il componente dashboard in Node-RED. Le opzioni di security ci permettono di proteggere adeguatamente il nostro editor e le nostre interfacce grafiche qualora decidessimo di esporle sulla rete, mentre la dashboard ci permette di creare in maniera estremamente semplice interfacce grafiche hostate dal server, che quindi possono essere raggiungibili tramite diversi dispositivi, come PC, tablet e smartphone, ma è anche possibile visualizzarle su un dispositivo di output locale, come un monitor o un display, e creare quindi interfacce uomo-macchina molto evolute. □

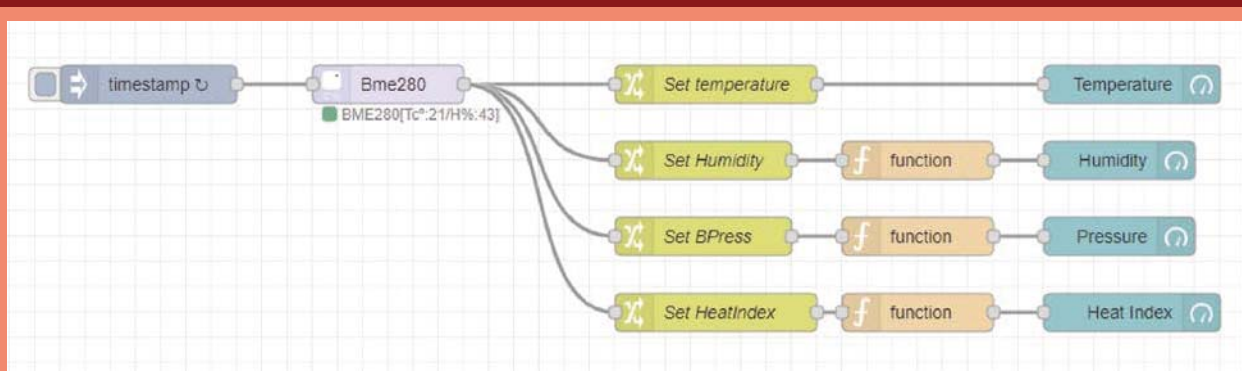


Fig. 17 - Flow con i collegamenti completi.



Fig. 18 - Dashboard della stazione meteo.

La sonda Solar Parker Probe ha toccato il Sole

Sembra quasi impossibile, visto che la nostra stella ha una temperatura stimata che farebbe fondere qualsiasi cosa la tocchi, ma è così: per la prima volta nella storia, un veicolo spaziale ha toccato il sole; si tratta della sonda Parker Solar Probe della NASA, che, lanciata nel 2018, ha sorvolato la parte corona (la zona più esterna) dell'atmosfera del sole, campionando particelle e campi magnetici. Questa storica missione aiuterà a raccogliere nuove informazioni sulla nostra stella e sulla sua influenza sul sistema solare. Già nel 2019, aveva scoperto che le strutture magnetiche a zig-zag nel vento solare, chiamate tornanti, sono abbondanti vicino al Sole, ma non era chiaro come si formassero; ora la sonda si è avvicinata abbastanza da identificare il luogo da cui hanno origine, ossia la superficie solare.

Il sole non ha una superficie solida ma un'atmosfera surriscaldata, fatta di materiale solare legato dalla gravità e dalle forze magnetiche; quando il calore e la pressione in aumento lo spingono lontano dal sole, esso raggiunge un punto in cui la gravità e i campi magnetici non riescono a trattenerlo. Quel punto, noto come superficie critica di Alfvén, segna la fine dell'atmosfera solare e l'inizio del vento solare, composto da quel materiale solare dotato dell'energia per attraversare quel confine, che trascina con sé il campo magnetico del Sole mentre corre attraverso il sistema solare, verso la Terra e oltre. Oltre la superficie critica di Alfvén, il vento solare si muove tanto velocemente che le onde al proprio interno vento non riescono a tornare al sole e se ne distaccano.

Prima dell'approccio di Parker Solar Probe non si sapeva dove si trovasse esattamente la superficie critica di Alfvén, collocata approssimativamente tra 10 e 20 raggi solari dalla superficie del sole (da 4,3 a 8,6 milioni di miglia). La traiettoria a spirale compiuta dalla sonda l'ha portata a una distanza inferiore ai 20 raggi solari (91% della distanza della Terra dal Sole) mettendola in grado di attraversare il confine della superficie di Alfvén; infatti all'ottavo giro della traiettoria, la Parker Solar Probe ha incontrato le condizioni magnetiche e particellari che caratterizzano tale zona, a 18,8 raggi solari (circa 8,1 milioni di miglia) sopra la superficie solare e tale condizione ha permesso agli scienziati di comprendere che la sonda ha attraversato la superficie critica di Alfvén. Di recente, la sonda è entrata nell'atmosfera solare.

www.nasa.gov



Lanciato il telescopio spaziale James Webb

Il giorno di Natale alle 9:20 ora locale a Kourou, un vettore Ariane 5 è decollato dallo spaziorpporto europeo al Guyana Space Center, per portare nella sua orbita di trasferimento il telescopio James Webb sviluppato sotto la guida dalla NASA in collaborazione con l'ESA e l'Agenzia spaziale canadese (CSA). 27 minuti dopo il decollo, il telescopio si è separato con successo dal razzo vettore, per arrivare alla sua destinazione finale, a 1,5 milioni di km dalla Terra, dopo aver percorso un viaggio lungo circa un

mese. Tre giorni dopo il lancio e la separazione dal vettore, James Webb ha aperto il proprio scudo termico e l'undicesimo giorno lo specchio secondario ha iniziato il suo posizionamento, succeduto dall'apertura e composizione dello specchio primario di 6,5 metri di diametro composto da 18 esagoni, creato per catturare quanta più luce possibile. Per sviluppare questo telescopio, sono state coinvolte tre agenzie spaziali: NASA, ESA e CSA. L'Europa ha giocato un ruolo importante nel



quadro di questa missione: l'ESA ha fornito il sistema di lancio a bordo di Ariane 5, così come lo spettrometro Nirspec, realizzato da Airbus, mentre il dipartimento d'astrofisica del CEA a Saclay e l'osservatorio di Parigi hanno concepito la camera Miri. È il telescopio spaziale più ambizioso mai inviato nello spazio. La missione del telescopio James Webb permetterà all'umanità di approfondire molto la conoscenza dell'universo. Questa missione ha richiesto 20 anni di preparazione con la NASA. Il lancio è un regalo di Natale fatto dalla comunità scientifica internazionale all'umanità, perché il telescopio James Webb, progettato per succedere al telescopio spaziale Hubble, il telescopio spaziale Webb (100 volte più potente) utilizzerà tecnologie migliorate e diverse per catturare il 70% di luce in più.

Grazie a questi miglioramenti, gli astronomi potranno compiere osservazioni mai fatte finora delle prime stelle e galassie formatesi dopo il Big Bang. Questo nuovo successo è senza dubbio uno dei lanci più emblematici di Ariane 5 per l'esplorazione dell'universo dopo Rosetta e i cargo spaziali europei ATV e il razzo Ariane 5 si è dimostrato il lanciatore più affidabile del mondo.

Per accompagnare il lancio è stato progettato un adattatore dedicato e fabbricato per ospitare il satellite in conformità con il volume disponibile sotto la carenatura: il telescopio, infatti, misura 10,5 metri di altezza e 14 metri di larghezza una volta dispiegato.

www.esa.int

Il robot bambino impara a volare

Ci sono voluti anni, ma ora sembra che per iRonCub, il bambino robotico nato dal genio dell'Istituto Italiano di tecnologia (IIT), sia arrivato il momento di spiccare il volo; e non si tratta di una metafora per dire che il progetto è maturo. Infatti il sogno, se mai questo bambino robotico sappia sognare, di volare sta per diventare realtà: volare come Iron Man, l'eroe dei fumetti, ma non per combattere qualche fantomatico nemico di celluloidi o della carta stampata, bensì per sorvolare le aree dove si è verificato un disastro naturale o meno, e mandare a terra immagini e informazioni utili alle squadre e alle unità di soccorso.

iRonCub ha già avuto modo di sorprenderci con le sue abilità; ciò che lo riporta in attualità è il fatto che all'IIT di Genova hanno deciso di provare a farlo volare, dotandolo di quattro piccoli turbogetti tipo quelli per aeromodellismo dinamico, la cui spinta verrà coordinata dall'elettronica per far raggiungere velocità e assetto richiesti dal volo. I robot volanti serviranno non solo nell'assistenza negli scenari dei disastri, ma anche a perfezionare le tecnologie di manipolazione aerea, attualmente esemplificata da quadricotteri dotati di un braccio robotico, che non possono muoversi per mezzo delle forze di contatto con l'ambiente e spesso lottano con il volo in ambienti ventosi mentre manipolano un oggetto, richiedendo un controllo preciso della posizione per svolgere compiti di manipolazione. Quindi la mano in più di un robot umanoide volante potrebbe stabilire un punto di contatto tra il robot e l'ambiente, rendendo così il controllo della posizione del robot più semplice e robusto. Inoltre la robotica umanoide aerea può essere utilizzata come banco di prova per gli esoscheletri volanti per esseri umani, portati alla ribalta da Richard Browning.

Far volare il robottino con dei motori a reazione ha richiesto coperture ignifughe per le gambe e piedi, che di fatto sono degli speciali pantaloni a prova di fuoco, perché se è vero che iRonCub è un robot, è pur vero che come tutti i bambini deve essere protetto...

<https://ami.iit.it/>



Salpa la prima nave a idrogeno liquido

Suiso Frontier è la prima nave porta idrogeno liquido al mondo, del consorzio Hydrogen Energy Supply-chain Technology Research Association (HySTRA). Trasporta gas liquefatto a una temperatura di -253°C , alla quale l'idrogeno occupa 1/800 del volume che occupa in forma gassosa. La capienza della nave, che ha una stazza di 8.000 t, è assicurata da una cisterna da 1.250 m^3 . Il gas liquefatto viene prodotto da AGL Energy, gestore di una miniera di lignite, che produce circa 70 kg al giorno di idrogeno blu, cioè ottenuto dalla combustione di fonti fossili (la lignite) con recupero della CO_2 , che non viene rilasciata in atmosfera, ma



catturata e stoccata sottoterra. La scelta è caduta sulla lignite per via dei costi, perché usare questo tipo di carbone costa tre volte meno di utilizzare elettrolizzatori per produrre idrogeno verde dall'acqua. Il con-

sorzio HySTRA mira a dimostrare la fattibilità del trasporto sicuro di idrogeno liquefatto in acque internazionali. Per il trasporto sicuro di volumi di massa di idrogeno, la nave è stata sottoposta a test Class NK in linea con le regole Guidelines for Liquefied Hydrogen Carriers, che descrivono i requisiti di sicurezza basati sulle Raccomandazioni intermedie dell'International Maritime Organization.

www.hystra.or.jp/en

4.200 km di cavo portano il fotovoltaico dall'Australia a Singapore

Entro il 2026 Singapore, nel sud-est asiatico, riceverà il 20% un quinto del suo fabbisogno di energia elettrica da un parco fotovoltaico australiano, nell'ambito del progetto "Australia-Asia PowerLink" che coinvolge la Sun Cable, azienda che poserà un cavo sottomarino di 4.200 km, il più lungo al mondo, tra la città portuale di Darwin e Singapore, nel nord dell'Australia, attraverso il quale scorrerà corrente continua ad altissima tensione. Si tratterà di una linea di trasmissione ad altissima tensione in corrente continua (HVDC) ottenuta dalle stringhe di pannelli fotovoltaici. A destinazione, l'elettricità verrà riconvertita e adattata alla rete locale.

Il parco fotovoltaico sorgerà su un'area di 120 km^2 a sud di Darwin e sarà dotata del sistema di accumulo più grande del mondo per garantire la continuità della potenza; l'elettricità viene trasportata a Darwin tramite un cavo HVDC lungo 800

km e parte rifornirà la città ed il resto, ben 3,2 GW, sarà poi inviato a Singapore e immesso nella rete. Anche nella stazione d'arrivo ci sarà una batteria come buffer. Il parco avrà una potenza di picco da 17 GW a 20 GW, la batteria associata con una capacità da 36 GWh a 42 GWh fungerà da tampone, garantendo che l'elettricità sia presente continuamente nel cavo HVDC anche quando il cielo è nuvoloso (una rarità in questa posizione) ed anche di notte. I moduli fotovoltaici in silicio utilizzati sfruttano una tecnica speciale chiamata Maverick, sviluppata dalla società australiana 5B e che vengono preassemblati a blocchi di 16 pannelli cablati in fabbrica, dotati di cerniere per poter essere ripiegati per il trasporto. A destinazione, vengono smontati e montati su basi preparate.

<https://suncable.sg/australia-asia-power-link/>



L'isola artificiale esporta elettricità verde

Continua il percorso di sviluppo delle rinnovabili che contraddistingue da anni la Danimarca, la quale punta a costruire due titanici parchi eolici offshore. L'elettricità verrà raccolta a Bornholm e su un'isola artificiale nel Mare del Nord (delle dimensioni di $445 \times 270\text{ m}$) che ospiterà l'infrastruttura per raccogliere l'elettricità, convertirla in trasformatori, raddrizzatori e inverter, inoltrarla a terra, ma anche personale di manutenzione ed un piccolo porto. Non è ancora stato deciso se sull'isola saranno installati anche sistemi di accumulo di energia elettrica, per garantire la continuità di fornitura. Il progetto, che sarà completato nel 2030, comporterà un investimento di circa 30 miliardi di euro. Allo scopo sono già stati firmati i contratti tra l'operatore danese del sistema di trasmissione (TSO) Energinet e il TSO tedesco 50Hertz, nonché con il Belgio.

La parte più complessa del progetto sono le isole, che saranno costruite a circa 80 km dalla costa, come piattaforme galleggianti o come isole artificiali. L'infrastruttura conterà di due parchi eolici offshore; grandi cavi elettrici sottomarini trasporteranno l'energia sotto forma di corrente trifase o corrente continua ad alta tensione alle isole e da qui arriverà alle utenze. La Danimarca ha già impianti eolici per una potenza di quasi 6 GW nel Mare del Nord e nel Mar Baltico, che coprono quasi il 60% del fabbisogno del Paese; il progetto punta a ulteriori 7 GW ottenibili dalle isole artificiali. Con esso la Danimarca intende coprire il 100% del proprio fabbisogno di elettricità da fonti rinnovabili ed esportare il surplus.

www.dtu.dk

**SONO ARRIVATI I NUOVI
SET DI COMPONENTI
CON ARDUINO UNO!**

LIVELLO 2



€ 42,90

Cod. **ARDUKITLEVEL2**

**SET con Arduino Uno Rev 3
e i componenti elettronici
per sviluppare applicazioni
con il linguaggio Arduino**

LIVELLO 1



€ 32,90

Cod. **ARDUKITLEVEL1**

Disponibili anche....

SET PER ARDUINO UNO REV3

€ 65,00

Cod. **ARDUKITV6**

**IL PIÙ
COMPLETO!**



Il Set comprende: il libro "l'ABC di ARDUINO", la board Arduino UNO Rev3, e tutti i componenti per effettuare esperimenti descritti nel libro.

**SET CON ARDUINO UNO REV 3
CON CAVO USB**

Contiene il necessario per utilizzare Arduino Uno Rev 3; per iniziare basta connettersi a un computer tramite il cavo USB o alimentarla con un trasformatore AC/DC o una batteria.



€ 25,50

Cod. **COMB06G**

Prezzi IVA inclusa

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

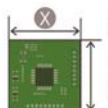
1 Layer

2 Layers

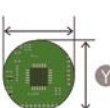
4 Layers

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.

www.futurashop.it

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

**FUTURA
 ELETTRONICA®**