

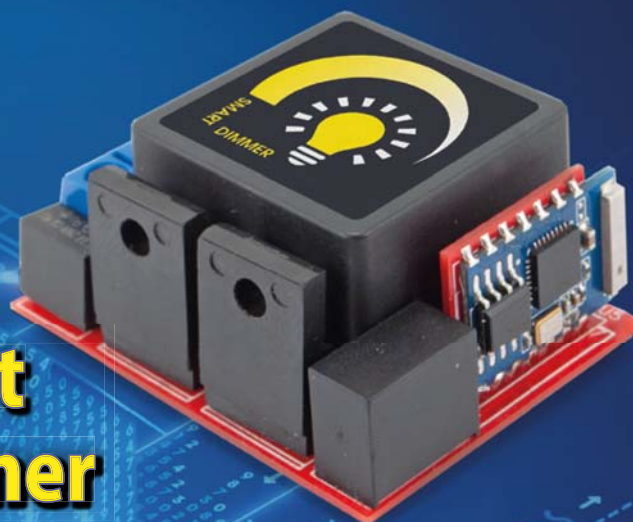
Elettronica In

www.elettronica.in.it

€ 6,00 - Anno XXVII - n. 259 - NOVEMBRE 2021



Smart Dimmer



■ **Oscilloscopio palmare DS0225: aggiorniamo il firmware**

■ **Monitoraggio temperatura corporea con identificazione**

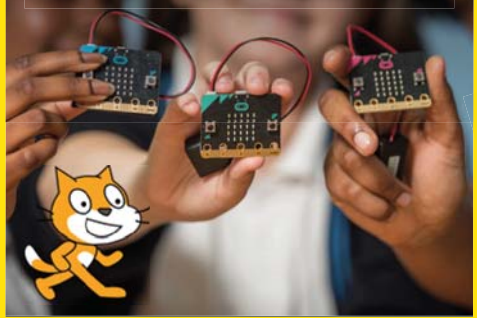
■ **Accendere il PC da remoto**

■ **Mini lettore MP3**

■ **Octoprint: la stampa con Raspberry Pi**



micro:bit la nuova scheda 2.0



Domotica con SCADA ed ESP8266



CONOSCERE
E USARE



Node-RED

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





L'elettronica Smart

"Hey Google, accendi la TV!", "Alexa metti un po' di musica!", quante volte abbiamo detto o sentito queste parole. In un mondo sempre più connesso, gli smart speaker ci permettono di interagire più facilmente con la tecnologia. Grazie all'integrazione di un'intelligenza artificiale possiamo condurre una conversazione in modo familiare e intuitivo per gestire la nostra casa intelligente, trasformandola nella nostra smart home.

Due persone su tre possiedono un prodotto per la smart home.

Si stima che tra i 25 e i 55 anni almeno due persone su tre possiedono un prodotto per la smart home e tra questi il più apprezzato è proprio lo smart speaker.

Negli ultimi mesi il mercato dei prodotti per la casa intelligente

ha ricevuto una spinta a causa della pandemia; il tempo passato nella propria abitazione è aumentato e questa è stata l'occasione per rinnovarla anche grazie all'impiego di dispositivi smart. Ma nonostante ciò, uno dei motivi che ne impedisce una diffusione capillare è il prezzo.

Come spesso accade in queste occasioni, la strada del DIY (Do It Yourself, come oggi chiamiamo il Fai Da Te per stare al passo coi tempi) è la soluzione per risparmiare e, perché no, anche imparare qualcosa. Noi lo sappiamo bene ed è perciò che in questo numero vi presentiamo uno Smart Dimmer per regolare l'intensità luminosa delle luci di casa, un comando (smart, ovviamente) per accendere a distanza il nostro PC e un sistema SCADA (Supervisory Control And Data Acquisition) nato per l'automazione industriale, ma che utilizziamo per implementare il comando e controllo via WiFi di un dispositivo IoT.

ON LINE

elettronica@elettronica.it

SI RINGRAZIA

Futura Elettronica
Mouser



In copertina:
Realizziamo uno smart dimmer per regolare la luminosità delle lampade con semplici comandi vocali e lo interfacciamo con Amazon Alexa e Google assistant.

Boris Landoni

boris.landoni@elettronica.it

Elettronica In

Rivista mensile, anno XXVII n. 259
NOVEMBRE 2021

EDITORIALE

Direttore Responsabile

Boris Landoni

boris.landoni@elettronica.it

Redazione

Stefano Garavaglia, Paolo Gaspari,
Davide Scullino, Gabriele Daghetta

redazione@elettronica.it

Grafica

Alessia Sfulcini

alessia.sfulcini@elettronica.it

Ufficio Pubblicità

Monica Premoli (0331-752668)

monica.premoli@elettronica.it

Ufficio Abbonamenti

Tel. 0331-752668

abbonati@elettronica.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.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.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.

© 2021 Futura Group srl

Futura
Group
Edizioni

Contenuti

01

RUBRICHE

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

Progetto di Copertina



79

Programmazione

Micro:bit la nuova scheda 2.0



19

19

ARTICOLI

PROGRAMMAZIONE

19 Micro:bit la nuova scheda 2.0

Nasce la nuova versione di micro:bit. Vediamo le differenze tra le varie versioni e con un esempio pratico verifichiamo flessibilità e potenza.

STRUMENTAZIONE

33 DSO225: aggiorniamo il firmware

Presentiamo un importante aggiornamento che amplia le capacità del nostro oscilloscopio DIY con le funzioni FFT, X/Y e la misura della potenza.

HEALTHCARE

40 Monitor temperatura con RFID

Implementiamo un sistema di monitoraggio della temperatura senza contatto e di identificazione RFID sfruttando il modulo uCHIP.

Strumentazione

DSO225: aggiorniamo il firmware



33

Healthcare**Monitor
temperatura
con RFID****40****Networking****Domotica con SCADA
ed ESP8266****95****DOMOTICA****57 Accendere il PC da remoto**

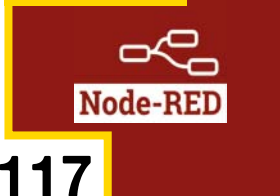
Accendiamo il nostro PC fisso da remoto o tramite automazioni sul nostro hub domotico.

AUDIO**67 MP3 con il modulo DFR0299**

Impariamo ad integrare un lettore MP3 nei nostri progetti, per riprodurre file audio vocali e musicali.

NETWORKING**79 Smart dimmer**

Regola l'intensità luminosa di lampade a incandescenza, CFL e a LED dimmerabili, controllato da smartphone, localmente o tramite assistente vocale.

Domotica**Accendere il PC
da remoto****57****Mondo 3D****Octoprint,
la stampa
smart****109****Didattica****Conoscere e usare
Node-RED****117****NETWORKING****95 Domotica con SCADA
ed ESP8266**

Realizziamo un dispositivo IoT capace di comandare carichi a 220V 10A, usando come interfaccia uno SCADA Open Source.

MONDO 3D**109 Octoprint, la stampa smart**

Permette di controllare le stampanti 3D anche da remoto rendendole a tutti gli effetti smart printer.

DIDATTICA**117 Conoscere e usare Node-RED**

Iniziamo a scoprire Node-RED, un tool di flow-based programming, molto orientato all'IoT ed alla connettività, originariamente sviluppato dall'IBM Emerging Technology Services team e adesso parte della JS Foundation. In questa puntata ci addentriamo maggiormente nell'uso del tool, esaminando i nodi di utilizzo più comune, vediamo come aggiungerne di nuovi e scopriamo cosa sono i context in Node-RED. Seconda puntata.

Migliaia di strumenti da centinaia di produttori fidati



MOUSER ELECTRONICS

La coppia dei servocomandi

Uno dei parametri più importanti dei servocomandi standard, siano essi analogici o digitali, è la coppia motrice, vale a dire la forza esercitabile dal dispositivo per una determinata lunghezza della leva ingranata dal perno o alberino che dir si voglia; tale coppia è (sempre) indicata dal costruttore tipicamente in kg/cm e va intesa come la forza massima esercitata all'estremità della leva, opposta a quella in cui è imperniata nell'alberino del servo. Questa forza è, intuitivamente leggendo l'unità di misura, tanto minore quanto maggiore diviene la lunghezza della leva, tant'è che un servo da 10 kg/cm esercita una forza di 10 kg radialmente a 1 cm dal perno, di 1 kg a 10 cm di distanza e da 100 g all'estremità di una leva di un metro di lunghezza. L'immagine in questa pagina chiarisce graficamente tale concetto.

Curare le ossa con il campo elettrico

Conosciamo da tempo le grandi potenzialità dell'elettronica nella cura di malattie e nell'ausilio alla medicina e alla chirurgia in generale; sappiamo anche delle proprietà dell'elettrostimolazione transcutanea (TENS) della ionoforesi (passaggio sottocutaneo di medicinali liquidi sotto l'effetto di un campo elettrico) e della magnetoterapia, la quale ha molta rilevanza negli stati infiammatori e nella riabilitazione post-chirurgica (specialmente in ortopedia) perché il campo elettromagnetico alternato impone ai vasi sanguigni una deformazione ritmica (contrazione alternata a rilassamento) stimolando la circolazione e la vascolarizzazione per riattivare l'azione osteoblastica e stimolare la produzione di collagene. Ma l'elettronica ci ha recentemente

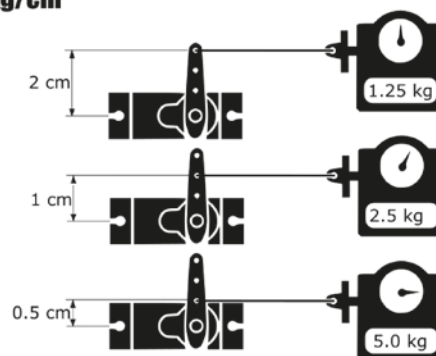
Tips & Tricks sono i nostri suggerimenti in pillole e i piccoli trucchi riguardanti sia l'hardware che il firmware, che sperimentatori, tecnici e progettisti elettronici vorrebbero conoscere per sviluppare meglio la propria passione: quella per l'Elettronica applicata. Ecco quelli che abbiamo preparato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

offerto una tecnica che si assimila alla magnetoterapia, differendo però per il minor peso e ingombro degli elementi irradiatori, per i consumi elettrici più ridotti, l'assenza degli inconvenienti dei campi elettromagnetici e soprattutto la possibilità di eseguire trattamenti prolungati durante il sonno o nell'attività quotidiana, in virtù del fatto che l'irradiazione avviene ad opera di due placchette facilmente indossabili. Si tratta della più recente soluzione tecnologica applicata all'attivazione

delle proprietà rigenerative dell'osso danneggiato da fratture o interventi chirurgici e si basa sulla stimolazione ad opera di un campo elettrico con specifiche proprietà.

Un tipico apparecchio del genere (Osteobit, per esempio, documentato su www.igeamedical.com/it) agisce esponendo la parte da trattare ad un campo elettrico alternato pulsato con burst di 60 kHz pulsati a 12,5 Hz da un'onda quadra modulante OOK con duty-cycle del 50%. Questa tecnica determina la polarizzazione alternativa

Servo da 2,5 kg/cm



↑ Ecco come varia la forza esercitata da un servocomando in funzione della lunghezza della sua leva considerata dal fulcro, ossia dall'asse dell'alberino.

delle cellule producendo una stimolazione simile a quella della magnetoterapia, ma agendo con un campo elettrico che agisce sulle cellule. In pratica lo stimolo determinato dal campo elettrico modifica la permeabilità della membrana cellulare nei confronti degli ioni di calcio, ovvero implica l'apertura dei canali di membrana e il conseguente aumento della concentrazione di calcio nel citoplasma stimola l'attivazione della calmodulina e conseguentemente la proliferazione cellulare dell'osso, accelerando il processo riparativo.

Le placche, altro non sono che le armature di un condensatore il cui dielettrico è la parte del corpo da trattare, la quale presenta una costante dielettrica variabile in base allo spessore, al tipo di tessuto e alle condizioni cliniche (l'apparecchio si adatta rilevando la quantità di carica trasferita tra le armature). La quantità di carica che attraversa questo dielettrico dipende dalla costante dielettrica e dalla tensione applicata alle placche o armature; la corrente passante è di conseguenza, ossia è legata alla quantità di carica elettrica trasferita. La corrente passante nella parte trattata e quindi attraverso le armature, ha intensità dell'ordine dei $15 \div 35 \mu\text{A}/\text{cm}^2$ e la tensione applicata è dell'ordine della decina di volt, quindi il trattamento è praticamente impercettibile e per nulla doloroso.

BEC: cos'è e a cosa serve

Nei sistemi alimentati a batteria si utilizza spesso un modulo BEC (Battery Eliminator Circuit): si tratta di un circuito elettronico utilizzato per escludere l'alimentazione dei carichi a maggior

assorbimento quando la tensione rilevata sulla batteria indica che quest'ultima è prossima alla scarica; lo scopo è sia proteggere batterie come quelle al litio dall'eccessiva scarica (se scendono sotto una tensione minima, le celle Li-ion e LiPo si danneggiano irreparabilmente e perdono parte della loro capacità) sia consentire il funzionamento di altri circuiti critici alimentati dalla stessa batteria. Il rilevamento dello stato di carica della batteria viene effettuato mediante un microcontrollore o un preciso comparatore e si basa sulla curva di scarica delle batterie.

Nel caso dei sistemi radiocomandati, il BEC permette di evitare la scarica completa della batteria, quindi di far funzionare a potenza ridotta i motori o escluderli per tenere in funzione altri circuiti. Ad esempio, negli aeromodelli ad unica batteria (per motori ed elettronica di controllo) se la batteria viene scaricata dai motori, viene a mancare l'alimentazione per i servocomandi che gestiscono le superfici di controllo dell'aeromodello, che così diventa ingovernabile e precipita. Invece se si sconnettono i motori e l'aereo è ben progettato ed equilibrato, è possibile portarlo a terra semplicemente facendolo planare.

Il BEC esiste in due versioni: quello per sistemi ad unica batteria funziona con la batteria e dispone di un'uscita per i motori (alimenta gli ESC) o attuatori di potenza ed una a bassa corrente per i circuiti sensori, radiocomando, servocomandi e in generale circuiti indispensabili; se rileva che la batteria è arrivata al limite tra carica e scarica, spegne l'uscita di potenza e lascia tensione sull'uscita dei circuiti "indispensabili". C'è poi un secondo tipo di BEC, fatto per sistemi due batterie: una per gli utilizzatori di potenza

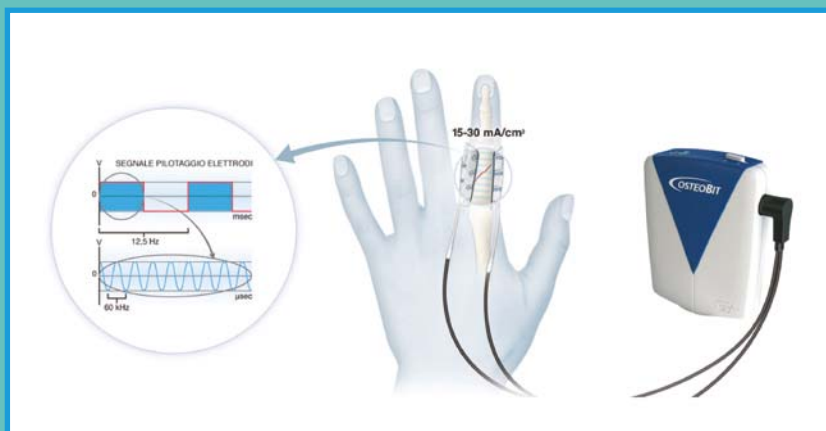
ed una per i circuiti da mantenere sempre accesi; questo secondo tipo di BEC viene alimentato dalla batteria a bassa potenza (quella per i circuiti indispensabili) e riceve anche la tensione della batteria per gli utilizzatori di potenza, solo che quest'ultima non alimenta l'elettronica ma viene semplicemente commutata mediante MOSFET dalla batteria all'uscita del BEC che va ai motori o carichi di potenza. Esistono poi i BEC integrati negli ESC, ossia nei controller dei motori elettrici nel senso che i moderni ESC sono spesso dotati di funzione BEC; in questo caso il controllo dell'uscita di potenza è più semplice perché avviene all'interno dell'ESC stesso.

Il BEC può anche essere di tipo standalone e limitarsi a sconnettere l'alimentazione fornita dalla batteria quando la tensione della stessa scende oltre il minimo ammissibile per evitare danni alle celle: un tipico esempio è il circuito presente in alcuni pacchi batteria di computer portatili ed altri apparati simili dove, quando le celle sono eccessivamente scariche la batteria non eroga più corrente e in apparenza è interrotta.

I più moderni e prestanti BEC per il controllo dell'alimentazione possono essere impostati per operare con due distinte modalità di intervento, che sono:

- Soft; il BEC riduce gradualmente la massima tensione erogabile ai carichi ad alto assorbimento man-mano che la batteria va scaricandosi;
- Hard; quando la batteria scende sotto il minimo livello di tensione previsto, il BEC sconnette improvvisamente i carichi a maggior assorbimento.

Altra impostazione dei BEC è la soglia di tensione della batteria cui staccare; in molti casi viene impostata automaticamente dal BEC perché riconosce il tipo di batteria e la quantità di celle (2s, 3s ecc.). Nel caso dei radiomodelli e dei dispositivi radiocomandati, il BEC più semplice, che blocca solo l'alimentazione dei motori, si applica a monte dell'alimentazione di questi ultimi. Il BEC doppio, si collega con l'uscita per l'ESC all'alimentazione di quest'ultimo e con quella per il ricevitore del radiocomando all'alimentazione di quest'ultimo. Quanto al BEC che controlla solo l'alimentazione del radiocomando, può essere usato per spegnere completamente il modello e quindi togliere potenza all'ESC anche senza intervenire sull'alimentazione di quest'ultimo: basta collegarne l'uscita ad una presa qualsiasi del ricevitore (tanto sul connettore di questo tipo di BEC si trovano i due soli fili dell'alimentazione, mentre il comando



↑ **Funzionamento di un tipico apparato per fisioterapia basata sul campo elettrico: si tratta dell'OsteoBit.**

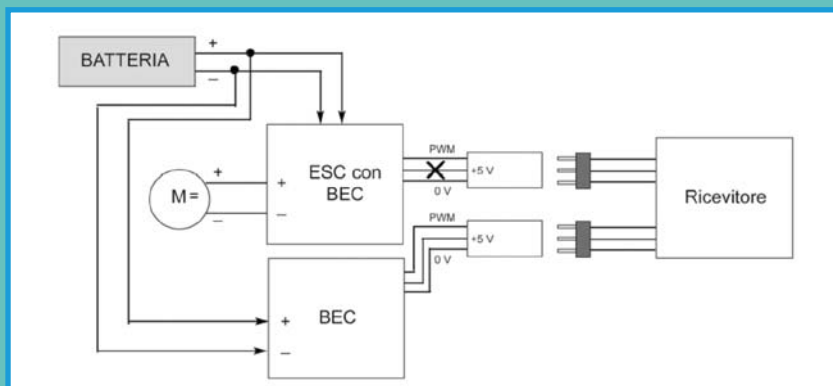
degli impulsi non è cablato) e tagliare il filo del positivo a 5 V del cavetto in arrivo dall'ESC. Tale manovra serve perché normalmente l'ESC fornisce la tensione stabilizzata di 5 volt al ricevitore del radiocomando e in tal caso la tensione fornita dal BEC sarebbe in parallelo con quella fornita dall'ESC: ciò porterebbe quasi sicuramente al danneggiamento di uno o dell'altro, dato che mai i due circuiti forniscono una tensione di valore uguale. Recentemente sono apparsi sul mercato degli accessori da radiomodellismo gli Ultra BEC o Ultimate BEC (U-BEC) i quali altro sono dei BEC stand-alone capaci di erogare più corrente, dotati di funzioni in più. Il BEC è un circuito che si può trovare anche nella trazione elettrica e quindi nei carrelli elevatori, tipicamente in quelli più moderni dotati di batteria al litio, oltre che nei MBS (Battery Management System) delle auto elettriche; in questo si tratta di un sistema integrato che funziona in modo "soft" e che inizia a scalare la potenza destinata al motore (o ai motori) di trazione man-mano che la carica va diminuendo, segnalando inoltre nel quadro strumenti affinché il conducente sappia adeguare la propria condotta di guida.

DMD: dai videoproiettori all'automotive

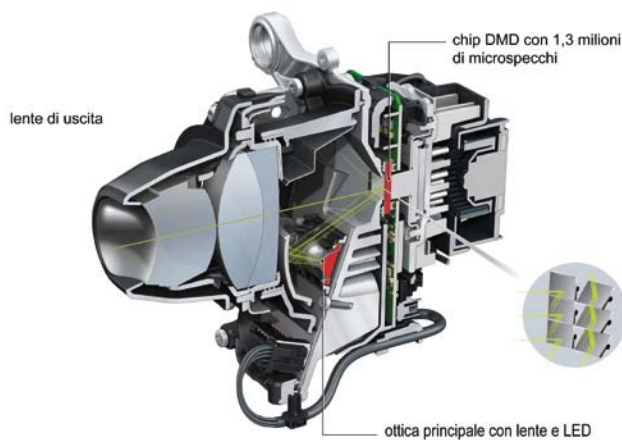
Nelle moderne automobili, in special modo in quelle elettriche, nelle quali il contenimento del consumo di elettricità è determinante per incrementare l'autonomia di marcia, le lampadine tradizionali, a filamento, sono ormai bandite, in special modo quelle dei proiettori

anabbaglianti, abbaglianti e fendinebbia anteriori, che sono quelle più energivore. Nuove tecnologie come quella dei LED ed ancor più del laser a luce bianca, le hanno sostituite. Un'altra tecnologia che ha fatto capolino nel mondo dell'automobile di recente è quella basata sui microspecchi e chiamata DMD (Digital Micromirror Device); in questo caso la fonte primaria di luce è un LED ad alta efficienza e luminosità, la cui luce viene riflessa verso una matrice di microspecchi azionati da dispositivi MEMS (Micro Electro Mechanical System) ossia da attuatori elettromeccanici integrati su chip di semiconduttore e pilotati a matrice, in modo da orientare ogni singolo microspecchio come desiderato. Tale soluzione nacque alcuni anni fa per i videoproiettori DLP da Texas Instruments, come alternativa ai classici LCD trasmissivi, per fornire immagini più brillanti, contrastate e luminose: le tre luci elementari (R, G, B) raggiungevano

la matrice di microspecchi i quali costituivano i pixel dell'immagine e venivano opportunamente orientati per dirigere le luci a comporre l'immagine. Nel settore automobilistico la tecnica è stata adottata nei recentissimi gruppi ottici delle vetture del gruppo VAG e permette, partendo da un fascio di luce fisso, di riflettere il flusso luminoso in vari modi, a seconda dell'impostazione che il chip di controllo riceve dalla centralina di comando, consentendo di ottenere varie geometrie di proiezione, che vanno dalla asimmetrica dell'anabbagliante alla simmetrica dell'abbagliante. Si realizza, insomma, un gruppo ottico capace di funzionare in vari modi e sostituire più luci; peraltro la matrice a microspecchi (in questo caso da 1,3 milioni di elementi) consente di allargare sui lati il fascio luminoso o concentrarlo in profondità in base alla richiesta del conducente, implementando in un solo dispositivo le luci di curva e quelle adattive AFS. L'interfacciamento con tale dispositivo avviene via bus, CAN o LIN.



↑ Collegamento tipico di un BEC per radiomodelli in un'elettronica dove l'ESC ha la funzione BEC integrata; il taglio del filo +5V del connettore che va dall'ESC al radiocomando serve per evitare il cortocircuito con i 5V forniti dall'ESC e dal BEC.



↑ Struttura del proiettore DMD dell'Audi e-tron Sportback 55 quattro.

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

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

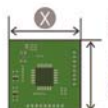
1 Layer

2 Layers

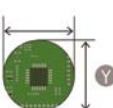
4 Layers

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.

IoT e AI stanno abilitando applicazioni che consentono la gestione contemporanea di tutti gli elettrodomestici di casa; per questo Haier, il colosso cinese degli elettrodomestici, ha realizzato l'app "hOn" che, collegata ad una piattaforma IoT open, elabora le immagini del bucato, o quelle dell'etichetta del vino, o quelle del cibo in forno, restituendo informazioni sul lavaggio ideale o sul tipo di bevanda, o impostando il ciclo di cottura più adeguato.

La piattaforma IoT di Haier si chiama CosmoPlat e, oltre a fare da tramite fra i dati raccolti e le immagini riprese dallo smartphone per impostare l'elettrodomestico da app, permette anche di sapere se una lavatrice sta lavorando bene o se c'è bisogno di attivare l'assistenza.

La piattaforma IoT Cloud-based

› Elettrodomestici connessi via app



di Haier di recente è stata integrata con AI e 5G; è in grado di dialogare anche con elettrodomestici di terze parti. La parte AI è costituita da software capaci di percepire l'ambiente che li circonda, e formulare decisioni basate sui dati per raggiungere

un certo obiettivo, ossia fare quel che l'uomo farebbe: impostare la lavatrice in base al tipo di capi da lavare, per esempio.

L'app hOn è quindi un assistente virtuale in grado di guidare gli utenti al migliore utilizzo dei loro elettrodomestici connessi. Il set di funzioni più interessante è quello offerto dalla Snap&Do experience, che grazie all'immagine recognition renderà facile lavare i capi d'abbigliamento e le stoviglie, (Snap&Wash), asciugare i vestiti (Snap&Dry), conservare i cibi (Snap&Cool), cucinare (Snap&Cook) e scegliere il vino giusto (Snap&Wine). hOn è stata sviluppata come evoluzione dell'app simply-Fi della Candy (anch'essa nelle mani di Haier) ma è più flessibile e consente di sostituire i comandi dell'elettrodomestico.

www.haier.com

› Il robot ispeziona disinfetta i locali con l'UV

Omron e UV Systems UK hanno prodotto un sistema di sterilizzazione robotizzato capace di muoversi autonomamente negli ambienti commerciali e lavorativi, sterilizzandoli dal virus del COVID-19 senza danni alle merci. Il robot UV Systems UV-C elimina il 99% del virus nell'intero ambiente con una tracciabilità completa. Il robot garantisce l'eliminazione di virus, agenti patogeni e batteri da tutti i punti di contatto all'interno di una struttura; il robot UV-C si è dimostrato un metodo sicuro per combattere il Covid e altri virus come SARS, MERS ed Ebola. La soluzione riduce inoltre la necessità di potenti sostanze chimiche che presentano rischi per la salute e la sicurezza dell'operatore.

Composto da un array di lampade UV-C di precisione sviluppate da Philips e montate su una piattaforma robot mobile OMRON all'avanguardia, il robot UV-C si sposta in modo autonomo grazie ai suoi sensori che gli consentono di rilevare gli ostacoli



lungo il percorso ed evitare collisioni, ma anche di spegnersi se una persona si trova a meno di 8 metri. Il robot, alimentato a batterie, ha un'autonomia fino a 4 ore ed è quindi capace di trattare grandi aree in modo efficiente, affidabile e rapido, anche ritornando alla stazione di ricarica quando è in "riserva". Inoltre, grazie al WiFi garan-

tisce la tracciabilità completa dell'area trattata, monitora e registra il proprio percorso per garantire la completa disinfezione; se non riesce a pulire un'area a causa di un ostacolo, ad esempio, segnala le informazioni in modo che il problema possa essere risolto in altra maniera.

industrial.omron.it

› CPU più sicure per l'Edge-IoT

Sondrel ha creato una potente piattaforma IP quad core, la SFA 200, ideale per le soluzioni ASIC per la raccolta e l'elaborazione remota di video e dati sull'edge con trasmissione sicura dei risultati. Gli ASIC a canale singolo risultanti possono essere uniti a formare soluzioni di elaborazione scalabili e funzionalità aggiuntive possono essere aggiunte in modo modulare. Le applicazioni includono smart metering, smart home, smart factory, dispositivi a comando vocale e infotainment. Ogni ASIC si basa su un cluster CPU per attività generiche e un cluster AI per attività di rete neurale con supporto di memoria e interfacce. Il cluster è normalmente formato da 4 CPU Arm A53 o A55 e fornisce una capacità di

elaborazione di 9-10 GMIPS a 1 GHz di clock. Il cluster Neural Net si basa su core Arm Ethos o core DSP-AI che forniscono prestazioni nominali 4 TOPS.

Il design del chip ha un tessuto di sistema componibile Network on Chip, percorso dati multi-path, multi-width (da 64b a 512b) a 800MHz nominali; prevede l'ottimizzazione delle prestazioni del flusso di dati per l'applicazione, l'arbitrato QoS e il supporto per la pianificazione. Il sottosistema di gestione del sistema (SMS) gestisce la configurazione, l'avvio (Boot) e le modalità di funzionamento del SoC insieme alla gestione attiva dell'alimentazione per applicazioni con alimentazione a bassa tensione o a batteria.

› IoT e Advanced Analytics per la siderurgia

Abilitare lo sviluppo di prodotti e servizi dedicati alla gestione degli asset, al controllo dei consumi energetici e alla gestione ottimizzata del processo e della qualità nell'industria dell'acciaio, grazie alla raccolta e all'elaborazione dei dati: sono questi gli obiettivi cui beanTech e Danieli Automation puntano con la loro piattaforma Q3-Platform. L'infrastruttura software permette la raccolta dei dati dal campo o da sistemi aziendali esistenti, con l'obiettivo di arricchirli, elaborarli, studiarli, al fine di migliorare i processi

produttivi e di business. Q3-Platform è uno strumento abilitante lo sviluppo di prodotti e servizi dedicati alla gestione degli asset, al controllo dei consumi energetici e alla gestione ottimizzata del processo e della qualità. Il settore siderurgico implica un processo complesso che genera una mole di dati spesso elaborati solo a produzione completata, impedendo l'applicazione di modelli predittivi in tempo reale che permettono, invece, di ottimizzare la vita dell'impianto, riducendo i costi e innalzando la qualità

e il livello di servizio. Garantire una elaborazione dei dati in tempo reale, monitorare il consumo energetico dell'impianto siderurgico e ottimizzare i carichi al fine di migliorare la produzione nel settore dell'acciaieria, sono gli obiettivi che Q3-Platform può aiutare a raggiungere, perché è uno strumento di Industrial IoT e Advanced Analytics verticale per il settore siderurgico, applicato in 100 impianti siderurgici nel mondo.

La Q3-Platform permette anche di centralizzare in un solo punto il governo degli asset, unificando la visione d'insieme e consente di gestire in modo automatico la raccolta, l'analisi e la sintesi dei dati (Data Analysis), tramite un portale web unificato. Inoltre, consente ai middle manager, nel settore della manutenzione, qualità ed energia, di avere un tool specifico che semplifica il reperimento delle informazioni, offrendo un unico endpoint da cui estrarre i dati per le analisi, senza ricorrere a software ad hoc.

La Q3-Platform si integra in infrastrutture IT/OT esistenti e con i sistemi meno recenti, permettendo la digitalizzazione degli impianti senza dover sostituire il parco macchine.

www.beantech.it



Il sottosistema di sicurezza integrale utilizza un Arm A53 o M33 a seconda dell'applicazione, per fornire monitoraggio di attività/intrusioni, firma software, crittografia e autenticazione per resistere all'hacking, ulteriormente potenziato dal monitoraggio del comportamento del sistema di guardia. Il design della piattaforma SFA 200 richiede solo l'IP del cliente o l'IP standard specifico per l'applicazione per creare l'ASIC finale, customizzato per soddisfare le esigenze del cliente, fornendo così un basso rischio ed un time-to-market più rapido che può ridurre i costi di progettazione fino al 30%. Sondrel offre anche un servizio di creazione di chip testato sulla base del progetto del cliente.

Il sistema SFA 200 fa parte della famiglia di piattaforme "IP Architecting the future" di Sondrel.

www.sondrel.com



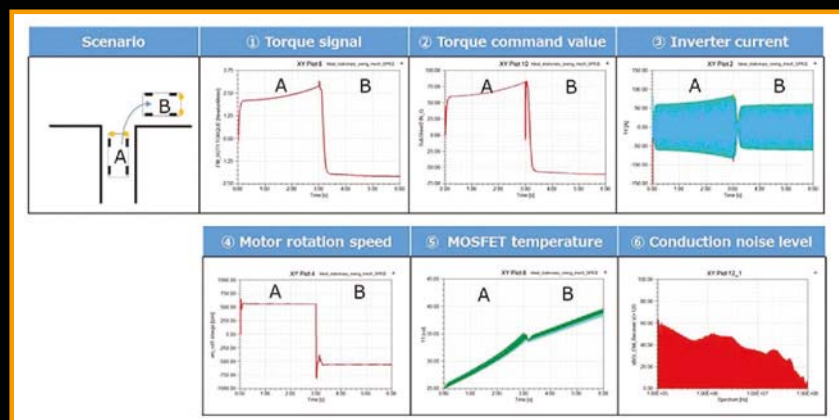
› Simulazione Model-Based veloce di semiconduttori

Toshiba ha sviluppato una tecnologia di simulazione di sviluppo model-based (MBD) che riduce del 90% i tempi di verifica per i semiconduttori impiegati nell'industria automobilistica; ciò permetterà agli sviluppatori di elettroniche automotive di accelerare la progettazione utilizzando i semiconduttori Toshiba. Questo è importante perché i circuiti elettronici per le auto diventano sempre più complessi, anche in virtù della diffusione dei sistemi avanzati di assistenza alla guida (ADAS). Lo sviluppo basato su modelli, una metodologia di simulazione software che aiuta a valutare le prestazioni in tempo reale, permette di perfezionare processi di progettazione complessi e di farlo in tempi ridotti. La MBD separa le funzioni in blocchi e verifica il comportamento complessivo del veicolo collegando ciascun blocco. Sono necessari modelli di simulazione dettagliati che includano il comportamento dei semiconduttori nei singoli blocchi per verificare il calore e l'interferenza elettromagnetica (EMI), parametri essenziali per valutare le prestazioni delle apparecchiature automobilistiche. Tuttavia, man mano che i modelli diventano più dettagliati e precisi, i tempi di verifica si allungano.

L'attuale MDB Toshiba simula contemporaneamente i circuiti elettronici e i componenti meccanici, implicando un gran numero di calcoli inutili e dispendiosi; è complicata perché adotta il modello SPICE che definisce oltre 100 parametri per la simulazione dei semiconduttori. Invece la nuova tecnologia Accu-ROM Toshiba, calcola separatamente i circuiti elettronici e i componenti meccanici. Prima verifica i componenti meccanici, poi semplifica il modello per i componenti elettronici e infine verifica l'intero sistema, compresi i circuiti elettrici. Questo approccio elimina i calcoli non necessari. Nella

valutazione dei circuiti elettrici si genera un modello di circuito integrato ad altissima velocità Hardware Description Language-Analog Mixed Signal (VHDL-AMS) dal modello SPICE, che limita l'intervallo di verifica a parametri essenziali come il calore e il rumore EMI, riducendo i tempi. Ad esempio, la verifica di un sistema di servosterzo passa da 32 ore a 3 ore e mezza. Toshiba utilizzerà la tecnologia per promuovere lo sviluppo di semiconduttori automotive e per altre applicazioni, come apparecchiature industriali ed elettrodomestici.

www.toshiba.com



Strumenti di misura per veri professionisti!



Rilevatore da parete 3 in 1 (legno, metalli, cavi elettrici)

Rileva la presenza di cavi elettrici sotto tensione (con passaggio di corrente), strutture in legno (telai, travetti, barre, ecc), oggetti metallici (travi, telai, tubi, tondini, perni, barre in metallo). Dispone di ampio display retroilluminato, buzzer di segnalazione, pulsante di calibrazione automatica. Si spegne automaticamente dopo 3 minuti di inutilizzo. Alimentazione con batteria da 9 V (acquistabile separatamente), dimensioni 156x70x26 mm.

€29,50

Cod. FR760

Rilevatore da parete professionale (legno, metalli, cavi elettrici)

Rileva la presenza di cavi elettrici sotto tensione (con passaggio di corrente), strutture in legno (telai, travetti, barre, ecc), oggetti metallici (travi, telai, tubi, tondini, perni, barre in metallo). Dispone di ampio display LCD retroilluminato, buzzer di segnalazione, spegnimento automatico dopo 5 minuti di inutilizzo. Alimentazione tramite batteria al litio (integrata) da 3,7 volt 1500 mAh. Dimensioni 195x90x35 mm.



- ✓ Professionale
- ✓ Batteria al litio ricaricabile integrata da 3,7 V - 1500 mAh con autonomia fino a 10 ore
- ✓ Case robusto
- ✓ Prestazioni migliorate

€78,00

Cod. FR801

Misuratore di distanza fino a 80 m con LASER verde e messaggi vocali

Metro LASER multifunzione compatto con display LCD retroilluminato, messaggi vocali, menu a icone, livella digitale, memoria interna (in grado di memorizzare fino a 100 misurazioni) e batteria ricaricabile agli ioni di litio. È in grado di effettuare le seguenti misurazioni: distanza (da 20 cm a 80 metri), area, volume, misurazione pitagorica, misurazione verticale e molte altre. Visualizza lo stato di carica delle batterie; si spegne automaticamente se inattivo (laser dopo 30 secondi, strumento dopo 2 minuti). Grado di protezione IP54. Dimensioni 120x50x26 mm.



€88,00

Cod. FR798

Misuratore di distanza LASER da 20 cm a 30 m

Compatto metro LASER con display LCD retroilluminato. Permette di misurare in modo estremamente facile e veloce la distanza (da 20 cm a 30 metri), l'area e il volume. Dispone di spegnimento automatico del LASER e del dispositivo; visualizza anche lo stato di carica delle batterie. Alimentazione con 2 batterie mini stilo da 1,5 volt tipo AAA (incluse). Dimensioni 105x34x20 mm.



€32,00

Cod. FR797

Prezzi IVA inclusa

RA4, i nuovi microcontrollori a 32 bit di Renesas

Renesas estende la famiglia RA di microcontrollori a 32 bit con quattro dispositivi basati su core Arm Cortex-M33, in package da 48 pin a 64 pin e da 256 kB a 512 kB di memoria Flash più 128 kB di SRAM. I dispositivi RA4E1 assorbono solo 60µA/MHz nell'esecuzione da Flash a 100MHz e 81µA/MHz eseguendo l'algoritmo CoreMark da Flash. Le caratteristiche degli RA4 sono:

- CPU Arm Cortex-M33 a 100 MHz;
- USB 2.0 Full-Speed Device;
- comunicazione seriale, CAN e QSPI;
- oscillatore interno ad alta precisione ($\pm 0,25\%$) su chip che utilizza FLL;
- GPIO analogico avanzato, rilevamento a bassa tensione e funzione di ripristino interno.

Renesas ha combinato gli MCU RA4E1 con dispositivi analogici e di alimentazione complementari che lavorano insieme in 250 combinazioni per cre-



are una varietà di applicazioni, tra cui un purificatore d'aria IoT intelligente e una macchina da caffè che usa gli MCU RA4E1. Per iniziare a sviluppare con gli MCU RA4E1, Renesas offre la scheda di prototipazione rapida FPB-RA4E1 con debugger on-chip. Tutti i dispositivi RA sono supportati dal Renesas Flexible Software Program (FSP) che include

driver e middleware altamente efficienti per facilitare l'implementazione delle comunicazioni e della sicurezza. La GUI dell'FSP semplifica e accelera il processo di sviluppo. Consente un uso flessibile del codice legacy, nonché una facile compatibilità e scalabilità con altri dispositivi della famiglia RA.

www.renesas.com

Gate driver digitale per MOSFET SiC



Microchip annuncia un nuovo gate driver digitale da 1.200 V a doppio canale AgileSwitch 2ASC-12A2HP, per MOSFET SiC, completamente configurabile. Rispetto ai gate driver convenzionali, i gate driver AgileSwitch includono la capacità di smorzare gli overshoot di tensione drain-source (V_{ds}) fino all'80% e ridurre le perdite di commutazione fino al 50%. Il driver supporta fino a 10 A di corrente di picco sink/source e include un convertitore CC/CC

isolato con barriera di isolamento a bassa capacità, per segnali di modulazione a larghezza di impulso e feedback di guasto. Il 2ASC-12A2HP è compatibile con l'ultima versione dell'Intelligent Configuration Tool (ICT) Microchip, che consente agli utenti di configurare i parametri del gate driver, inclusi i profili di commutazione di gate, i monitor critici del sistema e le impostazioni dell'interfaccia del controller.

www.microchip.com

Sensore ToF multizona 8x8 da ST

STM ha annunciato sensore FlightSense VL53L5CX che fornisce fino a 64 zone di rilevamento con rilevamento multi-target, misurazione della distanza fino a quattro metri in ciascuna zona e un ampio campo visivo con bordi quadrati con diagonale di 63°. Il sensore è adatto al riconoscimento dei gesti, all'analisi di scene complesse, inclusa la mappatura 3D della stanza per la robotica, il monitoraggio del livello di serbatoi o il monitoraggio del livello dei bidoni dei rifiuti per consentire la raccolta intelligente.

Supportando il riconoscimento dei gesti e della presenza umana, il VL53L5CX consente l'attivazione senza contatto di apparecchiature accessibili al pubblico come terminali di pagamento self-service, nonché di elettrodomestici. Dotato dell'elaborazione d'istogramma ST, che riduce l'impatto

della diafonia del vetro di copertura, il sensore conta sulla programmabilità del numero di array di rilevamento e su un massimo di 60 fps in modalità a 16 zone (4x4) a portata rapida. Il VL53L5CX contiene un MCU low-power, viene fornito in un modulo completamente integrato da 6,4x3x1,5 mm che contiene un emettitore a infrarossi VCSEL e un ricevitore a Single-Photon Avalanche Diode, più un motore di elaborazione ToF basato su istogramma. Il VL53L5CX è supportato dal kit di valutazione P-NUCLEO-53L5A1 e dal pacchetto di espansione software X-CUBE-TOF1.

www.st.com



Telecamere ADAS da Xilinx e Motovis



La piattaforma SoC di Xilinx Automotive (XA) Zynq e l'IP della rete neurale convoluzionale (CNN) di Motovis offriranno potenti sistemi di telecamere anteriori, che nell'ADAS perché forniscono le capacità di rilevamento necessarie ad esempio per l'assistenza al

mantenimento della corsia (LKA), la frenata di emergenza automatica (AEB) e il cruise control adattivo (ACC). La soluzione supporta parametri necessari per i requisiti NCAP 2022. La soluzione è scalabile sulle famiglie di SoC XA Zynq da 28 nm e 16 nm utilizzando

l'IP CNN di Motovis, una combinazione di funzionalità di partizionamento hardware e software ottimizzate con motori specifici che ospitano le reti di Deep Learning Motovis. La soluzione supporta risoluzioni d'immagine fino a 8 Mpx. www.xilinx.com

AFE elettromedicale a 4 parametri da Analog Devices

Si chiama MAX86178, il nuovo front-end analogico single-chip che consente la misurazione sincronizzata di elettrocardiogramma (ECG), frequenza cardiaca, SpO2 e frequenza respiratoria negli apparati ospedalieri; integra tre sistemi di misurazione (ottico, ECG e bioimpedenza) per ottenere quattro segni vitali comuni e frequenza respiratoria (usando BioZ). Il MAX86178 consente la temporizzazione ottica PPG ed ECG sincronizzata per le metriche di salute derivate. I progettisti di dispositivi medici sono desiderosi di ridurre le spese sanitarie annuali sostituendo le attività in ufficio sistemi di monitoraggio sanitario con dispositivi wireless più piccoli, a bassa potenza, indossati in modo discreto e continuamente in casa o in ufficio. Con tre sottosistemi di livello clinico integrati in un unico CI, il MAX86178 AFE sostituisce le implementazioni discrete integrando un sottosistema ottico PPG per misurare la frequenza cardiaca e la SpO2, un sottosistema ECG a derivazione singola, nonché un biopotenziale



e sottosistema di bioimpedenza (BioZ) per misurare la frequenza respiratoria. Consente piccoli dispositivi di segni vitali di adattando queste molteplici funzioni in un pacchetto da 2,6 mm x 2,8 mm. Inoltre, gli RPM indossabili di nuova generazione devono funzionare a bassa potenza per consentire batterie più piccole o prolungare la durata della batteria per consentire requisiti di ricarica più convenienti. Per abilitare le funzioni a

bassissima potenza, il MAX86178 fornisce a ciascun sottosistema opzioni configurabili per ottimizzare la durata della batteria per specifiche casi d'uso. Analog Devices offre il regolatore buck-boost MAX20343 e l'alimentazione MAX20360, CI di gestione come soluzioni di alimentazione ottimizzate per MAX86178. Il MAX86178 è incapsulato in package WLP da 2,6x2,8 mm. www.analog.com

Driver stepper-motor 40V/2A senza resistore current sensing da Toshiba

Toshiba presenta il driver per motori passo-passo TB675539FTG, che supporta il controllo a corrente costante, per l'automazione civile e industriale. Il nuovo driver esegue il controllo del motore a corrente costante senza un resistore di rilevamento della corrente. TB675539FTG incorpora l'ul-

timo dispositivo DMOSFET di Toshiba, che gli consente di operare fino a 40 V e 2 A. Il circuito a ponte H per il controllo del motore ha una configurazione Nch/Nch e un circuito a pompa di carica integrato per il controllo dello stadio di uscita. Inoltre non necessita del condensatore esterno che generalmente viene utilizzato

per pilotare i gate sul ponte ad H, perché è incorporato. Il TB675539FTG supporta alimentatori per motori da 4,5 V a 34 V e può essere utilizzato per applicazioni da 12 a 24V, con assorbimento di 1μA in sleep-mode. Il TB675539FTG è incapsulato in package QFN32 da 5x5 mm. www.toshiba.com



I driver motore BLDC Texas Instruments accelerano la progettazione



Texas Instruments ha presentato i primi driver per motori CC brushless (BLDC) da 70 W capaci di fornire un controllo trapezoidale e orientato al campo (FOC) senza richiedere la scrittura di codice e senza sensori. I dispositivi semplificano la progettazione del controllo per motori BLDC al punto da poterne mettere a punto uno in meno di 10 minuti.

Con il controllo integrato in tempo reale di fino a 18 componenti discreti, inclusi i MOSFET di potenza, i dispositivi TI accelerano anche i tempi di risposta del sistema riducendo lo spazio sul PCB fino al 70%. I driver del motore BLDC MCF8316A e MCT8316A includono un set unico di algoritmi di controllo della commutazione che eliminano la necessità di sviluppare, mantenere e qualificare il software di controllo del motore, eliminando mesi di tempo di progettazione. Gli algoritmi, insieme a un alto livello di integrazione, consentono di gestire funzioni critiche come il rilevamento dei guasti del motore, implementando meccanismi di protezione per aumentare l'affidabilità del sistema. L'adozione della tecnologia sensorless per determinare la posizione del rotore, elimina la necessità di sensori Hall esterni.

Il driver del motore di controllo trapezoidale sensorless

MCT8316A consente ai progettisti di mettere a punto un motore utilizzando solo cinque pin hardware, semplificando i sistemi eliminando la necessità di un'interfaccia microcontrollore. I driver del motore BLDC MCF8316A e MCT8316A forniscono funzionalità avanzate di controllo in tempo reale per ridurre le emissioni acustiche attraverso una tecnica brevettata e precisa di compensazione automatica dei tempi morti, che compensa la distorsione della corrente del motore. Il driver MCT8316A può raggiungere una velocità elettrica massima del motore di 3,5 kHz, che è più di qualsiasi altro driver del genere. Ciò accelera la risposta del sistema in applicazioni come gli aspirapolvere robotici che richiedono un controllo del motore rapido e preciso.

L'MCT8316A e l'MCF8316A sono anche i primi del settore a fornire un modo rapido e controllato per decelerare attivamente un motore, consentendo di arrestare un motore il 50% più velocemente rispetto alle tradizionali tecniche di controllo del motore. I dispositivi integrano tre gate driver e sei MOSFET high-side e low-side con 50 mΩ di Rdson, capaci di commutare fino a 70 W di potenza con 8 A, in sistemi a 12 e 24 V.

www.ti.com

Sensore ToF QVGA con filtro IR da Melexis

Melexis ha sviluppato una nuova versione della sua fotocamera 3D time-of-flight MLX75026 integrandole un filtro passa-banda a infrarossi (IRBP) grazie al quale non è più necessario includere un filtro IR separato nell'obiettivo o nel gruppo sensore. Unica nel settore, questa soluzione riduce la complessità e i costi di progetto. MLX75026 con IRBP è un sensore ToF volo QVGA qualificato AEC-Q100 e si rivolge alle applicazioni automobilistiche come il monitoraggio del conducente (DMS), il monitoraggio in cabina, il controllo dei gesti, le applicazioni di sicurezza relative all'attivazione dell'airbag, l'allineamento del conducente del display head-up (HUD) ma anche alle applicazioni industriali come i veicoli a guida automatica (AGV) e la robotica collaborativa.

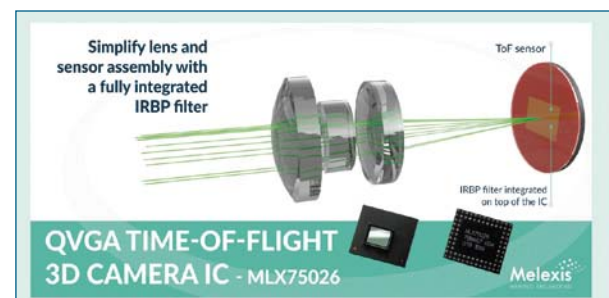
L'integrazione del corretto filtro passa-banda IR non è banale: la banda passante deve essere adattata all'illuminazione e il filtro deve accettare la luce da un'ampia gamma di incidenza con uno spostamento spettrale minimo (filtro otticamente duro). Ciò facilita l'utilizzo di obiettivi ToF con aperture elevate (numero f basso). Melexis è l'unico produttore

di sensori ToF che attualmente offre questa opzione, ora disponibile come variante dell'MLX75026. L'MLX75026 offre compatibilità software con altri sensori ToF Melexis gen 3.

MLX75026 con filtro IRBP ha una risoluzione QVGA (320x240 pixel) con un formato ottico da 1/4 di pollice e supporta l'illuminazione a 940 nm. La configurazione avviene tramite I²C-Bus e fornisce l'output dei dati seriali CSI-2. L'elevata integrazione, le dimensioni ridotte e il basso consumo energetico del sensore, insieme all'integrazione dei filtri passa-banda IR, consentono la progettazione di telecamere ToF compatte ed economiche.

Per lo sviluppo, verrà rilasciato il kit di valutazione EVK75026+ basato su un MPSoC Xilinx Zynq Ultrascale+, che offre funzioni di calibrazione e filtro dei pixel, oltre al supporto per funzionalità sofisticate come l'estensione della gamma ambigua, la risoluzione di più percorsi e il rifiuto delle interferenze. L'interfaccia di programmazione (API), scritta in linguaggio C, include wrapper che consentono di utilizzare il kit sia con Matlab che con Python.

www.melexis.com





ABBONAMENTO
ANNUALE



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



Visita: www.elettronica.in/abbonamenti



ABBONAMENTO ANNUALE PER RIVISTA DIGITALE

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

COME ABBONARSI

→ **on-line**

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

→ **e-mail**

scrivendo a
abbonati@elettronica.in

→ **telefono**

telefonando a
+39-0331-752668

ABBONAMENTO ANNUALE PER RIVISTA **CARTACEA**

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

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

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

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



Taglia e incidi con **Cricut MAKER**



Prezzo IVA inclusa

Macchina da taglio intelligente



€ 439,00

cod. CRICUTMAKER

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





The micro:bit logo, featuring a stylized white 'O' with two dots inside, followed by the text 'micro:bit' in a bold, sans-serif font.

LA NUOVA SCHEDA 2.0

Nasce la nuova versione di micro:bit. Vediamo le differenze tra le varie versioni e con un esempio pratico verifichiamo flessibilità e potenza.

di **MARCONATALE PARISE**
e dell'Ing. **VINCENZO GERMANO**

Ideato da BBC (British Broadcasting Corporation, ovvero l'emittente britannica) è nato in principio come metodo facile ed economico per avviare giovani studenti del Regno Unito allo studio dell'Informatica, ma a oggi gli utilizzi che se ne possono fare sono i più disparati. La sua distribuzione iniziò nel 2016, come regalo per tutti gli studenti dell'Inghilterra e successivamente nel 2017 la scheda fu resa disponibile per la vendita in altre nazioni dell'Unione Europea, per poi dilagare rapidamente anche in altri Paesi.

Questo dispositivo grande quanto il palmo di una mano (appena 4 x 5 centimetri) racchiude in sé un ritrovato tecnologico che altre schede concorrenti non possono vantare; se a questo si aggiunge un costo di qualche decina di euro, il risultato è imparagonabile. Come vedremo a breve in questo articolo, in estrema sintesi la scheda è basata su un micro-

controllore Nordic in tecnologia ARM (Advanced RISC Machine) Cortex M a 32 bit, corredata di varie periferiche ingresso/uscita, Bluetooth, sensoristica, LED, pulsanti, touch, speaker e molto altro, per renderla usabile e scalabile per un numero elevato di applicazioni.

Tenendo presente che la missione primaria di Microbit è di rendere le tecnologie "numeriche" il più accessibili nell'ambiente dell'istruzione primaria, questa scheda può essere utilizzata per una grande varietà di progetti a livello didattico, principalmente di carattere informatico e di Coding, rendendo molto semplice la realizzazione/creazione di giochi, robot, strumenti musicali, oggetti indossabili e molto altro.

Programmabile in diverse modalità come vedremo a breve, tuttavia quella principale fa uso di un linguaggio di programmazione "grafico", costituito da blocchi funzionali che possono essere tra loro interconnessi. La programmazione avviene mediante linguaggi grafici molto diffusi nella scuola come Scratch e non a caso ben si adatta a essere impiegato con studenti di scuola elementare e/o scuola media; può avvenire usando qualsiasi tipo di dispositivo (Tablet, Smartphone, PC) e può essere effettuata direttamente on-line senza nessuna installazione sul proprio computer, nel rispetto dei più recenti paradigmi di insegnamento dell'informatica. Oltre all'offerta di strumenti di programmazione liberamente accessibili via web, sono presenti innumerevoli guide di esempio e tutorial di qualsiasi tipo, non solo orientati a bambini/ragazzi per la prima informatizzazione ma anche per un utilizzo complesso e molto avanzato della scheda.

A tutto ciò si aggiunge anche la disponibilità di avere specifiche tecniche e "hardware open source",

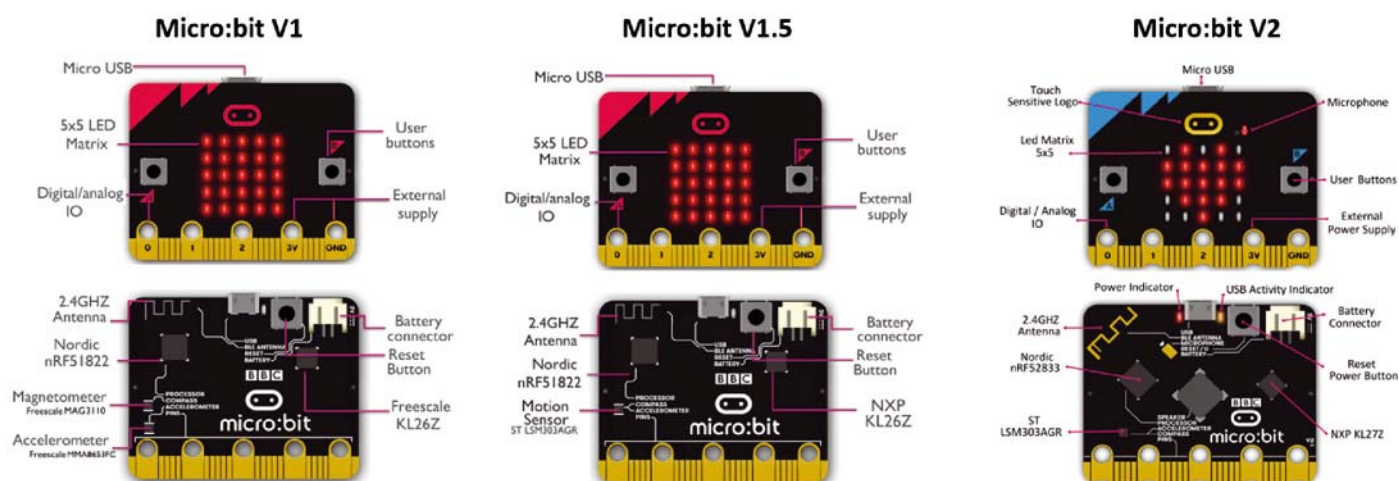
dai disegni di assemblaggio al pin-out della scheda, che rendono molto fruibile l'inserimento di questa board in ambiti più complessi.

Prima di passare a un'analisi di dettaglio e di confronto tra le varie versioni rilasciate va detto che esistono varie "confezioni d'acquisto", che vanno dalla scheda singola e null'altro, a pacchetti che hanno schede d'espansione e sensoristica aggiuntiva facilmente collegabile mediante board millefori, perciò anche in questo caso c'è l'imbarazzo della scelta.

CONFRONTO VERSIONI MICROBIT

Come appena visto, la BBC ha lanciato la prima versione del Microbit nel 2016, aggiornando negli anni il suo prodotto con caratteristiche hardware sempre più performanti e introducendo periferiche maggiormente innovative. In questa sezione vedremo un confronto tra le versioni Microbit attualmente presenti sul mercato mettendo in risalto le differenze salienti tra i vari rilasci di versione. Partiamo con la trattazione riguardo le differenze dalla **Fig. 1**, nella quale viene mostrata una panoramica del dispositivo nelle tre versioni a oggi acquistabili. Per completezza viene riportato sia il lato TOP (le tre immagini in alto) che il lato Bottom (le tre immagini in basso) di ogni scheda. Come si può notare a colpo d'occhio il lato TOP non ha subito molte evoluzioni nei vari passaggi, se non l'inserimento del touch direttamente sul logo e l'aggiunta del microfono. Gli stravolgimenti maggiori interessano principalmente il lato Bottom della versione V2, nella quale si apprezza un cambio layout sostanziale dovuto principalmente all'utilizzo di nuovi componenti hardware, all'inserimento dello speaker e a una riprogettazione dell'antenna.

Fig. 1
Panoramica
Microbit .



HARDWARE	VERSIONE 1	VERSIONE 1.5	VERSIONE 2
Matrice di LED 5x5	Presente	Presente	Presente
Tasti Utente	Presente	Presente	Presente
Alimentazione esterna	3.3V	3.3V	3.3V
Logo sensitive touch	Non Disponibile	Non Disponibile	Disponibile
Microfono	Non Disponibile	Non Disponibile	Indicatore LED Foro per microfono input
Antenna 2.4GHz	BLE, Broadcast radio, Bluetooth 4.1	BLE, Broadcast radio, Bluetooth 4.1	BLE, Broadcast radio, Bluetooth 5.1
Connettore Batteria	JST connessione 3V	JST connessione 3V	JST connessione 3V
Magnetometro	Freescale MAG3110	Non Presente	Non Presente
Accelerometro	Freescale MMA8653FC	Non Presente	Non Presente
Sensore di Movimento	Non Presente	ST LSM303AGR	ST LSM303AGR
Chip Interfaccia USB	Freescale NXP KL26Z	Freescale NXP KL26Z	Freescale NXP KL27Z
Tasto di Reset	Non Presente	Non Presente	Presente
Speaker	Non Presente	Non Presente	Presente
Chip Principale	Nordic nRF51822	Nordic nRF51822	Nordic nRF52833

Tabella 1
Confronto
tra le varie
versioni
Microbit
rilasciate.

Come vedremo a breve, senza ombra di dubbio l'ultima versione introduce nuove caratteristiche molto più vicine a quelle che possono essere le esigenze del mercato ma anche una normale evoluzione tecnologica.

Prima di addentrarci in una trattazione di dettaglio, riassumiamo in una tabella molto schematica le principali differenze tra le versioni in termini di "macro" funzionalità. In **Tabella 1** vengono riportate in verde le differenze/modifiche tra le varie versioni. Dalla tabella si evince che oltre alle macro novità di microfono, speaker e touch, la versione V2 porta con sé l'aggiunta del Bluetooth 5.1 e un nuovo chip della Nordic, ma procediamo con ordine analizzando nel dettaglio le differenze hardware specifiche tra le varie versioni confrontando le tecnologie e i chip interessati.

Per agevolare la lettura dei successivi contenuti, la **Fig. 2** schematizza il Microbit V2 rispetto al Microbit V1 così da evidenziare le interconnessioni e le parole chiave che successivamente verranno presentate.

SYSTEM ON CHIP NORDIC NRF5 SERIE

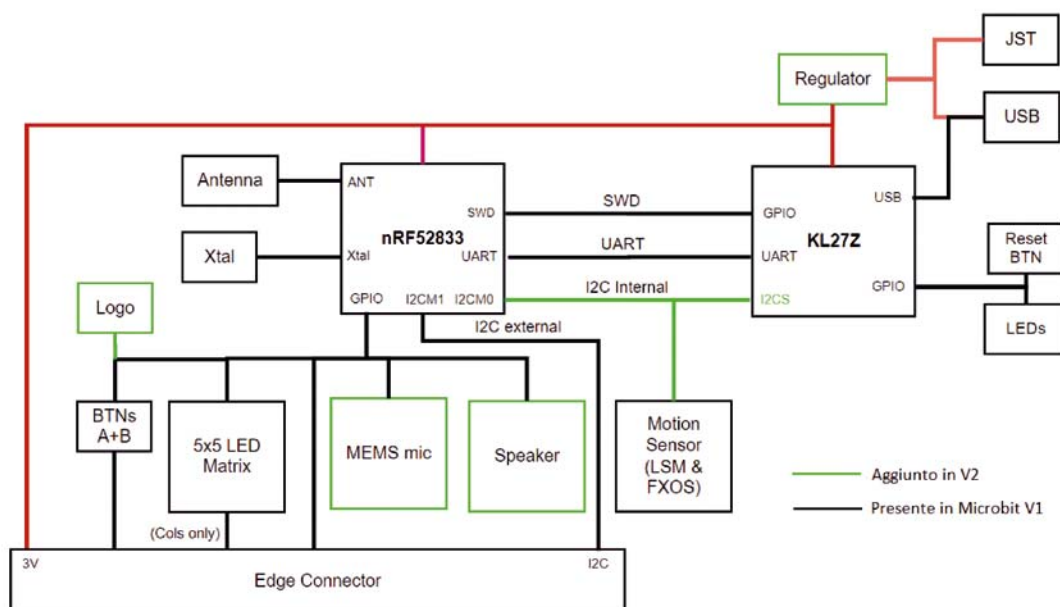
Partiamo con il confronto dal cuore della scheda, ovvero il chip. Sin dalla prima versione V1, la board prevede un chip della Nordic Semiconductor, una società di semiconduttori "fabless" (ovvero senza una fabbrica proprietaria) con sede in Norvegia,

specializzata in sistemi wireless su chip (System on Chip - SoC) a prestazioni ultra-basse e dispositivi di connettività per la banda ISM a 2,4 GHz, orientata principalmente nel contenere il consumo energetico e i costi dei dispositivi hardware.

Con riferimento alla **Tabella 1**, tra le varie versioni di Microbit rilasciate, si nota che nella V2 il chip della Nordic ha subito un cambio di "part number" (ovvero il codice univoco e identificativo del chip), che ha portato a un cambio sostanziale del chip, ma andiamo per gradi. Sulle board in versione V1 e V1.5 viene montato un SoC Nordic nRF51822 che offre soluzioni per applicazioni wireless ultra low power tramite il processore 16MHz ARM Cortex-M0 32bit, che prevede una memoria flash di 256kB e una RAM di 16kB. Questo SoC mette a disposizione differenti periferiche, per citarne alcune, timers, counters, temperature sensor, analog Input-Output (IO), digital IO, SPI, I2C, PWM ma anche periferiche di sistema e differenti GPIO (general purpose Input-Output) pin configurabili in ottica low-power. Tra le innumerevoli caratteristiche, sicuramente, la serie nRF51 è stata progettata e ottimizzata per operare nel mondo ISM (Industrial, Scientific and Medical) con frequenze di banda da 2.4 a 2.4835 GHz e quindi applicazioni Bluetooth Low Energy (BLE). Nel mondo delle telecomunicazioni la Banda ISM è il nome assegnato dall'Unione Internazionale delle Telecomunicazioni (ITU) a un



Fig. 2
Schema a blocchi
Microbit V2.



insieme di porzioni dello spettro elettromagnetico riservate alle applicazioni di radiocomunicazioni non commerciali, ma per uso industriale, scientifico e medico. L'uso di queste bande può differire da Stato a Stato a causa di specifiche regolamentazioni nazionali, vi è tuttavia una definizione generale a livello globale. Recentemente tali bande sono state utilizzate per sistemi di comunicazione senza fili non licenziate come le wireless LAN (IEEE 802.11b/g/n e Bluetooth operano nella banda dei 2,4 GHz, mentre IEEE 802.11a opera nella banda dei 5,8 GHz).

Tornando al "mondo" Microbit la V2 aggiorna il SoC con il Nordic nRF52833, che integra e migliora le caratteristiche del chip precedentemente menzionato, prima tra tutte, il chip diventa un ARM Cortex-M4 a 32bit con una velocità nettamente superiore, ovvero 64MHz e grazie alla Floating Point Unit (FPU) garantisce alte performance e migliore efficienza energetica. Un'altra caratteristica da non sottovalutare è l'espansione delle memorie a 512kB per la flash e per la RAM a 128kB. In termini di applicazioni radio, il Nordic nRF52833 è stato identificato come SoC multiprotocollo in quanto supporta molteplici protocolli di comunicazione come IEEE 802.15.4, Thread, Zigbee, Bluetooth mesh ma soprattutto protocolli Bluetooth 5.1.

BLUETOOTH (4.1 VS 5.1)

Tra tutte le funzionalità disponibili, sicuramente il Bluetooth ha subito un notevole aggiornamento dalla versione 4.1, presente su Microbit v1 e v1.5, alla versione 5.1 disponibile con il Microbit v2.

La specifica Bluetooth è stata sviluppata dalla Ericsson e in seguito formalizzata dalla Bluetooth Special Interest Group (SIG), ovvero un'associazione di varie società. Questo protocollo è stato concepito come sistema standard per trasmissioni di dati per reti senza fili e con esso sono nate anche associazioni di attività produttive di dispositivi elettronici. Partendo dal presupposto che il sistema Bluetooth sfrutta onde radio sulla breve distanza, ogni dispositivo dotato di un chip è in grado di creare una rete di dimensioni limitate, a seconda della versione utilizzata (il raggio d'azione può variare da 10 m a un massimo di 30 m). Nel 2016 la quinta generazione si evolve aggiungendo, tra le altre cose, tre caratteristiche importanti, come mostrato in

Fig. 3:

- Consente ai dispositivi BLE (Bluetooth Low Energy) di usare una larghezza di banda di 2MHz e quindi raddoppiare i tempi di velocità di trasmissione.
- Il range di trasmissione è stato quadruplicato grazie all'introduzione di un nuovo schema di codifica nella gestione degli errori cosicché i segnali possono essere ricevuti correttamente anche per distanze più lunghe.
- La capacità di trasmissione è aumentata di otto volte utilizzando tutti i canali messi a disposizione. Nella generazione 4.x, dei 40 canali disponibili, solo 3 canali venivano usati per eventi di "advertising". Con il Bluetooth 5.0, i 3 originali canali sono gestiti come primari mentre gli altri 37 canali possono essere usati come canali secondari ma sempre per eventi di "advertising".

Per comprendere l'enorme passo avanti di questa tecnologia, basti pensare che una qualsiasi periferica BLE o nodo centrale, prima di accettare una connessione, invia sempre un messaggio di "advertising". Tali pacchetti sono l'unico modo che consente a un nodo centrale o a uno scanner di scoprire una qualsiasi periferica o altro nodo.

- Visto che con l'avvento della generazione 5.x, sono emerse nuove problematiche legate alla mancanza di estensibilità e breve copertura (un limite non da poco per applicazioni BLE), per superare queste problematiche è stato introdotto il modello "Mesh" che consente ai dispositivi connessi alla rete di comunicare tra di loro.

Come è facile immaginare il Bluetooth è una tecnologia wireless fondamentale, che combinando le caratteristiche della nuova generazione con il protocollo Low Energy e un sistema di rete di tipo Mesh, nella variante 5.x ha dato la possibilità a questa nuova versione di Microbit di fare un grande balzo in avanti, anche per qualsiasi applicazione Internet of Things (IoT).

MOTION SENSOR

Passando oltre ma sempre con riferimento alla **Tabella 1**, si evince come il Microbit v1 utilizza il magnetometro NXP Freescale MAG3110 in combinazione con l'accelerometro NXP Freescale MMA8653FC per realizzare una bussola elettronica in grado di fornire informazioni precise sulla direzione. Il magnetometro Freescale MAG3110 è un dispositivo low-power capace di misurare i campi magnetici lungo i tre assi X, Y, Z con un "data rate" fino a 80Hz. Sono disponibili le modalità operative standby e active dove la differenza risiede nella disattivazione/abilitazione di alcune periferiche.

Tramite opportuni registri di configurazione è possibile accedere alle informazioni relative all'intensità del campo magnetico, lungo i tre assi, espresso in micro Tesla (uT). Il protocollo di comunicazione per la lettura dati è il noto protocollo I2C (ovvero "Inter Integrated Circuit" o noto anche come IIC).

Mentre l'accelerometro Freescale MMA8653FC è un dispositivo lineare a tre assi con uscita digitale a bassa potenza. Le modalità operative disponibili sono Off, Standby e Wake/Sleep; proprio in quest'ultima tutte le periferiche sono disponibili così da poter utilizzare funzionalità come la rilevazione del "Freefall" (caduta libera), la rilevazione dell'orientamento e gesture aggiuntive tramite algoritmi software. Anche in questo caso il protocollo di comunicazione I2C consente di accedere ai dati relativi alla misura dell'accelerazione.

Al contrario il Microbit v1.5 introduce il motion sensor ST LSM3031AGR che combina in un unico chip le funzionalità di magnetometro e accelerometro. In particolare, il magnetometro si differenzia dal precedente chip riducendo i consumi e il rumore di misura grazie alle modalità operative low-power ed alta risoluzione con un "data rate" massimo di 150Hz. Tra le funzionalità aggiuntive, è possibile compensare il campo magnetico misurato nel caso di distorsione dovuta alla vicinanza con un oggetto magnetico.

Per quanto riguarda l'accelerometro, a differenza del precedente, il chip LSM3031AGR fornisce le modalità operative *high-resolution-mode*, *normal mode* e *low power mode* che consentono di ridurre i consumi con un data rate fino a 5376Hz. Il sistema aggiunge anche la funzione "Attività/Inattività" che consente di ridurre il consumo energetico relativo alla funzione accelerometro del chip riducen-

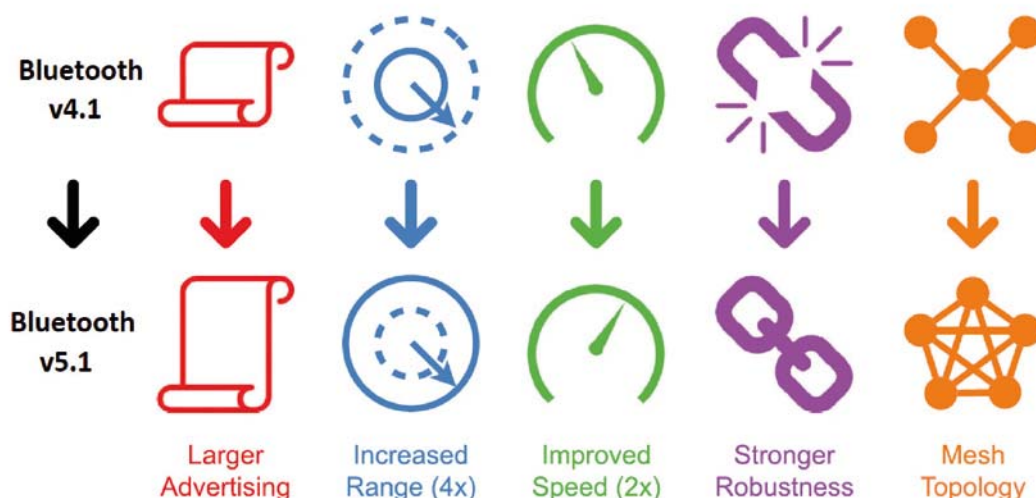
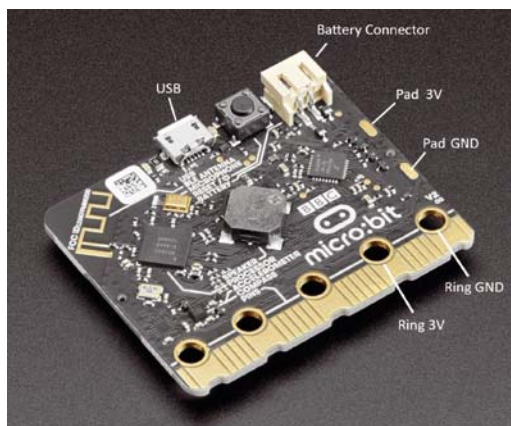


Fig. 3
Generazioni
Bluetooth a
confronto.



Fig. 4
Sorgenti
Alimentazione.



do automaticamente la frequenza di campioni acquisiti. Le funzioni relative al magnetometro ed all'accelerometro possono essere abilitate o messe in power-down mode separatamente. Infine il sistema include un'interfaccia I2C e un'interfaccia SPI.

USB INTERFACE CHIP

Il Microbit fino alla versione V1.5 presenta un'interfaccia USB che consente all'utente di programmare il dispositivo senza installare alcun driver. L'interfaccia USB è rappresentata da un Freescale KL26Z che comunica con il target Nordic attraverso i protocolli di comunicazione Serial Wire Debug (SWD) e UART (Universal Asynchronous Receiver-Transmitter - ricevitore-trasmettitore asincrono universale). La SWD consente la programmazione del dispositivo ma anche il debugging del codice sviluppato mentre l'UART viene utilizzata per lo scambio di informazioni dal Nordic al PC attraverso il cavo USB.

Il chip USB garantisce elevate performance grazie al processore Cortex-M0 che consente di spostare/copiare (drag-and-drop) i files all'interno del dispositivo al fine di caricare il codice nell'applicativo del SoC.

Con il Microbit V2 l'interfaccia USB è stata aggiornata con la serie Freescale KL27 che migliora le prestazioni del dispositivo raddoppiando la memoria fino a 256KB, aggiunge una memoria dedicata per l'aggiornamento software del dispositivo, introduce la periferica FlexIO (Flexible input and output peripheral) che emula una grande varietà di protocolli di comunicazione (UART, I2C, SPI) ma soprattutto riduce i consumi energetici grazie a un'accurata gestione della tensione di alimentazione e del clock interno (da 32KHz a 48MHz).

POWER SUPPLY

Un ulteriore punto a favore è relativo all'alimentazio-

ne della scheda. Il Microbit fino alla versione V1.5 può essere alimentato attraverso USB, connettore batteria, terminali sul connettore oppure tramite dei pad dedicati disposti a lato della board come è mostrato in **Fig. 4**.

Quando alimentato tramite USB, il chip KL26/KL27 è usato per fornire 3.3V alle periferiche tramite il suo regolatore interno con una corrente massima di 120mA. Se la scheda viene alimentata utilizzando il connettore batteria, il chip KL2x non verrà alimentato e il relativo LED rimarrà spento. Lo stesso accade quando la scheda viene alimentata tramite i due pad (GND e 3V) disposti sul connettore. Con il Microbit V2 le sorgenti di alimentazione sono rimaste invariate ma l'introduzione di nuove periferiche ha richiesto un'evoluzione dell'interfaccia USB con il chip KL27 e l'introduzione di un regolatore a monte del chip tale da ottimizzare i consumi energetici.

PERIFERICHE

Infine la scheda Microbit V2 introduce le funzionalità già brevemente introdotte come il microfono, lo speaker e il logo touch come mostrato in **Fig. 2**. Tra queste ultime innovazioni, quella più significativa è relativa al microfono. Realizzato in tecnologia MEMs (Micro Electro Mechanical Systems), ovvero una famiglia di dispositivi costituiti da strutture meccaniche e circuiti elettronici integrati sullo stesso chip di silicio, che fornisce un input sonoro al Microbit aprendo un ventaglio di possibilità e nuove applicazioni. In più tramite un led presente sul front della scheda, informa l'utente se il chip del MEMs è alimentato o meno; funzionalità molto comoda e intuitiva.

Infine, un'ultima periferica interessante è il logo touch, che come visto in precedenza viene gestito dal dispositivo ST LSM303AGR, che tramite algoritmi software aggiunge nuove funzionalità al dispositivo.

Dopo questa breve carrellata riguardo le differenze tra le varie versioni, passiamo a una panoramica riguardo la messa in funzione della scheda, utilizzando alcuni dei vari linguaggi di programmazione messi a disposizione.

TOOL-CHAIN MICROBIT

Dopo aver introdotto la board Microbit e averne analizzato le caratteristiche e differenze tra le varie versioni rilasciate negli anni, passiamo alla "messa in opera" della scheda facendo una breve carrellata di alcuni dei software utilizzabili per programmarla. Tutto ciò allo scopo di mostrare la semplicità d'uso pur con una scheda fortemente complessa come

la Microbit. Le caratteristiche hardware e software del Microbit combinate con gli strumenti di coding editing mettono a disposizione dell'utente diverse tecniche di aggiornamento software.

PYTHON TEXT

Partiamo da Python Text, è un tool online raggiungibile all'indirizzo <https://python.microbit.org/v/2>. Questo software è utilizzabile attraverso un qualsiasi browser (esempio Chrome, Firefox e altri) consentendo il download del programma realizzato tramite "webUSB" (che favorisce la comunicazione con dispositivi USB attraverso il browser) oppure scaricando direttamente il file ".hex" del progetto. Se si sceglie la programmazione tramite webUSB, è necessario seguire i vari step definiti nella **Fig. 5**. Vediamoli brevemente insieme, dopo aver collegato la scheda Microbit al PC tramite cavo USB, è necessario la verifica della connessione tramite il tasto "Connect" (**Fig. 5** Punto 1); se la sincronizzazione non presenta errori, è possibile procedere con il flashing del dispositivo (**Fig. 5** Punto 2) attraverso il tasto "Flash" e quindi aggiornare il codice all'interno del vostro Microbit (**Fig. 5** Punto 3). In alternativa, se si sceglie il secondo metodo, è possibile scaricare il file ".hex" attraverso l'apposito tasto di "Download" (**Fig. 5** Punto 1), che successivamente dovrà essere copiato all'interno della memoria flash del Microbit visibile sul PC come unità esterna (mass storage). Durante la procedura di copia, l'esecuzione del precedente codice verrà

interrotta e un led giallo presente sul lato bottom della scheda inizierà a lampeggiare, al termine della procedura il programma verrà avviato. Il tool consente anche di caricare o salvare il progetto software. Oltre a tali funzionalità c'è anche la possibilità di utilizzare una seriale. Come editor è molto scarso ma rapido ed efficace.

SCRATCH

Il secondo editor che presentiamo è Scratch, editor grafico online orientato verso utenti principianti che intendono acquisire le principali nozioni dello sviluppo software di dispositivi come il Microbit. La versione online è raggiungibile al seguente indirizzo: <https://scratch.mit.edu/projects/editor/>. La **Fig. 6** mostra la schermata principale del tool dove attraverso la selezione dei vari blocchi è possibile creare la propria applicazione. Al fine di interagire con il Microbit è necessario installare un'opportuna estensione che metterà a disposizione le funzioni principali input/output del dispositivo. Per la creazione del codice è necessario trascinare i blocchetti, disponibili nel foglio "Code", verso il centro della schermata; nell'esempio mostrato in **Fig. 6** sono utilizzate funzioni Microbit (evidenziate in arancione) e funzioni relative a "Costumes", disponibili in anteprima sulla destra (evidenziate nei riquadri in blu), per generare delle azioni alla pressione di un tasto (nell'esempio di **Fig. 6** il tasto è "A") tra cui la visualizzazione su PC di icone personalizzate ma anche la visualizzazione del testo

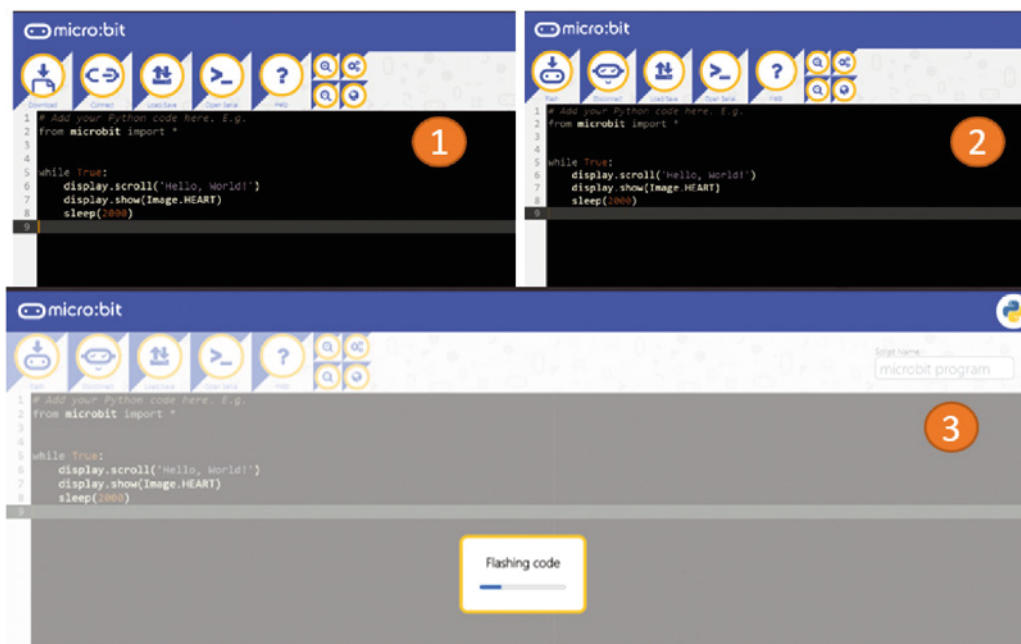
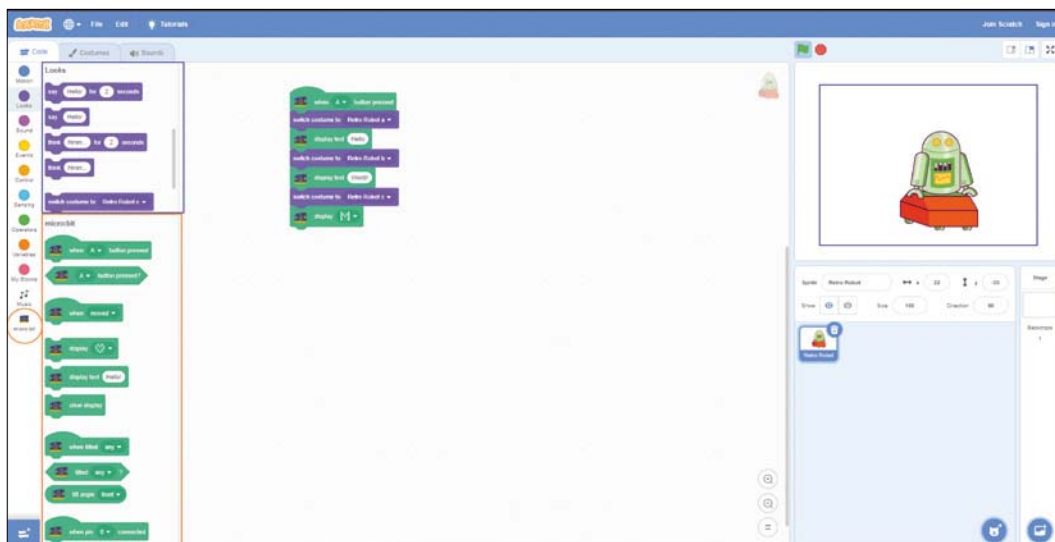


Fig. 5
Procedura
Flashing
Python Text.

Fig. 6
Editor grafico
Scratch.



desiderato tramite la matrice di led.
Per integrare le funzionalità del Microbit all'interno del tool Scratch è importante seguire la procedura definita a questo link:

<https://scratch.mit.edu/microbit>

Dopo aver aggiunto l'estensione, si può sincronizzare il tool Scratch con il Microbit, come mostrato in **Fig. 7** e quindi procedere alla scrittura del codice tramite le funzioni disponibili.

MAKE CODE BLOCK

Completiamo la carrellata di software per la programmazione di Microbit con Make Code Block, che è un tool grafico della Microsoft, a differenza di Scratch, è esclusivamente dedicato per applicazioni Microbit offrendo una vasta gamma di funzionalità. Come mostrato in **Fig. 8**, l'utente ha la possibilità di scegliere tra tre linguaggi di programmazione (Javascript, Python, Blocks) selezionabili dalla barra di navigazione del tool (**Fig. 8** Punto 1).

Il tool mette a disposizione una vasta gamma di funzioni, raggruppate per funzionalità (**Fig. 8** Punto

2), che variano dalla creazione di una variabile all'esecuzione di più istruzioni in corrispondenza di uno specifico evento (per esempio la pressione di un tasto).

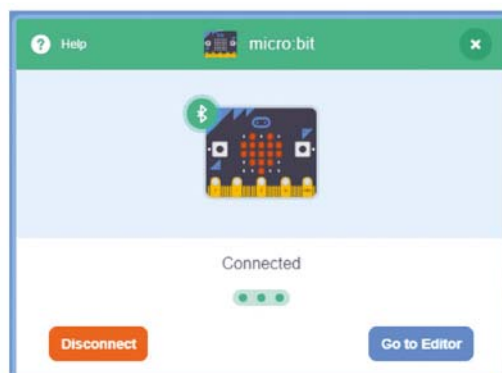
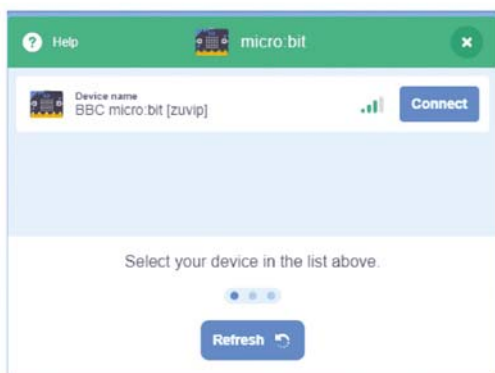
L'utente ha la possibilità di "debuggare" il codice oppure valutare il risultato della sequenza di istruzioni tramite la simulazione del Microbit (**Fig. 8** Punto 3).

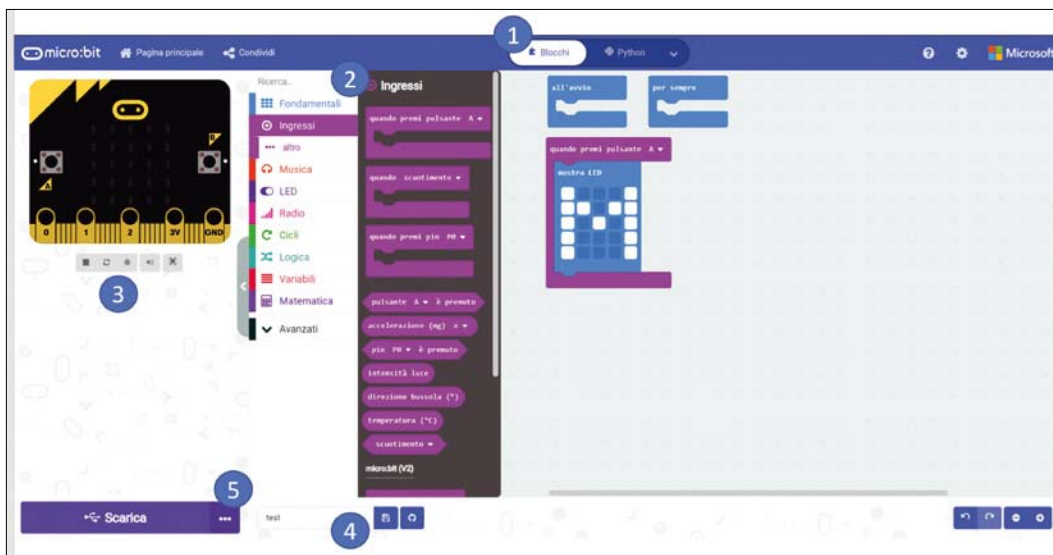
Il primo step da mettere in pratica è accoppiare il Microbit alla piattaforma, quindi:

- connettere il dispositivo al PC tramite cavo USB,
- cliccare sui puntini (**Fig. 8** Punto 5) e selezionare "Pair Device",
- tramite il pop-up mostrato, selezionare il dispositivo trovato e cliccare su "Connetti".

Seguendo questi semplici passaggi, saremo in grado di interagire con il Microbit tramite il Make Code Block e quindi programmare il dispositivo. Se la simulazione rispecchia l'atteso allora si può procedere con il flashing del dispositivo tramite il tasto "Scarica" oppure scaricare l'eseguibile del progetto (**Fig. 8** Punto 4) su PC.

Fig. 7
Setup Scratch
Microbit





← **Fig. 8**
Make Code
Block.

La piattaforma Make Code Block è disponibile anche su Android, sarà sufficiente cercare e scaricare l'applicazione "Microbit" sul Play Store.

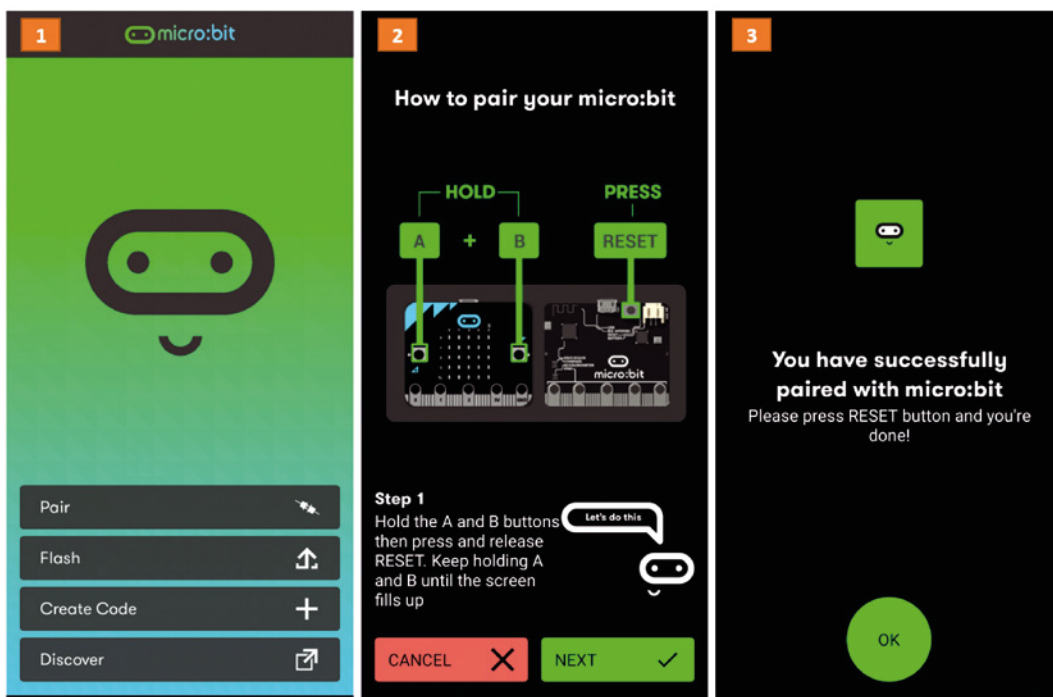
All'avvio, l'applicazione si presenta come mostrato in **Fig. 9** Punto 1 dove è possibile scegliere tra:

- *Pair*, operazione di accoppiamento necessaria per proseguire con l'attività di programmazione;
- *Flash*, per l'aggiornamento software del dispositivo;
- *Create Code*, per la scrittura del codice;
- *Discover*, reindirizzamento verso la pagina web per chi ha necessità di familiarizzare con il

dispositivo.

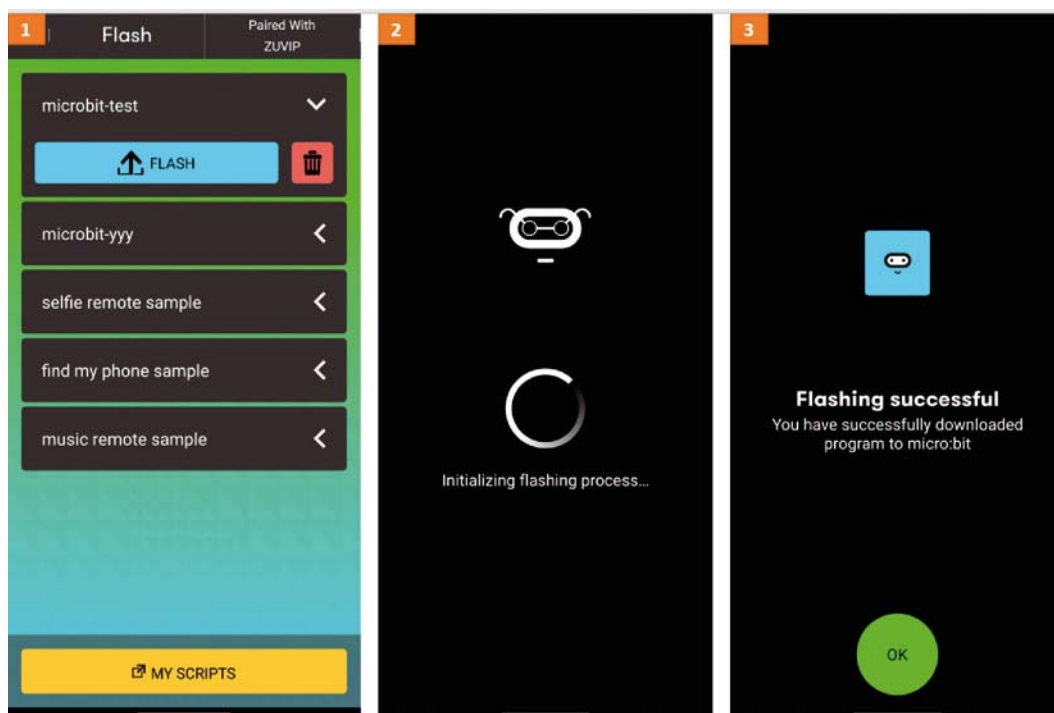
Per eseguire la procedura di accoppiamento (**Fig. 9** Punto 2) è necessario:

- Alimentare il dispositivo tramite cavo USB o batteria;
- Tenere premuti i tasti A e B sul dispositivo;
- Premere il tasto di Reset e rilasciarlo;
- Mantenere premuti i tasti A e B finché i led non verranno accesi;
- Digitare su applicazione il simbolo mostrato su display;
- Attendere qualche secondo fin quando la procedu-



← **Fig. 9**
Accoppiamento
Microbit con
Smartphone.

Fig. 10
Flashing
Microbit tramite
Smartphone.



ra non sarà completa (**Fig. 9** Punto 3).

A questo punto, è possibile procedere con la scrittura del codice che presenta un'identica interfaccia dell'applicazione desktop quindi non verrà approfondita. Per avviare la procedura di flashing (**Fig. 10**), è necessario seguire i seguenti passaggi:

- Selezionare l'icona "Flash" mostrata in **Fig. 9** Punto 1
- Selezionare il codice creato (**Fig. 10** Punto 1).
- Avviare la procedura di flashing tramite il tasto "Flash" (**vedi Fig. 10** Punto 1)

Se la procedura viene completata senza errori (**Fig. 10** Punto 3) allora il dispositivo sarà pronto per eseguire l'applicazione creata.

UN ESEMPIO PRATICO

Il progetto che verrà presentato è orientato all'utilizzo delle principali periferiche Microbit V2 quali Bluetooth, speaker, touch e protocolli di comunicazione. Make Code Block è la tool-chain che verrà utilizzata per la creazione del progetto in quanto risulta essere un software molto intuitivo e semplice da utilizzare per chi volesse addentrarsi nella realizzazione di un progetto abbastanza articolato. Prima di descrivere nel dettaglio i blocchi utilizzati e la loro disposizione, è importante definire le funzionalità che il nostro programma dovrà implementare:

1. Protocollo Bluetooth per l'invio e la trasmissione di dati che verranno visualizzati anche su display.

2. Intercettare pressione del logo touch per trasmettere via Bluetooth e mostrare su display la temperatura rilevata dal dispositivo nell'ambiente circostante.
3. Intercettare la pressione del tasto "A" per trasmettere via seriale l'informazione relativa al nome del dispositivo Microbit.

Per la verifica delle funzionalità avremo bisogno ovviamente del Microbit collegato al PC ma anche di due strumenti per la ricezione e trasmissione dati Bluetooth e Seriale.

A tale scopo utilizzeremo l'applicazione desktop Putty (scaricabile dal link: <https://www.putty.org>) e l'applicazione Android *Serial Bluetooth Terminal* disponibile sul Play Store.

La configurazione dei due strumenti aggiuntivi verrà dettagliata in seguito, iniziamo con la creazione del progetto e quindi con definire le prime azioni all'avvio del dispositivo.

Partiamo con il trascinare il blocchetto "all'avvio", disponibile nella sezione "Fondamentali", all'interno dell'area di codice e collegare le funzioni con le configurazioni definite di seguito:

- Collegare "serial set baud rate" disponibile nella sezione Seriale con un baud rate di 9600;
- Riprodurre un suono tramite la funzione "riproduci suono" disponibile nella sezione Musica;
- Definire l'icona da mostrare su display (matrice di LED 5x5) all'avvio, disponibile all'interno della

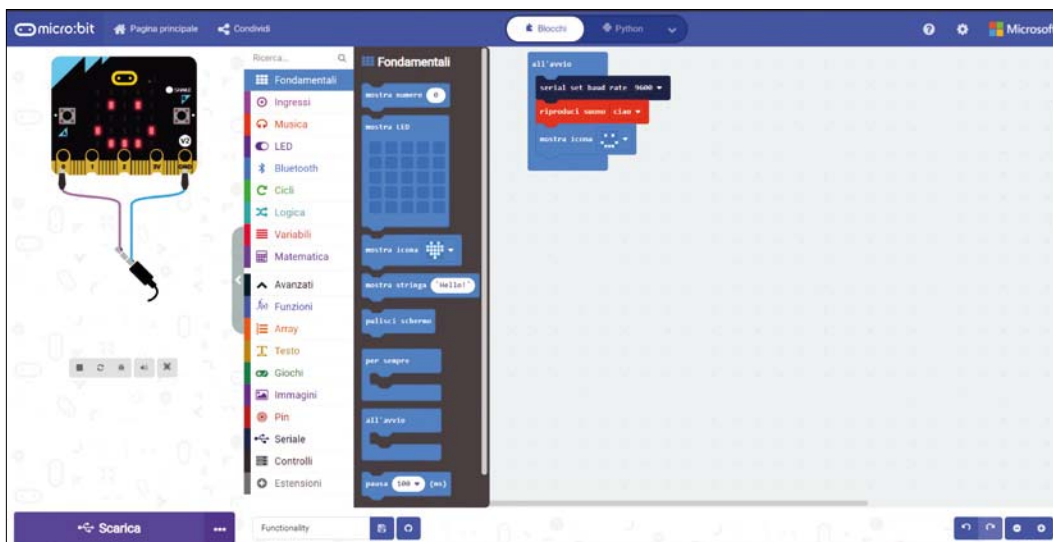


Fig. 11
Avvio
Applicazione.

sezione Fondamentali.

A questo punto, è possibile definire le prime azioni alla pressione del tasto A e del logo touch presenti all'interno della sezione "Ingressi" come mostrato nella **Fig. 12**.

Per quanto riguarda la funzionalità da attribuire alla pressione del tasto A:

- aggiungere il blocco "seriale – scrivi riga" disponibile nella sezione Seriale
- selezionare la variabile "nome del dispositivo" disponibile nella sezione "Controlli – Altro", trascinarla e collegarla al blocchetto di scrittura su seriale come mostrato in **Fig. 12**.
- aggiungere il blocchetto "mostra icona" disponibile nella sezione Fondamentali e selezionare l'icona desiderata

La funzionalità che verrà attribuita alla pressione del logo è implementata in **Fig. 12** dove è richiesto:

- la creazione di una variabile "temp", alla quale verrà associato il valore di temperatura acquisito dalla scheda. Per fare ciò, cliccare in sezione "Variabili", premere il tasto "Crea una variabile", digitare il nome della variabile ed infine trascinare il blocchetto di "Imposta variabile" all'interno dell'area di codice. Nello specifico, questa implementazione richiede la conversione della variabile temperatura in stringa tramite la selezione del blocchetto "unisci". Perciò nella sezione Testo, trascinare e collegare la variabile "temperatura" disponibile nella sezione Ingressi ed aggiungere in coda la stringa "deg".
- utilizzare la variabile "temp" per visualizzare la

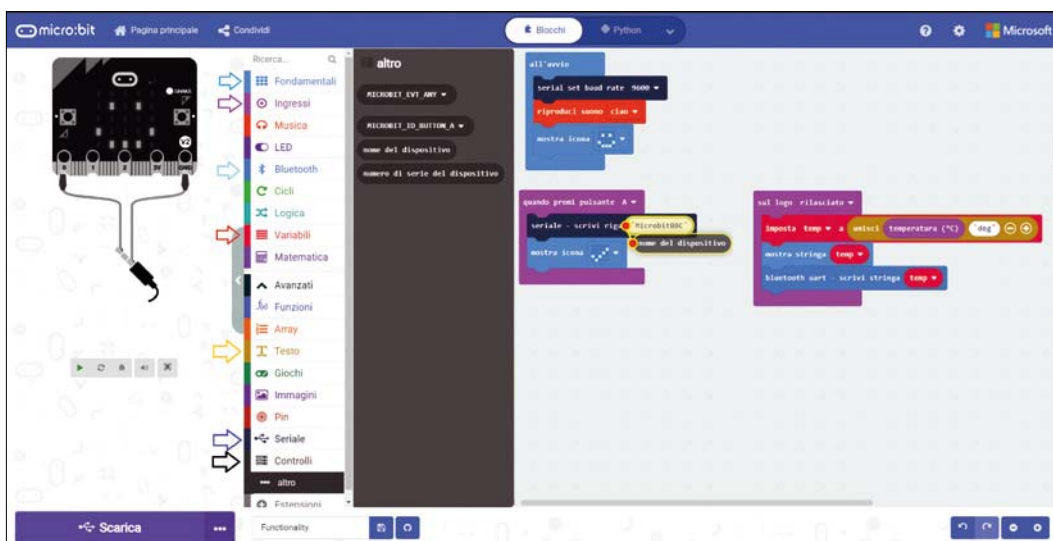
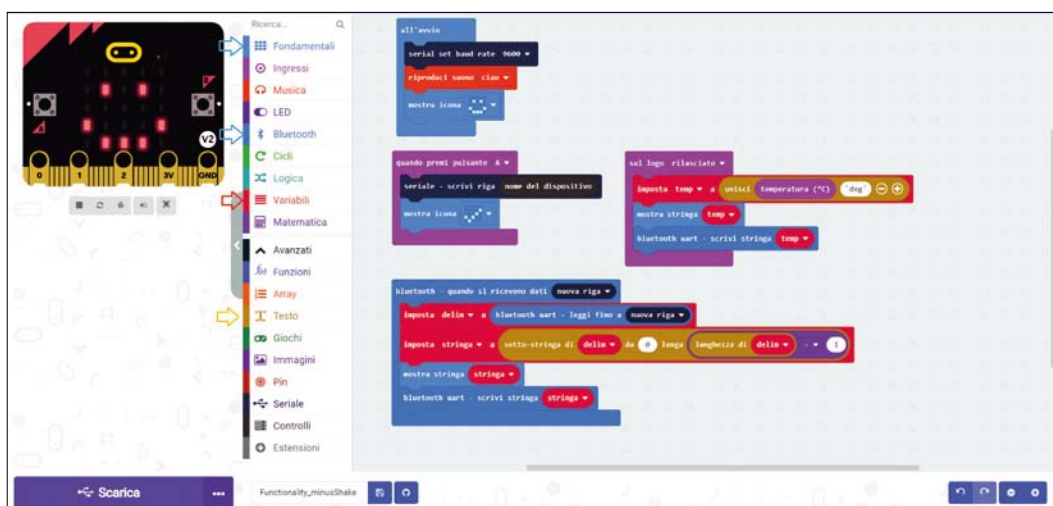


Fig. 12
Gestione
tasti.

Fig. 13
Applicazione
Completa.



temperatura su display e trasmetterla via Bluetooth tramite la funzione “bluetooth uart – scrivi stringa” disponibile in Bluetooth- altro.

L’implementazione completa sin qui illustrata è presente in **Fig. 13**.

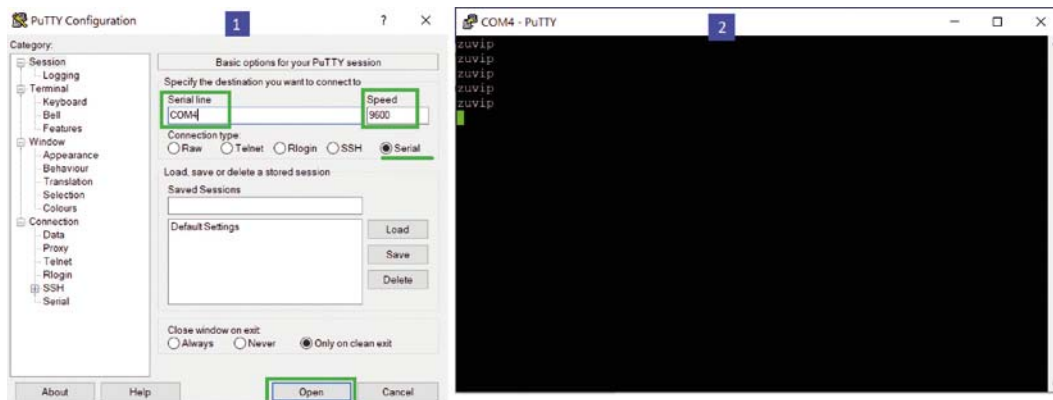
L’ultima fase di progettazione richiede la trasmissione e ricezione di dati Bluetooth, a tale scopo dovrà essere seguita la seguente procedura:

- Aggiungere il blocco “bluetooth -quando si ricevono dati” nella sezione Bluetooth e selezionare “nuova riga” dal menu a tendina del blocchetto.
- Creare la variabile “delim” ed assegnare il valore letto via Bluetooth delimitato da “nuova riga”.
- Creare la variabile “stringa” ed assegnare una sottostringa di “delim” compresa tra indice 0 e la sua lunghezza-1 al fine di rimuovere carattere inattesi. La funzione “sotto-stringa” è disponibile all’interno della sezione Testo.
- Utilizzare la variabile “stringa” per mostrare il dato su display e ritrasmettere il dato via Bluetooth.

A questo punto la scrittura del programma è completa, quindi è possibile aggiornare il Microbit e verificare i risultati attraverso gli strumenti Putty e l’applicazione Android Serial Bluetooth Terminal. Terminato il flashing, Il dispositivo visualizzerà uno “smile” ed emetterà un suono come definito all’interno della funzione “all’avvio” mentre per la verifica delle altre funzionalità è necessario procedere con la configurazione degli strumenti. Per configurare Putty, accedere alla gestione dispositivi del proprio PC così da verificare la COM alla quale è connesso il Microbit quindi aprire Putty, selezionare Serial, impostare lo speed a 9600, la COM alla quale è connesso il dispositivo e infine cliccare sul tasto “Open” come mostrato in **Fig. 14** Punto 1. Alla pressione del tasto A verrà visualizzato il nome del dispositivo (**Fig. 14** Punto 2).

Per configurare l’applicazione Android Serial Bluetooth Terminal, aprire l’applicazione dal menù disponibile sulla sinistra, selezionare Devices e verificare che il Microbit sia accoppiato come mostrato

Fig. 14
Putty.



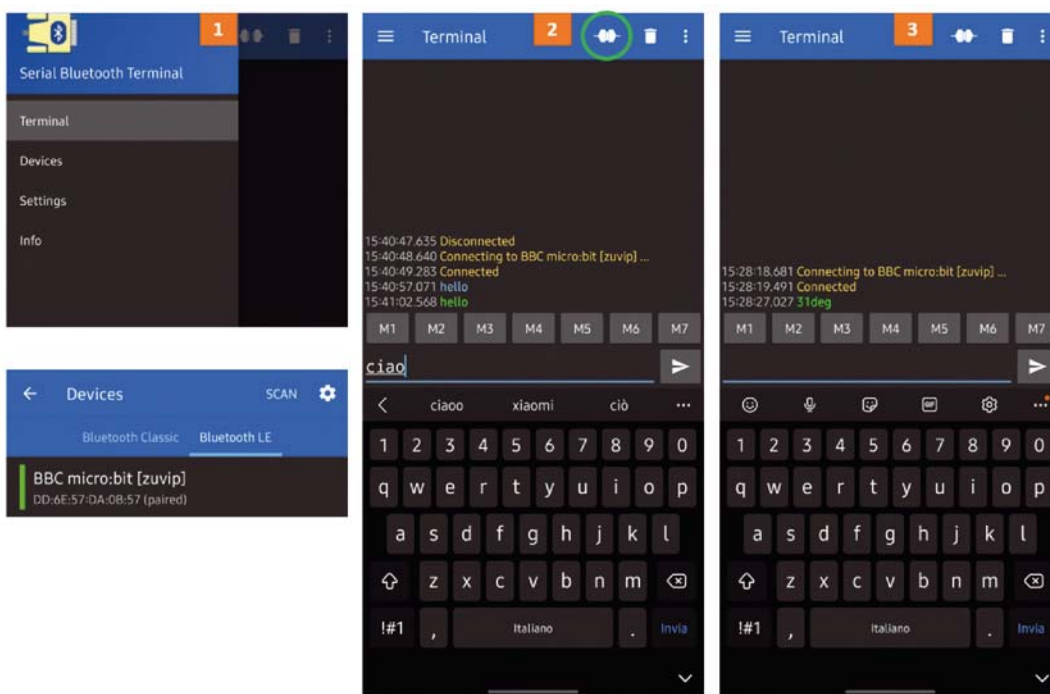


Fig. 15
Android
Serial
Bluetooth
Terminal.

in **Fig. 15** Punto 1; qualora non lo fosse, sarà necessario eseguire la procedura di accoppiamento descritta in **Fig. 9**. Selezionando il dispositivo, si aprirà il terminale, e tramite il tasto cerchiato in verde (**Fig. 15** Punto 2) sarà possibile avviare la comunicazione così da verificare le funzionalità implementate. La parola digitata verrà mostrata su display e re-inviata dal Microbit sul vostro smartphone (**Fig. 15** Punto 2) mentre alla pressione del logo touch, la temperatura dell'area circostante verrà visualizzata su display e trasmessa via Bluetooth (**Fig. 15** Punto 3).

Con quest'ultimo step, l'attività di verifica del progetto è completa. Abbiamo evidenziato la modularità e il semplice utilizzo della tool-chain, prendendo in esame funzionalità molto ricercate anche in ambito IoT.

CONCLUSIONI

Come visto ci siamo focalizzati sulla valutazione del prodotto Microbit e dei principali software oggi disponibili che forniscono all'utente gli strumenti necessari per la realizzazione dell'applicazione desiderata. Abbiamo confrontato le varie versioni rilasciate ed è emerso un importante aggiornamento con il Microbit v2 che evidenzia prestazioni low-power, nuove periferiche ma anche funzionalità aggiuntive come il Bluetooth 5.1 che garantisce un'ottima integrazione con applicazioni IoT. Questo articolo ha messo in risalto un oggetto di

piccole dimensioni in grado di fornire performance applicative al di sopra delle schede presenti sul mercato grazie al cuore del sistema rappresentato dal chip Nordic nRF52833 e dal lavoro svolto dalla BBC nella realizzazione di un dispositivo dalla semplicità di configurazione e programmazione disarmante.

Il chip Nordic nRF52833 ha mostrato tutte le sue potenzialità integrative, che consentono la coesistenza di molteplici periferiche grazie agli innumerevoli protocolli di comunicazione supportati e soprattutto il suo marchio di fabbrica rappresentato dalle tecnologie wireless ultra-low power che trovano grandi applicabilità nel campo della IoT. ■



Cosa occorre?

La board micro:bit V2 è disponibile con lo starterkit (cod. MICROBITQSKIT) a Euro 39,90.
I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

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

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.



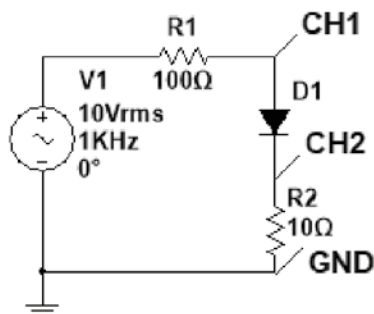
DSO225: AGGIORNIAMO IL FIRMWARE

dell'Ing. MIRCO SEGATELLO

Presentiamo un importante aggiornamento che amplia le capacità del nostro oscilloscopio DIY con le funzioni FFT, X/Y e la misura della potenza.

Nel numero 250 della rivista avevamo presentato un oscilloscopio palmare dotato di touch screen adatto per misure "al volo" in campo hobbistico; con una banda passante di circa 1MHz e frequenza di campionamento di 2,5Msps era facile eseguire misure su un circuito con Arduino ed in generale in banda audio. A fine articolo ci siamo lasciati con la promessa di implementare nuove funzioni al fine di ampliarne il campo di utilizzo e siccome ogni promessa è debito, eccoci qui a descrivere le nuove funzioni implementate e quali utilità possono avere. Diciamo subito che l'aver usato un display touch per interagire con l'oscilloscopio ci ha permesso di slegarci dall'hardware, potendo in questo modo modificare l'interfaccia utente semplicemente modificando il codice sorgente, possiamo quindi aggiungere e togliere pulsanti a

➔ **Fig. 1**
Schema elettrico
per il test di un
diodo.



← **Fig. 2**
Funzione X/Y
usata per
il test di un
diodo.

nostro piacimento, l'unico limite è la capacità di memoria del microcontrollore.

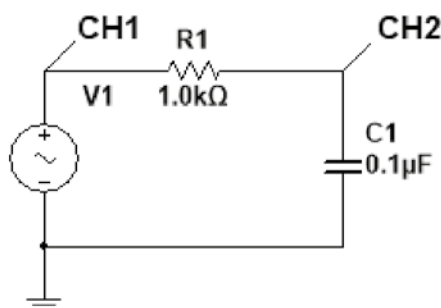
FUNZIONE X/Y

Questa funzione può essere implementata solamente in oscilloscopi a due canali in quanto è un modo alternativo di rappresentare le due tracce sullo schermo. In condizioni normali le due tracce, quelle del canale 1 e 2, vengono rappresentate separatamente e visualizzate in funzione del tempo secondo la base dei tempi impostata. Nella modalità X/Y invece, il canale 1 viene usato per modulare la traccia rispetto l'asse orizzontale, mentre il canale 2 viene usato per modulare la

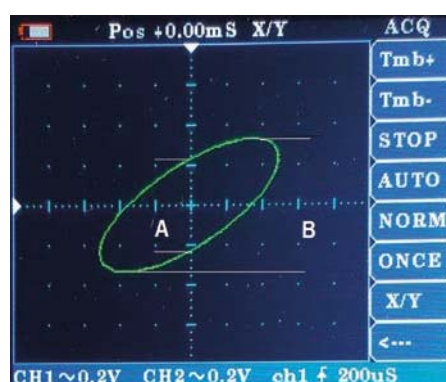
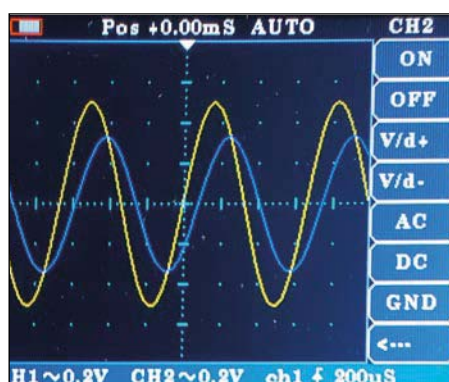
traccia rispetto l'asse verticale, è quindi possibile rappresentare una tensione in funzione di un'altra tensione, modalità solitamente usata per il test dei componenti elettronici. Se volessimo ad esempio testare la funzionalità di un diodo, potremmo collegarlo come in **Fig. 1** ed alimentare il tutto con un generatore di funzioni impostato per generare un'onda triangolare di ampiezza 10Vpp e offset 3V. Dopo aver aggiustato adeguatamente la sensibilità dei due ingressi vedrete sul display (**Fig. 2**) la classica curva di risposta V/I con in evidenza il punto in cui il diodo comincia a condurre, comunemente chiamato ginocchio. Un'altra funzionalità possibile in modalità X/Y è la misura dello sfasamento tra due segnali sinusoidali sfruttando le figure di Lissajous.

Nel circuito riportato in **Fig. 3** i canali dell'oscilloscopio sono connessi in ingresso ed in uscita di un quadripolo alimentato in regime sinusoidale. In **Fig. 4** le due tracce visualizzate nel modo normale mentre in **Fig. 5** il display in modalità X/Y. Se entrambi i canali saranno impostati con la stessa scala si vedrà un'ellisse e tramite i marker si potranno misurare le ampiezze A e B (**Fig. 5**)

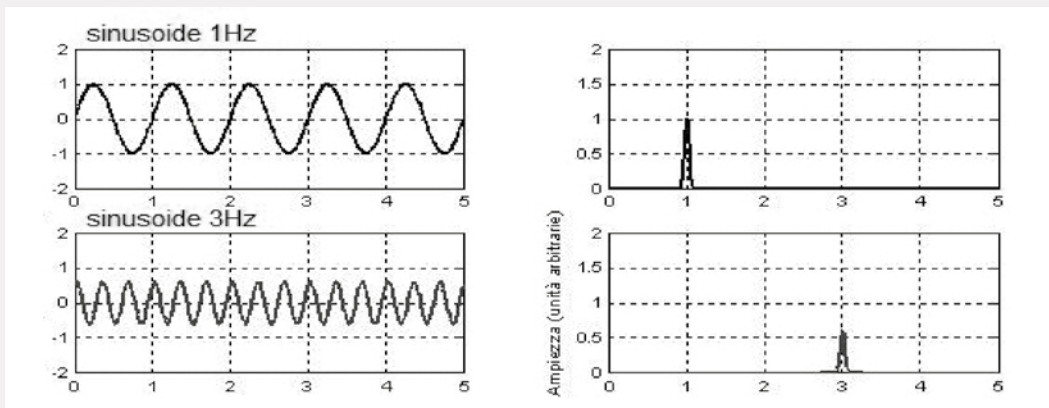
➔ **Fig. 3**
Collegamenti
elettrici per la
misura in un filtro
RC.



➔ **Fig. 4**
Visualizzazione
delle due tracce in
funzione del tempo



← **Fig. 5**
Visualizza-
zione delle
tracce in
modalità X/Y.



← **Fig. 6**
Spettro di
un segnale
sinusoidale.

dalle quali potrà essere ricavato lo sfasamento con l'equazione:

$$\text{sen}(\phi) = A/B$$

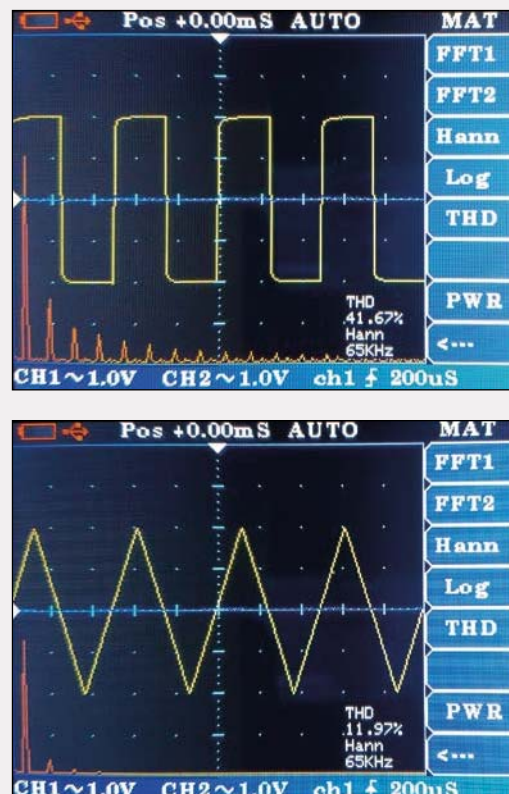
FUNZIONE FFT

La modalità X/Y pur essendo poco nota è senz'altro di facile comprensione, mentre la funzione FFT è decisamente più complessa da comprendere. Innanzi tutto il termine FFT (acronimo di Fast Fourier Transform) indica solamente l'utilizzo di un algoritmo di calcolo veloce per la determinazione della trasformata di Fourier; ma cos'è questa trasformata? Matematicamente parlando dovremmo dire che questo operatore permette di trasformare una funzione in un'altra, nel nostro caso da una funzione nel tempo ad una funzione in frequenza. In pratica significa che la trasformata fornisce informazioni dirette sul contenuto in frequenza di un segnale che in gergo tecnico si chiama spettro del segnale, infatti la scala orizzontale ha come unità di misura gli hertz (Hz). Per comprendere meglio questo concetto riferiamoci ad un'armonica a bocca strumento nel quale il suono è prodotto dal passaggio dell'aria soffiata sulle ance, sottili lamine metalliche, che entrano in vibrazione ciascuna ad una specifica nota.

A seconda di quali e quante ance entrano in vibrazione si ottengono suoni diversi; l'onda sonora risultante rappresentata in funzione del tempo sarebbe poco significativa, mentre lo spettro del segnale metterebbe in evidenza quali note vengono emesse. Nel caso di un segnale periodico lo spettro è composto da righe spettrali chiamate armoniche, più è distorto più armoniche (e di maggiore ampiezza) saranno presenti nello spettro. In **Fig. 6** potete vedere un segnale sinusoidale alla frequenza di 1Hz raffigurato in funzione del tempo (a sinistra) ed il relativo spettro (a destra) in cui è

evidente che esso è costituito da un unico segnale alla frequenza di 1Hz. La stessa figura riporta anche il caso di una sinusoide alla frequenza di 3Hz ed il relativo spettro con l'unica armonica posta in corrispondenza dei 3Hz.

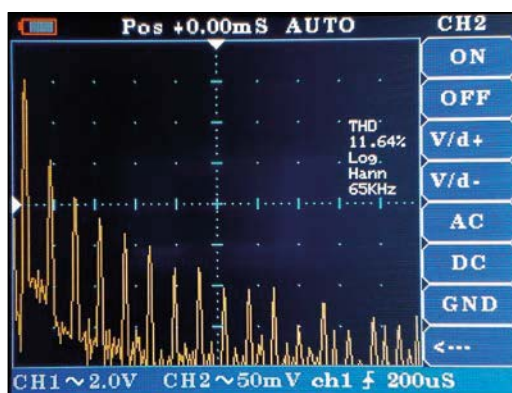
La funzione è attivabile dal menù MATH scegliendo FFT1 o FFT2 a seconda se si vuole utilizzare il canale 1 o 2. La scala delle ampiezze è fissa e coincide con la scala delle ampiezze del relativo canale, la scala delle frequenze parte da zero sino ad arrivare al valore indicato a destra ed è legato alla base dei tempi. In accordo con il teorema del



← **Fig. 7**
Spettro di un
segnale ad
onda quadra.

← **Fig. 8**
Spettro di un
segnale
triangolare.

Fig. 9
Utilizzo della scala
logaritmica.



campionamento lo spettro risulta corretto solo se si campiona ad una frequenza almeno doppia della frequenza dell'armonica a più alta frequenza. In Fig. 7 è riportata la schermata dell'oscilloscopio relativamente alla misura di un'onda quadra alla frequenza di 2 kHz, in cui viene rappresentato il segnale in funzione del tempo ed il relativo spettro che in questo caso è composto da una riga alla frequenza di 2 kHz e da una serie di armoniche di ampiezza decrescente. Analogamente in Fig. 8 con un segnale triangolare che, essendo più simile ad un'onda sinusoidale, presenta una quantità minore di armoniche. A destra del display il valore di 65 kHz indica che la scala delle frequenze va da 0 a 65 kHz (6,5 kHz per divisione) e che la frequenza di campionamento è il doppio, ovvero 130 kHz. La trasformata di Fourier prevede che il campionamento abbia durata infinita; se invece, come avviene in pratica, la durata del campionamento è limitata, diviene necessario modulare i dati campionati (funzione finestra) prima di eseguirne la trasformata. Nel nostro caso è possibile attivare la funzione Hann (dal matematico Hanning) che esegue questo calcolo. I valori forniti dalla trasformata di Fourier sono anche fondamentali per la misura della distorsione di un segnale, pertanto abbiamo introdotto la funzione THD (Total Harmonic Distortion) definita dall'equazione:

$$THD = \frac{\sqrt{V_2^2 + V_3^2 + \dots}}{V_1}$$

nella quale V_1 è l'ampiezza dell'armonica principale (che ha la stessa frequenza del segnale di partenza), mentre $V_2, V_3, \dots$ sono le ampiezze delle armoniche successive. Il limitato numero di bit del convertitore ADC abbinato ad uno stadio di ingresso non particolarmente raffinato e la limitata capacità di calcolo in virgola mobile non ci

permettono ovviamente di ottenere prestazioni professionali, pertanto non sarà possibile misurare distorsioni al di sotto di 0,5%. Per ultimo abbiamo implementato la visualizzazione in scala logaritmica delle ampiezze che è utile per mettere in evidenza le armoniche di minore intensità, infatti già con un'onda triangolare risultano visibili non più di 2-3 armoniche, mentre con la funzione logaritmica (Fig. 9) si mettono in evidenza molte più armoniche. La scala delle ampiezze è quindi in dB con un valore fisso di 20 dB per ogni divisione per un totale di 80 dB che è circa la massima dinamica disponibile con un ADC a 12bit.

FUNZIONE PWR

Si tratta della misura di potenza in regime sinusoidale anche in presenza di un carico complesso, ovvero un carico che contempli anche una componente reattiva. È il caso di un carico che oltre ad una resistenza presenti una componente induttiva o capacitiva, in questo caso tensione e corrente sul circuito risultano sfasate e oltre ad una potenza attiva risulta presente una potenza reattiva. Si definiscono pertanto tre tipi di potenza: potenza attiva (P), potenza reattiva (Q) e potenza apparente (S). Esse vengono rappresentate nel triangolo delle potenze (Fig. 10) e definite dalle equazioni:

$$P = V \cdot I \cdot \cos(\varphi)$$

$$Q = V \cdot I \cdot \sin(\varphi)$$

$$S = V \cdot I$$

La potenza attiva è legata alla reale energia utilizzata ed è anche quella che paghiamo al nostro fornitore (in realtà il contatore misura la potenza entro un $\cos\varphi$ di $0,95 \div 1$), mentre la potenza reattiva non fornisce contributo al lavoro svolto, per questo motivo in presenza di elevate potenze, questo valore deve essere quanto più possibile contenuto. Per essere più esatti, la potenza attiva in un utilizzatore è quella realmente dissipata o convertita

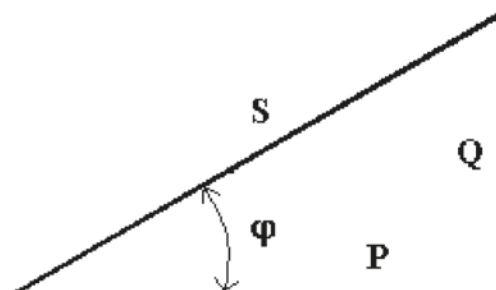


Fig. 10
Triangolo delle
potenze.

in altra forma (per esempio in energia meccanica) mentre quella reattiva è una potenza "palleggiante" ossia che viene momentaneamente impegnata e poi restituita dalle componenti reattive degli utilizzatori. Tale potenza può creare non pochi problemi, sia perché è concomitante a uno sfasamento anche rilevante tra corrente e tensione, sia perché determina picchi di assorbimento tali da sovraccaricare le linee, tanto più marcati quanto minore diviene il $\cos(\varphi)$; quando viene rilasciata, peraltro, può dare luogo a fenomeni transitori indesiderati e sovratensioni sulla linea elettrica. Dopo questa parentesi, torniamo al nostro strumento: la tensione può essere direttamente misurata con un canale dell'oscilloscopio mentre la corrente viene misurata tramite la caduta di tensione ai capi di una resistenza che svolge la funzione di shunt, nel nostro caso di valore 0,1ohm. Potevamo utilizzare sistemi più raffinati come trasformatori di corrente o sensori ad effetto hall ma una comunissima resistenza è senz'altro la soluzione più economica e ci assicura un'elevata banda passante tanto da poter utilizzare questo sistema anche per frequenze maggiori dei 50Hz. Di seguito riportiamo le equazioni che ci hanno permesso di determinare i valori sopra elencati in cui $V[i]$ e $I[i]$ sono i valori campionati di tensione e corrente e N è il numero totale di campioni.

$$V_{eff} = \sqrt{\frac{1}{N} \sum_1^N V[i]^2}$$

$$I_{eff} = \sqrt{\frac{1}{N} \sum_1^N I[i]^2}$$

$$P = \frac{1}{N} \sum_1^N V[i] \cdot I[i]$$

$$S = V_{eff} \cdot I_{eff}$$

$$Q = \sqrt{S^2 - P^2}$$

$$\cos(\varphi) = \frac{P}{S}$$

In **Fig. 11** potete vedere la misura eseguita su un bipolo formato dalla serie di una resistenza da 8ohm ed un condensatore da 22μF alimentati a 1 kHz tramite un amplificatore di potenza e connessi come visibile in **Fig. 12**. Potete usare questo sistema per valutare la reale

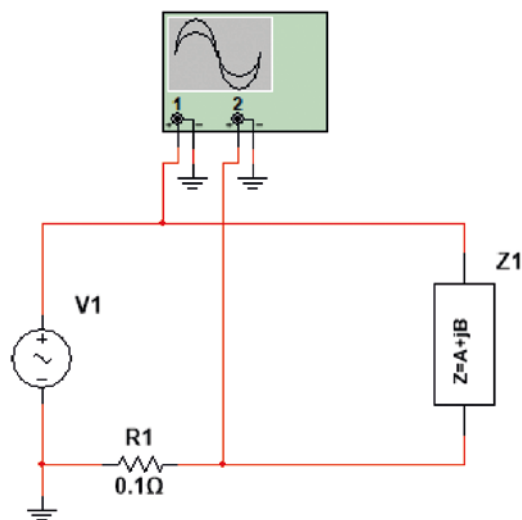


Fig. 11
Cablaggio
per la misura
di potenza.

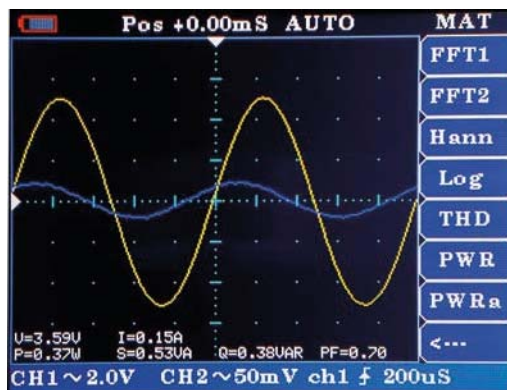


Fig. 12
Misura di
potenza.

potenza fornita dal vostro amplificatore alle casse acustiche ad una specifica frequenza.

FUNZIONE PWRa

Avendo a disposizione ancora un po' di memoria abbiamo implementato la misura di potenza di picco. Ci sono essenzialmente due casi in cui può essere utile questo tipo di misura: un primo caso si ha quando l'alimentazione è fissa ma il carico può variare, come ad esempio un motore all'avvio, che presenta una corrente di spunto assai più elevata della corrente assorbita a regime, un secondo caso potrebbe essere la necessità di misurare la tensione e la corrente di picco fornite in uscita da un amplificatore audio con ingresso un brano musicale. Come sapete in un brano musicale il livello varia in continuazione, così come l'impedenza di una cassa acustica che è diversa a seconda di quale frequenza è interessata. In questa situazione quando verrà avviata la funzione PWRa (e finché non verrà rimossa) verranno visualizzati a video i valori istan-

Scheda: "Generic STM32F103C series" >
 Variant: "STM32F103CB (20k RAM. 128k Flash)" >
 Upload method: "STM32duino bootloader" >
 CPU Speed(MHz): "72Mhz (Normal)" >
 Optimize: "Smallest (default)" >
 Porta: "COM11 (Maple Mini)" >
 Acquisisci informazioni sulla scheda

Fig. 13
Impostazioni
di Arduino.

tanei di picco di tensione e corrente ed il valore in potenza calcolato come semplice prodotto di tensione e corrente denominato potenza di picco. Una simile indicazione viene di solito fornita dai VU meters che però non tengono conto del reale assorbimento di corrente ma valutano solo il livello di tensione, inoltre la misura presenta una bassa velocità di risposta e non permette di rilevare il reale valore di picco.

Attivando la funzione per un intero brano si potrà valutare la massima tensione e corrente erogati dall'amplificatore e capire quanto questo sia stato effettivamente sfruttato. Si tratta di valori massimi erogati per un breve istante che non influiscono sul riscaldamento termico dell'amplificatore ma possono far saltare le protezioni da corto circuito oppure danneggiare i transistor finali.

PROCEDURA DI PROGRAMMAZIONE

In questa versione del firmware abbiamo abilitato il trasferimento dei dati acquisiti verso il PC, pertanto è necessario che quest'ultimo riconosca il nostro oscilloscopio come periferica esterna. Per fare questo è necessario installare sul microcontrollore il bootloader similmente a quanto avviene in una scheda Arduino. Per installare il bootloader bisogna connettere un convertitore USB/SERIALE a 3,3V come indicato nella **Tabella 1** sfruttando l'apposito connettore sul PCB. Impostate i jumper BOOT0 (quello esterno) su 1 e BOOT1 (quello in-

terno) su 0 ed alimentare la scheda (oppure se già alimentata premere il pulsante di reset). Avviate il software della ST Flash Loader e aprite il file generic_boot20_pc13.bin fornito assieme ai file della rivista, quindi procedete alla programmazione; per i dettagli si faccia riferimento al precedente articolo. Scollegate il programmatore e riposizionate il jumper BOOT0 su 0. Da questo momento sarà possibile la programmazione direttamente da USB con il software Arduino, prima però è necessario installare il pacchetto aggiuntivo per dispositivi STM32. Recatevi sul web all'indirizzo https://github.com/rogerclarkmelbourne/Arduino_STM32 e scaricate l'intero pacchetto Arduino_STM32-master, scompattatelo e copiatelo nella cartella hardware del software Arduino. Avviate l'IDE di Arduino e verificate che venga riconosciuto sia il pacchetto appena installato che la nuova periferica. Quest'ultima viene riconosciuta con il nome di Maple Mini, se così non fosse installate anche i driver disponibili assieme al pacchetto appena scaricato. Aprite lo sketch del progetto DSO225 e dal menu Impostazioni configurate Arduino come visibile in **Fig. 13**, quindi procedete alla programmazione. Notate che non è necessario installare alcuna libreria, in quanto quelle occorrenti sono già disponibili nel pacchetto STM32 oppure sono comprese nello sketch.

CONCLUSIONI

Per il momento ci fermiamo qui, ma essendo quello del nostro oscilloscopio DIY un progetto work in progress, nulla vieta che in futuro si apportino migliorie o si aggiungano ulteriori funzioni; nel caso ve ne daremo conto in un ulteriore articolo. Per il momento non ci resta che augurarvi buon lavoro e un uso proficuo dell'oscilloscopio. □

Tabella 1
Connessioni tra modulo e interfaccia seriale-USB.

FDT13V3	Bluepill
3V3	3V3
GND	GND
TX	A10
RX	A9



Cosa occorre?

I componenti utilizzati in questo progetto sono di facile reperibilità. Il cuore del progetto, il modulo con STM32 (cod. STM32ARMCORE), è in vendita a Euro 14,80, il display touch da 2,8" (cod. LCDTOUCH28SER) è disponibile a Euro 22,90, la batteria litio da 1100 mA (cod. PLIB1100) costa Euro 7,90. Il circuito stampato può essere realizzato utilizzando il servizio di prototipazione dei PCB (cod. PCBPRODUCTION). 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

€ 319,00



Disponibile anche
nella versione a 200MHz

cod. DS0202

€ 459,00

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



cod. DS0FG102

€ 429,00

- ✓ 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

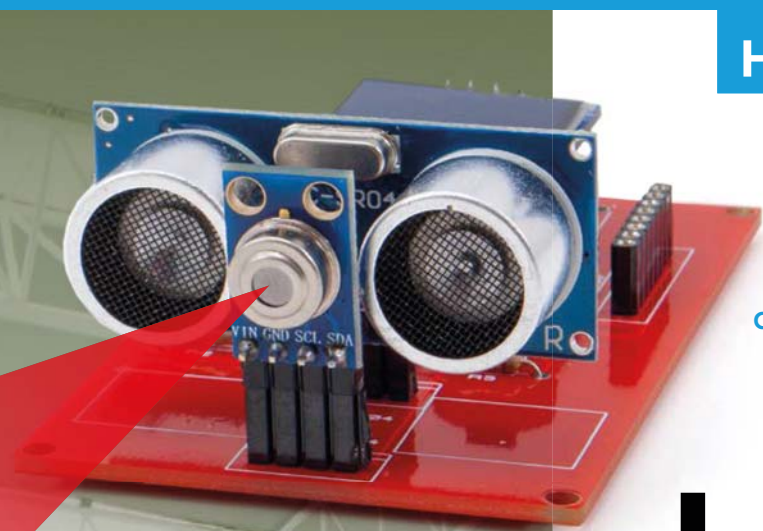
Disponibile anche nella versione
a 2 canali 70 MHz con generatore
di forme d'onda e multimetro

cod. MH02D72GEN

€ 268,00

MONITOR TEMPERATURA CON RFID

dell'ing. FULVIO DE SANTIS



Implementiamo un sistema di monitoraggio della temperatura senza contatto e di identificazione RFID sfruttando il modulo uCHIP.

Uno degli elementi strategici di ogni nazione è la salute della popolazione e in quest'ottica si colloca lo screening precoce delle malattie più impattanti. Specialmente nelle aree socialmente svantaggiate, i potenziali pazienti potrebbero non avere un accesso tempestivo a tale screening, quindi possono essere utili soluzioni che consentono l'auto-monitoraggio. Se queste prevedono l'utilizzo di sensori che promuovono una modifica positiva del comportamento umano, come ad esempio i sensori nei dispositivi indossabili di monitoraggio sanitario, possono derivarne altri vantaggi. La raccolta di dati da più soggetti per lo screening e la ricerca dei contatti può essere un potenziale utilizzo nella pandemia COVID-19 in corso, perché i termometri contactless a infrarossi vengono utilizzati come strumento di screening per monitorare le persone negli aeroporti, nelle stazioni ferroviarie e in altri luoghi affollati per identificare potenziali pazienti Covid-19.

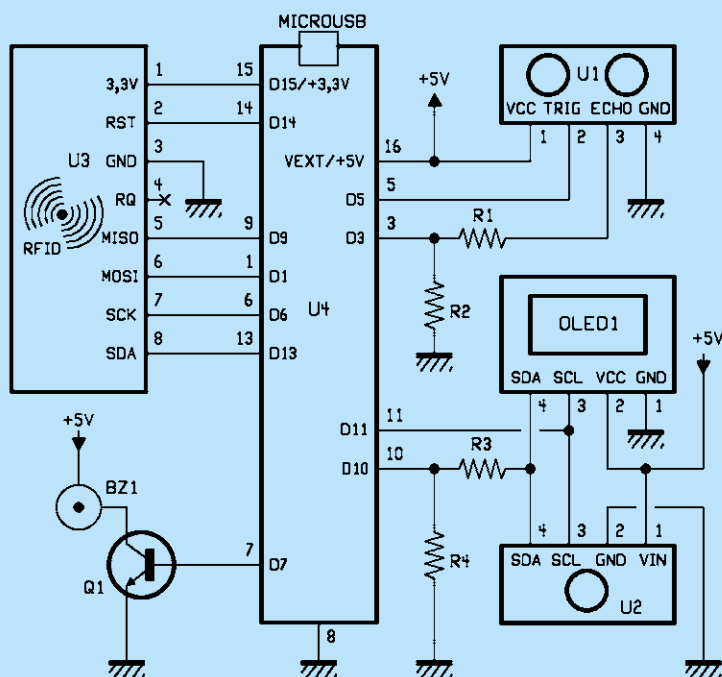
Lo screening della temperatura è stato ed è un punto focale nel rilevamento di casi sospetti, poiché la febbre è una delle tipiche manifestazioni iniziali della malattia.

In questo articolo proponiamo un sistema di monitoraggio della temperatura e di identificazione dei soggetti sottoposti allo screening mediante l'impiego della tecnologia RFID.

IL NOSTRO PROGETTO

Il sistema di monitoraggio della temperatura senza contatto e di identificazione mediante RFID si basa sul sensore di temperatura senza contatto MLX90614, un display OLED SSD1306 per la visualizzazione dei valori della temperatura ambiente e della temperatura corporea, un modulo lettore RFID RC522 con chip MFRC522 integrato, il sensore a ultrasuoni HC-SR04 e il modulo uChip; il tutto è descritto dallo schema elettrico visibile in queste pagine.

Il sistema funziona con persone che indossano un tag RFID ed è quindi destinato, ad esempio, ai dipendenti di un'azienda o un ente in possesso di badge RFID. Quando una persona si avvicina, mediante il sensore a ultrasuoni viene calcolata la distanza e quando questa è minore o uguale a quella necessaria per rilevare la temperatura, inizia l'interrogazione del tag. Una volta che la scheda viene avvicinata al modulo lettore RFID, il lettore acquisisce i dati identificativi della persona e li salva nella memoria del sistema. Successivamente, i dati acquisiti del tag scansionato vengono confrontati con l'UID



predefinito nel codice. Se l'UID corrisponde al tag scansionato, viene effettuata la misura della temperatura della persona identificata e il valore di temperatura e il nome della persona vengono registrati, come vedremo, in un foglio Excel.

IL MODULO RFID

Il modulo lettore RFID RC522 riportato in **Fig. 1** è basato sull'integrato MFRC522, il quale dispone di un ricetrasmittitore che irradia un segnale RF alla frequenza di 13,56 MHz. Quando la card tag RFID entra nel raggio del segnale del trasmettitore dell'RC522, questo segnale colpisce il transponder che si trova all'interno della card tag. Il tag trae l'alimentazione dal campo elettromagnetico generato dal lettore e trasferisce tutte le informazioni al modulo lettore RFID, quindi anche l'UID

specifico del soggetto, mediante un segnale a radio frequenza. Questi dati vengono successivamente inviati al microcontrollore uChip utilizzando la comunicazione seriale SPI.

Il chip MFRC522 supporta le card MIFARE e NTAG ISO/IEC 14443 A; il trasmettitore dell'MFRC522 è in grado di pilotare un'antenna progettata per comunicare con card e transponder ISO/IEC 14443 A/MIFARE e il ricevitore demodula e decodifica i segnali del transponder. Il modulo digitale integrato gestisce l'intero frame ISO/IEC 14443 A e le funzionalità di rilevamento degli errori (parità e CRC). L'MFRC522 supporta velocità di trasferimento fino a 848 kBaud in entrambe le direzioni. Seguendo da sinistra lo schema a blocchi dell'MFRC522 di **Fig. 2**, l'interfaccia analogica del chip gestisce la modulazione e la demodulazione dei segnali analogici. L'UART contactless attraverso il REGISTER BANK gestisce i requisiti del protocollo di comunicazione richiesti dal dispositivo esterno (host). Il buffer FIFO garantisce il trasferimento veloce dei dati da e verso l'UART contactless di cui vengono rese disponibili varie interfacce di comunicazione seriale.

Queste le specifiche tecniche del modulo RFID:

- tensione di alimentazione di 3,3 Vcc;
- corrente assorbita in operatività = 13÷26 mA;
- corrente di stand-by <80 uA;
- interfacce di comunicazione SPI fino a 10 Mbit/s, UART seriale RS232 fino a 1.228,8 kBaud, I²C Bus fino a 400 kBaud in modalità Fast, fino a 3.400 kBaud in modalità High-speed;
- dimensioni: 60x40x6 mm.

IL SENSORE DI TEMPERATURA IR

Un importante componente del progetto è il sensore MLX90614, prodotto da Melexis Microelectronics Integrated Systems, che sfruttando gli infrarossi passivi (ossia quelli irradiati da un corpo caldo, purché sopra lo zero assoluto) rileva la temperatura degli oggetti senza entrarvi in contatto. Il sensore MLX90614 integra due dispositivi: l'unità di rilevamento termopila a infrarossi e l'unità di calcolo costituita da un DSP per il condizionamento del segnale. L'unità rilevatrice del sensore misura la quantità di energia IR emessa da un oggetto e l'unità di calcolo la converte in un valore di temperatura utilizzando un convertitore ADC integrato ad alta risoluzione a 17 bit che invia i dati tramite il protocollo di comunicazione I²C. Il sensore misura sia la temperatura dell'oggetto che la temperatura ambiente che viene utilizzata per calibrare internamente al sensore il valore esatto della tempera-

Fig. 1
Il modulo
RC522.



tura dell'oggetto. Il sensore MLX90614 è dotato di uscita digitale PWM e del sistema di comunicazione SMBus (System Management Bus). Da impostazione standard, il generatore PWM a 10 bit è configurato per trasmettere continuamente il valore di temperatura nell'intervallo da -20 a 120 °C, con una risoluzione di 0,14 °C. L'SMBus è un bus a due fili simile all'I²C, usato per la comunicazione seriale a bassa velocità con periferiche quali controller DC/DC, sottosistemi per la gestione delle batterie ricaricabili, sensori di temperatura. Un diodo zener a 5,7V è disponibile al pin SCL per il collegamento di un alimentatore esterno da 8 a 16V per alimentare il dispositivo MLX90614Axx (versione a 5V). Il pin PWM/SDA è un ingresso/uscita digitale. Utilizzato come PWM fornisce in uscita un segnale PWM con il valore della temperatura dell'oggetto sotto test; come SDA, il pin viene utilizzato nelle comunicazioni compatibili SMBus. In modalità SMBus il pin SDA è automaticamente configurato come open drain NMOS. L'MLX90614 comunica con il microcontrollore uChip mediante il protocollo I²C e necessita della libreria **Adafruit_MLX90614** per la sua gestione.

L'SMBUS DELL'MLX90614

Lo scopo originale dell'SMBus era di definire il collegamento di comunicazione intelligente tra una batteria, caricabatteria e un microcontrollore in comunicazione con il resto del sistema. Tuttavia, l'SMBus può anche essere utilizzato per collegare un'ampia varietà di dispositivi, inclusi i dispositivi relativi all'alimentazione, sensori, dispositivi di comunicazione, EEPROM di archiviazione e altro ancora. Sono necessarie solo due linee di bus; una linea dati seriale (SDA) e una linea seriale di clock (SCL). Ogni dispositivo connesso al bus è indirizzabile via software tramite un indirizzo univoco. SMBus è un bus multi-master che include il rilevamento delle collisioni e l'arbitrato per prevenire il danneggiamento dei dati se due o più master avviano simultaneamente il trasferimento dei dati. Di seguito, le specifiche del sensore MLX90614:

- Temperatura rilevabile: -70÷380°C;
- Precisione:
 - da -20°C a +90°C: ±0,5°C;
 - da +90°C a +150°C: ±0,625°C;
 - da -50°C a -20°C: ±0,7°C.
- Tensione di alimentazione: 3Vcc per MLX90614Bxx o 5Vcc per MLX90614Axx.
- Massima corrente assorbita: 2,5 mA.
- Dimensioni: 16,9x11,3x7,6 mm.

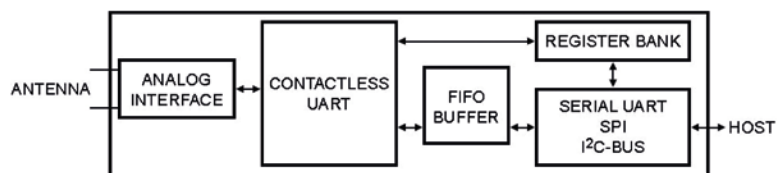


Fig. 2
Schema a blocchi
del chip MFRC522.

In **Fig. 3** è proposto l'integrato MLX90614. Ma vediamo in pratica come funziona il sensore MLX90614. Come abbiamo già detto, ci sono molti sensori di temperatura disponibili sul mercato utilizzati ampiamente per la misura della temperatura. Nel caso di questo progetto, per misurare la temperatura corporea senza entrare direttamente in contatto con il soggetto da monitorare, l'MLX90614 soddisfa questo requisito utilizzando l'energia IR emessa per rilevare la temperatura. Il sensore misura la temperatura di un oggetto e la temperatura ambiente che viene utilizzata per calibrare il valore della temperatura dell'oggetto. Ma quale dovrebbe essere la distanza tra il sensore e l'oggetto sotto test per ottenere il corretto valore di temperatura? Il valore di questa distanza si ricava dal "Field of View" (FOV), che per il nostro sensore MLX90614xAA il campo visivo è di circa 90° (+45°/-45°) per una sensibilità di misura del 50%. Immaginando sia conico il FOV del sensore, man mano che il sensore si allontana dall'oggetto si estende l'area di rilevamento ma la precisione di misurazione diminuisce. Come specificato nel datasheet, la precisione è garantita solo in condizioni isotermiche stabilizzate e se l'oggetto occupa completamente il FOV del sensore. Per maggiori dettagli fate riferimento al datasheet dell'MLX90614.

IL DISPLAY OLED SSD1306

Per visualizzare i valori della temperatura ambiente e di quella del soggetto da monitorare viene utilizzato il display OLED SSD1306, monocromatico da 0,96 pollici con risoluzione di 128x64 pixel. Il display OLED ha un consumo ridottissimo perché non richiede retroilluminazione e per questo lo abbiamo scelto, dato che il sistema funziona a batteria. Il display (**Fig. 4**) da noi usato ha quattro pin: GND, VCC, SCL, SDA, e comunica con il uChip utilizzando l'I²C-Bus facente capo agli ultimi due; esistono comunque versioni dotate di una linea di RESET e di interfaccia SPI. Per controllare il display utilizziamo le librerie Adafruit SSD1306 e Adafruit GFX.

IL SENSORE A ULTRASUONI

Il modulo a ultrasuoni HC-SR04 (**Fig. 5**) è un



→ **Fig. 3**
Il sensore di
temperatura
MLX90614.



senore misuratore di distanza senza contatto in grado di misurare distanze da 2 cm a 400 cm con una precisione di circa 3 mm.

Il modulo integra un trasmettitore a ultrasuoni, un ricevitore e circuiteria di controllo. Il funzionamento dell'HC-SR04 prevede l'invio al pin "Trigger" di un segnale a livello alto della durata di almeno 10 μ S a cui segue da parte del trasmettitore l'invio di un treno di otto impulsi alla frequenza di 40 kHz. Nel contempo il ricevitore rileva se è presente un segnale impulsivo di ritorno causato da un ostacolo colpito dagli impulsi inviati dal trasmettitore. Se il segnale di ritorno è di livello sufficiente, sul pin "Echo" viene generato un impulso di livello alto la cui larghezza è proporzionale al tempo trascorso dall'invio degli impulsi al loro ritorno. La distanza "d" si ricava dalla formula:

$$d = \text{durata del livello alto Echo} \times \text{velocità del suono} / 2$$

In **Fig. 6** viene riportato il diagramma di temporizzazione per il rilevamento della distanza con l'HC-SR04.

Di seguito, le specifiche tecniche del modulo.

- Tensione alimentazione: 5 V.
- Corrente assorbita: 15mA.
- Frequenza del segnale: 40 kHz.
- Distanza misurabile: 2 cm ÷ 4 m.
- Angolo di misura: 15°.
- Segnale di ingresso di trigger: Impulso TTL a livello alto per 10 μ S.
- Segnale di uscita di eco: Impulso TTL a livello alto per la durata in funzione della distanza dell'oggetto misurata, da 150 μ s (2 cm) a 23 ms (4 m). Se non viene rilevato alcun ostacolo entro i 4 m, l'impulso ha una durata di circa 38 ms.

→ **Fig. 4**
Il display OLED
SSD1306.

LA SCHEDA DI SVILUPPO UCHIP

Il modulo uChip è il gestore dei sensori del sistema di monitoraggio; si tratta di una scheda di sviluppo open source con microcontrollore SAMD21E18 Cortex M0 AT a 32 bit compatibile con Arduino (Arduino Zero ha la stessa MCU) che si adatta a un socket standard DIL a 16 pin a profilo stretto, requisito molto importante per la realizzazione di sistemi elettronici compatti. Il chip SAM D21 implementa la maggior parte delle funzioni di Arduino Zero, inclusa la funzionalità USB-OTG (USB On-The-Go), anche se alcune funzioni e pin potrebbero non essere implementati al 100% a causa del numero ridotto di pin. Come Arduino Zero, ha 256 kB di memoria flash e 32 kB di RAM.

Il modulo uChip contiene un LED (connesso internamente al pin D0) programmabile via software, un pulsante di programmazione e reset, un quarzo, una circuiteria per la gestione automatizzata delle alimentazioni, una circuiteria di protezione EMI/ESD, un connettore micro-USB per la programmazione, la comunicazione seriale e l'alimentazione, a cui è possibile associare la funzionalità OTG, ovvero, è possibile alimentare dispositivi esterni dal connettore USB stesso.

uChip contiene un sofisticato sistema di gestione delle alimentazioni che abbiamo già spiegato nel progetto di monitoraggio ambientale con uCHIP, pubblicato nel fascicolo n° 252. Il modulo uChip include due convertitori DC-DC: boost (convertitore elevatore di tensione o step-up) e buck (convertitore riduttore di tensione o step-down).

Il convertitore boost consente di fornire al connettore USB una tensione di 5 V con corrente erogabile fino a 500 mA quando viene applicata



una tensione esterna al pin VEXT (pin 16) compresa tra 3,3 V e 5 V.

Il convertitore buck fornisce al pin VEXT selezionabile tramite software, una tensione di 3,3 V o quella fornita dall'USB (normalmente 5 V) con corrente erogabile fino a 1 A; quando uChip è alimentato tramite USB da un PC, normalmente la corrente erogabile massima è di 500 mA, ma si può anche avere una corrente erogabile maggiore fino a 1 A alimentando uChip con un powerbank in grado di erogare 1 A. Quando viene selezionata la tensione da sorgente USB, la tensione di uscita VEXT effettiva dipenderà dalla tensione dell'USB (che potrebbe essere più bassa di 5 V) e dalla corrente di uscita (da cui la caduta di tensione è tipicamente $0,5\Omega \times I$, dove I è la corrente di uscita in ampere). Quando invece si seleziona 3,3 V, la tensione di uscita dipenderà solo dalla corrente.

Inoltre, applicando una tensione esterna al pin VEXT o alimentando da USB, mediante un ponticello è possibile ottenere al pin 15 di uChip una tensione regolata di 3,3 V con corrente erogabile fino a 100 mA. Un circuito ausiliario impedisce il flusso di corrente inversa tra la tensione esterna al pin 16 VEXT e l'alimentazione proveniente dall'USB, ovvero, quando l'alimentazione non è in modalità USB-OTG. Un circuito di supporto implementa anche la selezione automatica dell'alimentazione tra USB e la tensione esterna VEXT, e impedisce di avere alimentazioni contemporaneamente da due sorgenti, ossia da USB e VEXT.

Sebbene uChip accetti e possa fornire una tensione esterna fino a 5V al pin 16, tutti gli altri pin I/O non sopportano livelli di tensione a 5V ma accettano solo livelli digitali o analogici di 3,3V. Inoltre, non bisogna applicare una tensione superiore a 5,5 V sul pin 16, poiché danneggerebbe uChip anche in questo caso. Se collegate un carico induttivo (ad esempio servomotori) alla linea VEXT di uChip, ovvero, se è necessario alimentare un carico induttivo tramite il pin VEXT di uChip, o anche se

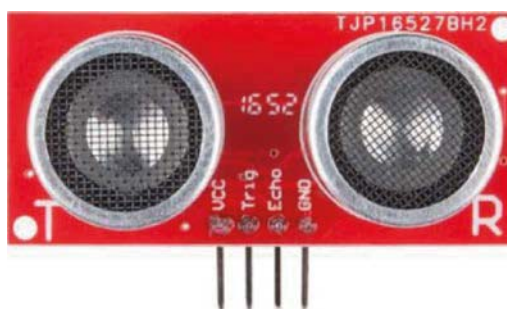


Fig. 5
Sensore di distanza a ultrasuoni HC-SR04.

il carico induttivo è alimentato esternamente, ma VEXT è collegato alla stessa fonte di alimentazione, prestate particolare attenzione ad eliminare l'insorgere di extra tensioni induttive, che potrebbero far salire VEXT oltre 5,5 V e distruggere uChip, prevedendo una idonea circuiteria di soppressione delle extratensioni induttive.

Nella **Fig. 7** viene illustrata la pin-out semplificata del modulo uChip nella quale dalle colonne più interne verso quelle più esterne, viene indicato:

- il numero di pin digitale/analogico (colonna più interna);
- il numero effettivo della porta del chip SAMD21 (questo è utile soprattutto se non si utilizza il Framework di Arduino);
- le funzionalità del particolare pin;
- la funzione predefinita dei pin (colonne più esterne).

Inoltre uChip dispone di 14 GPIO di cui 13 PWM, 8 ADC a 12 bit, 1 DAC a 10 bit, 5 Interfacce seriali (SPI, I²C, UART, I²S), 2 interfacce di debug SWD (pin 12, 13); 2 pin di alimentazione (VEXT, GND).

Ponticelli SC e 3V

uChip dispone di due ponticelli denominati SC e 3V. Il ponticello SC è in cortocircuito per impostazione predefinita e collega al pin 13 la porta PA30 che per default è il segnale di clock dell'interfaccia di debug SWD (Serial Wire Debug).

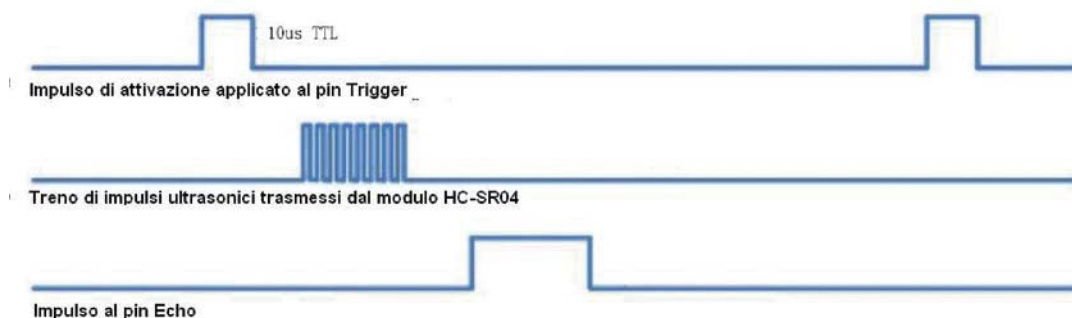


Fig. 6
Diagramma di temporizzazione per il rilevamento della distanza.

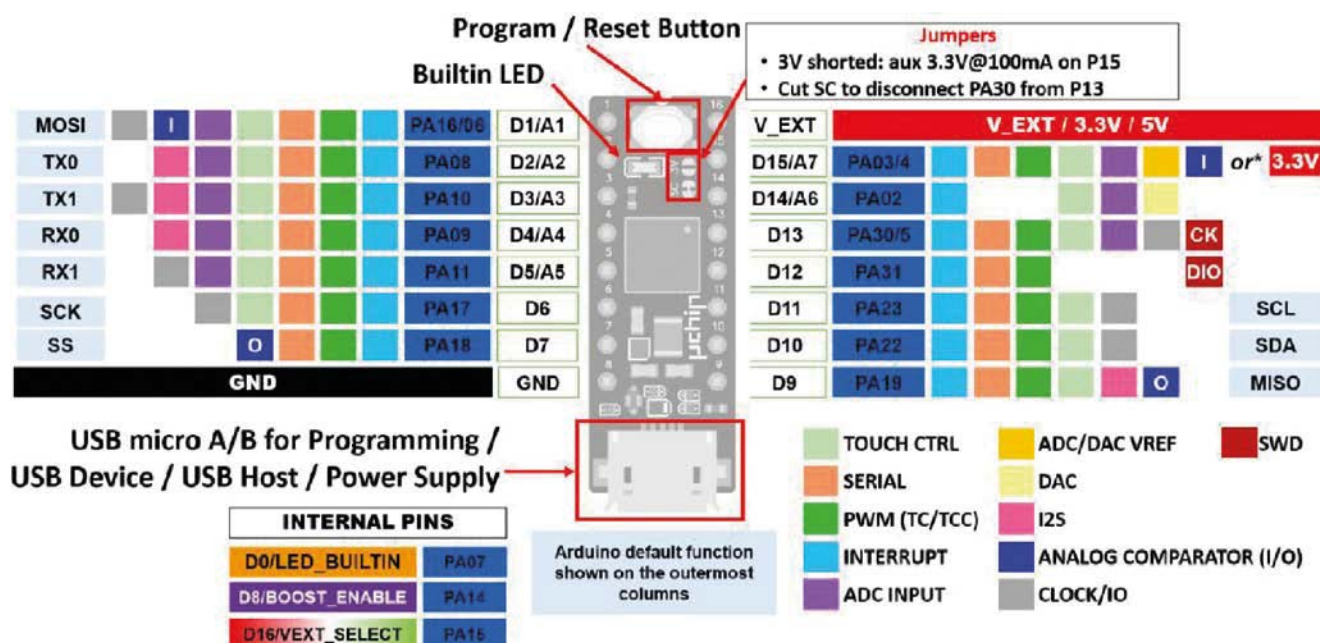


Fig. 7
Pin-out del
modulo
uChip.

Il pin 13 è collegato anche alla porta PA05 e se SC venisse interrotto, il pin 13 sarebbe ancora collegato alla porta PA05 ma si interromperebbe il collegamento pin 13-Porta30. Se l'applicazione non necessita di debug SWD e se il pin 13 restasse a livello basso anche durante il reset, si può interrompere il ponticello SC tagliando la traccia sul PCB. Se in seguito è necessario utilizzare PA30 o la funzione SWD, si dovrà cortocircuitare nuovamente il ponticello SC. Notate che il pin 13-PA30/05 è a livello alto di default e deve restarci durante il reset per consentire l'avvio del microcontrollore SAMD21, altrimenti rimarrà nello stato di reset. Ciò è dovuto all'implementazione SWD nel microcontrollore SAMD21. Cortocircuitando il ponticello 3V, aperto per impostazione predefinita, uChip è in grado di fornire all'esterno dall'uscita ausiliaria Pin 15, una tensione di alimentazione regolata di 3,3 V con corrente erogabile fino a 100 mA. In questo modo, uChip può fornire contemporaneamente sia 5V (dal pin 16) che 3,3V (dal pin 15). Se il ponticello 3V non viene cortocircuitato, il Pin 15 può essere utilizzato come un normale pin GPIO. L'uscita ausiliaria a 3,3 V (pin 15) è fornita da un regolatore lineare da 3,3 V collegato a VEXT (pin 16). Dato che il regolatore alimenta anche il SAMD21 e consente di erogare fino a 300 mA, è opportuno non assorbire più di 100 mA dall'uscita ausiliaria per evitare rischiose temperature di funzionamento elevate. Come evidenziato in **Fig. 8**, sulla scheda uChip sono indicate le piazzole dei ponticelli SC e 3V.

IL PULSANTE DI RESET

uChip include anche un pulsante a doppia funzione: una singola pressione resetta uChip, mentre due pressioni in rapida sequenza la portano in modalità programmazione; quest'ultima è utile se si desidera programmare non con l'IDE di Arduino ma ad esempio con Atmel Studio, oppure se lo sketch non risponde. Di seguito riportiamo le caratteristiche tecniche del modulo uCHIP.

- CPU: integrata della serie Cortex M0 + ATSAMD21 a 32 bit a 48 MHz (compatibile con Arduino Zero e varianti analoghe).
- Flash: 256 kB (248 kB per il bootloader).
- RAM: 32 kB;
- Alimentazione tramite USB o esterna da 3,3 a 5V.
- 14 pin di I/O.
- LED di stato: controllato via software con singola istruzione.
- 8 ingressi ADC a 12 bit
- 1 uscita DAC a 10 bit
- 14 pin di interrupt
- Interfacce seriali: SPI, I²C, UART
- Porta I²S per decoder audio ad esempio UDA1334A).
- 13 pin PWM.
- Compatibilità con pin-out di integrati CMOS: alimentazione e GND sono sui pin 16 e 8, quindi è anche possibile emulare alcuni circuiti integrati CMOS a 16 pin (serie 4000 e 74HC).
- Dimensioni: 28,5x10,16 mm.

LE PERIFERICHE DEL MODULO UCHIP

Riteniamo utile riportare una sintetica descrizione delle principali funzioni delle porte I/O del microcontrollore uChip, utili per espandere il progetto qui descritto: l'ADC, il DAC e le porte seriali.

Il convertitore analogico-digitale (ADC) della uChip ha una risoluzione di 12 bit ed è in grado di convertire con velocità fino a 350 ksps. L'ADC può essere abbinato ad uno stadio amplificatore per aumentare il range dinamico del segnale.

L'ADC può essere attivato dal software applicativo o da un evento causato da un'altra periferica, ma anche da temporizzazioni preimpostate senza l'intervento del software. È possibile utilizzare tensioni di riferimento sia interne che esterne. L'ADC dispone di un sensore di temperatura integrato per la compensazione termica. L'ADC può essere configurato per una risoluzione a 8, 10 o 12 bit. I risultati digitali della conversione ADC possono essere posizionati a sinistra o a destra della stringa binaria, il che facilita il calcolo quando il risultato è rappresentato come valore con segno. È possibile utilizzare la funzionalità DMA (accesso diretto alla memoria) per spostare i risultati dell'ADC direttamente nella memoria o nelle periferiche al termine delle conversioni. In **Fig. 9** viene riportato lo schema a blocchi funzionale dell'ADC. L'ADC ha un'opzione di sovracampionamento con decimazione dei bit che può estendere la risoluzione fino a 16 bit. I valori di input possono essere anche interni (ad es. proveniente dal sensore di temperatura interno) o esterni (dai pin I/O collegati a dispositivi esterni). L'utente può anche scegliere mediante configurazione di registri se la conversione debba essere single-ended o differenziale. Nella configurazione di base, l'ADC campiona i valori degli input provenienti da sorgenti interne o esterne configurate nel registro INPUTCTRL. La velocità di conversione dipende dalla frequenza (GCLK_ADCx) e dal prescaler del clock. Per convertire i valori analogici in valori digitali, è necessario inizializzare l'ADC. La conversione dell'ADC è sincronizzata dal GCLK_ADC che, mediante l'impiego di un prescaler nell'ADC, imposta la conversione a frequenze di clock inferiori. Il risultato della conversione viene memorizzato nel registro dei risultati RESULT sovrascrivendo il risultato della precedente conversione. Per evitare la perdita di dati quando è abilitato più di un canale ADC, il risultato della conversione deve essere letto non appena è disponibile (INTFLAG.RESRDY). In caso contrario si verificherà una condizione di errore di overrun, indicata dal bit OVERRUN. L'ADC ha due modalità di conver-

sione degli input: differenziale e single-ended. Se il potenziale dell'ingresso positivo può diventare minore di quello dell'ingresso negativo, è necessario utilizzare la modalità differenziale per ottenere risultati corretti. Se l'ingresso positivo è sempre positivo, è necessario utilizzare la conversione single-ended in modo da avere nella conversione una risoluzione a 12 bit pieni. L'ingresso negativo deve essere collegato a massa. Se l'ingresso positivo può scendere al di sotto dell'ingresso negativo creando alcuni risultati negativi, per ottenere risultati corretti dovrebbe essere utilizzata la modalità differenziale.

Convertitore digitale-analogico

Il DAC della uChip ha un canale con risoluzione a 10 bit ed è in grado di convertire fino a 350.000 campioni al secondo (350ksps).

Fig. 8
Ponticelli
SC e 3V.

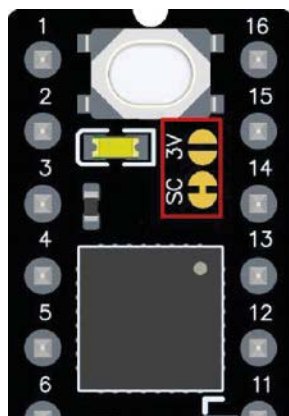
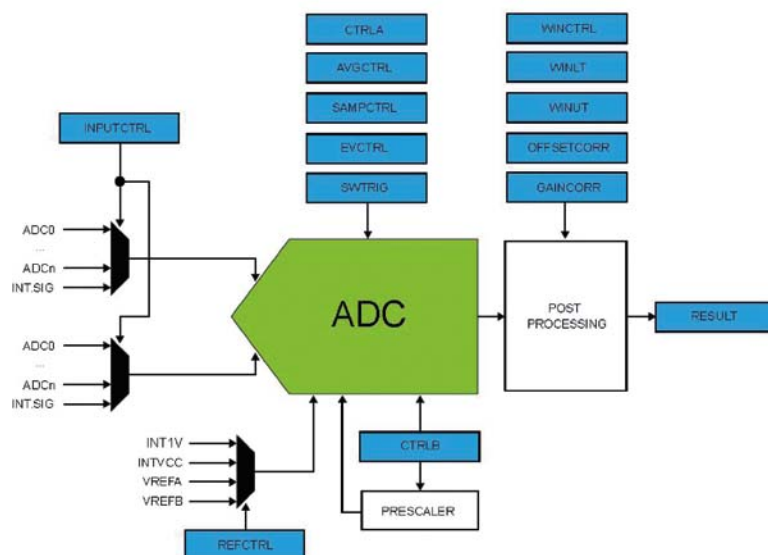


Fig. 9
Schema
a blocchi
funzionale
dell'ADC.



Il DAC converte il valore digitale d'ingresso situato nel registro dati DATA in una tensione analogica disponibile all'uscita DAC (VOUT). Quando vengono scritti nuovi dati nel registro dati, viene avviata la conversione. La tensione risultante è disponibile all'uscita del DAC dopo il tempo di conversione. È inoltre possibile avviare una conversione a seguito di eventi provenienti dal sistema di eventi. In **Fig. 10** è riportato lo schema a blocchi del DAC. L'intervallo di conversione è compreso tra il riferimento di tensione selezionato del DAC e GND. Il riferimento di tensione predefinito è il riferimento interno. Altre opzioni di riferimento di tensione sono la tensione di alimentazione analogica (VDDANA) e il riferimento di tensione esterno (VREF). Il riferimento di tensione viene selezionato scrivendo i bit di selezione del riferimento nel registro di controllo Control B (CTRLB.REFSEL).

INTERFACCE SERIALI

A differenza della maggior parte delle schede Arduino che utilizzano la porta USB per la programmazione, nella uChip (come anche in Arduino Zero e relative varianti) la porta USB, oltre che da interfaccia di programmazione, funge da connessione seriale USB nativa del microcontrollore. Se la porta USB viene utilizzata per programmare non è possibile impegnarla in una connessione seriale, e viceversa. Ad esempio il software (lo sketch) che utilizza l'oggetto "Serial" per comunicare con il monitor seriale deve essere sostituito dall'oggetto SerialUSB. In alternativa, è possibile utilizzare un dispositivo convertitore seriale USB-UART a 3,3 V, come quelli basati sul chip FTDI, da connettere a una delle altre porte seriali della scheda. Quindi, l'oggetto "SerialUSB" viene utilizzato per inviare e ricevere dati seriali tramite la porta USB, ad es. al monitor seriale dell'IDE di Arduino. L'oggetto Serial viene utilizzato per il collegamento della porta seriale TX0-RX0 rispettivamente ai pin digitali D2 e D4 della scheda. È possibile utilizzare un convertitore seriale USB-UART TTL a 3,3 V per

il collegamento a questa porta. L'oggetto *Serial1* viene utilizzato per il collegamento della porta seriale TX1 e RX1 rispettivamente ai pin digitali D3 e D5 della scheda. Anche in questo caso può essere utilizzato un convertitore USB-UART collegato a questi pin.

REALIZZAZIONE PRATICA

Bene, snocciate le caratteristiche della scheda uChip andiamo alla parte pratica del nostro sistema, che vi offre due opzioni: la prima è il cablaggio tramite breadboard e jumper di tutti i componenti e le schede secondo lo schema di cablaggio che vedete in queste pagine.

Chi lo desidera, può invece assemblare il sistema sul circuito stampato da noi preparato allo scopo, realizzabile con le tracce rame scaricabili dal nostro sito www.elettronica.in.it; il PCB realizza le poche interconnessioni richieste e supporta fisicamente uChip e i moduli RFID, termometro e radar a ultrasuoni. Rispecchia in tutto lo schema elettrico che trovate in queste pagine.

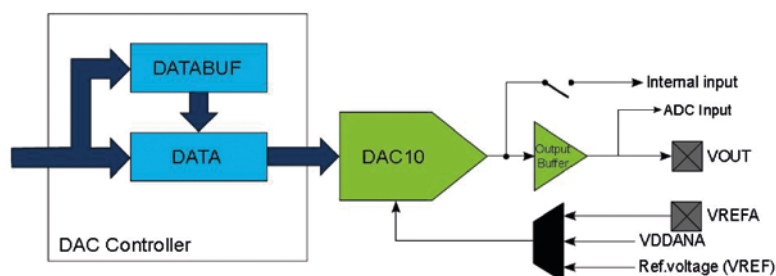
Comunque abbiate fatto, una volta completate le interconnessioni, per rendere operativo il sistema innanzitutto installate nel vostro computer l'IDE di Arduino dal sito ufficiale www.arduino.cc (in questo articolo si fa riferimento alla versione software Arduino 1.8.9). All'avvio dell'IDE si deve installare la scheda di sviluppo uChip, impartendo il comando *File -> Impostazioni* e, avuto accesso alla relativa finestra di dialogo, alla voce "URL aggiuntive per il Gestore schede" inserendo il seguente link: http://github.com/ITACAInnovation/uChip/raw/master/package_uChip_index.json.

Poi, con il comando *Strumenti > Scheda > Gestore Schede*, digitate "uChip" nella barra di ricerca, quindi selezionate e installate il firmware di gestione "uChip SAMD". Ora siete pronti!

INSTALLAZIONE DEL DRIVER

Eseguita l'installazione del package di uChip, tranne Windows 10, Linux, Mac e OS X, per i sistemi operativi Windows 8, 8.1, 7, Vista e XP bisogna installare il driver per la comunicazione seriale di uChip con il computer. I driver si trovano nella cartella drivers raggiungibile dal percorso `%USERPROFILE%\AppData\Local\Arduino15\packages\uChip\hardware\samd\1.0.1\drivers\` oppure, se avete una versione portatile di Arduino, troverete la cartella "drivers" seguendo il percorso `D:\...\\arduino-1.8.9\portable\packages\uChip\hardware\samd\1.0.1\drivers`. Aprite la cartella "drivers" e fate doppio clic sul file

Fig. 10
Schema
a blocchi
funzionale
del DAC.



eseguibile "uchip_driver_installer.exe". Windows chiederà l'autorizzazione per installare i driver, quindi cliccate su continua. La procedura di installazione dovrebbe richiedere alcuni secondi. Se non utilizzate l'IDE di Arduino, in alternativa potete scaricare i driver dal sito: <https://github.com/ITACInnovation/uChip/raw/master/uChip1.0.1.zip>. Una volta decompresso questo archivio, troverete i driver "uchip_driver_installer.exe" nella directory: uChip1.0.1\drivers\. Poi, come sopra, con doppio clic sul file "uchip_driver_installer.exe" installate i driver seguendo la procedura guidata.

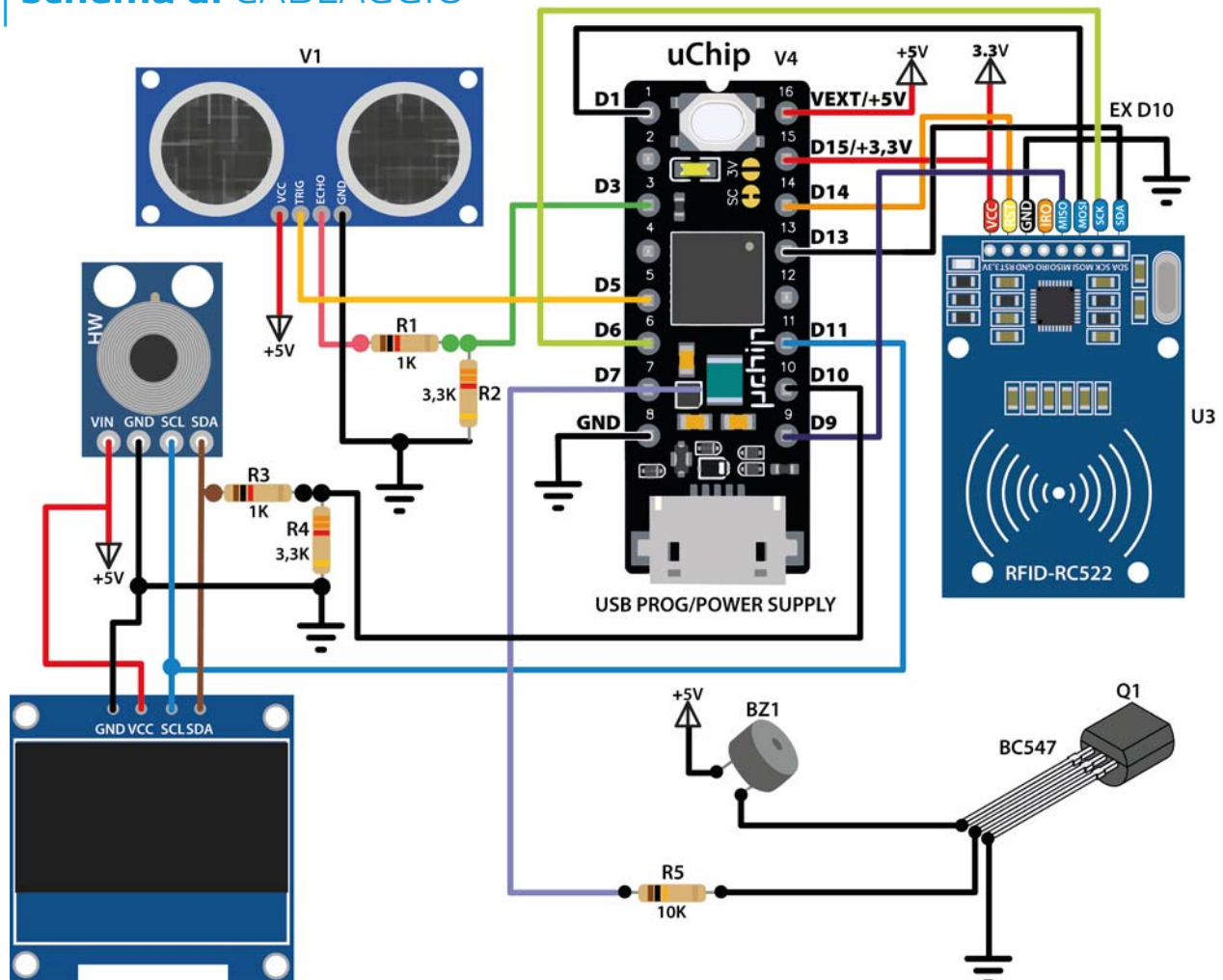
SKETCH E LIBRERIE

Procediamo ora con la creazione dello sketch e la descrizione del codice dell'applicazione: aprite l'IDE di Arduino e in un nuovo sketch copiate il codice

riportato nel nostro sito tra i file del progetto. Salvate il codice con il nome RFID_Temperatura, o come volete. Prima di caricare il codice nella scheda di sviluppo uChip, occorre scaricare i file .zip delle librerie Wire e MLX90614 rispettivamente da <https://github.com/PaulStoffregen/Wire> e <https://github.com/adafruit/Adafruit-MLX90614-Library>. Copiate i due file .zip scaricati in una cartella, quindi dall'IDE Arduino impartite il comando *Sketch -> #include libreria -> Aggiungi libreria da file .ZIP*, selezionate la cartella Wire-master.zip della libreria Wire che avete scaricato e copiato in una cartella. Fate lo stesso procedimento per aggiungere anche il file Adafruit-MLX90614-Library-master.zip della libreria MLX90614.

Restando su Gestore librerie, passiamo a installare le librerie per l'OLED SSD1306.

schema di CABLAGGIO

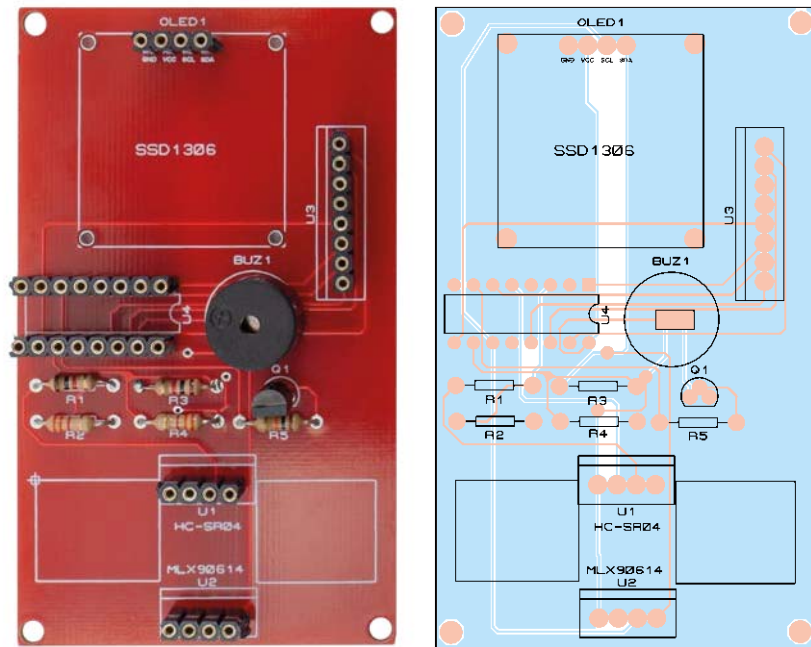


Elenco Componenti:

R1-R3: 1Kohm
 R2-R4: 3K3 ohm
 R5: 10 Kohm
 Q1: BC547
 BZ1: Buzzer senza elettronica
 U1: Misuratore ultrasuoni (MISDIST04)
 U2: Sensore temperatura (MLX90614)
 U3: Modulo RFID (VMA405N)
 U4: Uchip (UCHIPWSH)
 OLED1: Display oled (OLEDGVSCSD)

Varie:

- strip femmina 8 poli
- strip femmina 4 poli
- stampato s1566 (85x50)



Digitate SSD1306 nella barra di ricerca e installate la libreria Adafruit SSD1306 che appare come quella mostrata nella **Fig. 11**. Poi digitate "GFX" nella barra di ricerca e installate la libreria Adafruit GFX che appare come quella mostrata nella **Fig. 12**. A questo punto aprite l'IDE di Arduino e lo **sketch RFID_Temperatura**, cui andrete ad aggiungere dei file della libreria di gestione delle alimentazioni di uChip. A riguardo precisiamo che il pacchetto di supporto della scheda di sviluppo uChip include la libreria uChipPowerManagement, che consente un'agevole configurazione via software del sistema di alimentazione di uChip. Per utilizzare questa libreria si devono aggiungere allo sketch i file **uChipPowerManagement.h** e **uChipPowerManagement.cpp** tramite il comando *Sketch -> Aggiungi file -> uChipPowerManagement.h* e poi *Sketch -> Aggiungi file -> uChipPowerManagement.cpp*. Questi file si trovano nella cartella *C:\Arduino...\packages\uChip\hardware\samd\1.0.11\libraries\uChipPowerManagement\src*. Dopo aver aggiunto i file della libreria "uChipPowerManagement" chiudete e riaprite lo sketch **RFID_Temperatura**. L'IDE avrà l'aspetto di **Fig. 13**. A destra si noti la presenza di due tab corrispondenti ai file **uChipPowerManagement.h** e **uChipPowerManagement.cpp** che sono stati aggiunti precedentemente allo sketch. Cliccando sulla tab **uChipPowerManagement.cpp** potrete vedere tutte le funzioni della libreria di seguito descritte.

void uChipEnableBoost(uint8_t enable)

Abilita il convertitore boost interno se è inserito un cavo OTG (pin ID basso). I relativi parametri d'ingresso sono:

- **enable = UCHIP_BOOST_DISABLED**: boost sempre disabilitato, indipendentemente dalla presenza del cavo OTG;
- **enable = UCHIP_BOOST_ENABLED**: il boost è abilitato in presenza del cavo OTG.

void uChipSetVextValue(uint8_t value)

Imposta la tensione di uscita VEXT solo se uChip è alimentato tramite USB. I parametri sono:

- **value = VEXT_USB**: il Pin 16-VEXT è impostato alla tensione USB (in genere 5 V +/- 10%).
 Notare che la tensione effettiva dipenderà dalla corrente di uscita;
- **value = VEXT_3V3**: il Pin 16-VEXT è impostato a 3,3V.

void uChipSleep(uint8_t sleepMode, uint8_t sleepOnExit)

Imposta uChip in una modalità a basso consumo. I relativi parametri d'ingresso sono:

- **sleepMode = SLEEP_IDLE0**: il dominio del clock della CPU è arrestato;
- **sleepMode = SLEEP_IDLE1**: i domini di clock della CPU e AHB vengono arrestati;
- **sleepMode = SLEEP_IDLE2**: i domini di clock della CPU, AHB e APB vengono arrestati;

- sleepMode = SLEEP_DEEP: tutti i clock vengono arrestati tranne quelli richiesti da un modulo in esecuzione o che hanno il bit ONDEMAND impostato a zero. Inoltre il regolatore e la RAM sono impostati in modalità a basso consumo;
- sleepOnExit = SLEEP_ON_EXIT: quando si verifica un interrupt, riattiva uChip e la routine di interrupt (ISR) viene servita; dopo che l'ISR è completo, uChip torna automaticamente nello stato di stop;
- sleepOnExit = NO_SLEEP_ON_EXIT: come SLEEP_ON_EXIT, ma quando l'ISR è completa, uChip non torna automaticamente in modalità sleep.

DESCRIZIONE DEL CODICE

Seguendo lo sketch analizzeremo il significato del codice. Il codice inizia includendo le librerie necessarie che si occupano:

- Wire.h della gestione del protocollo I²C,
- Adafruit_MLX90614.h del sensore di temperatura MLX90614,
- SPI.h del protocollo SPI,
- MFRC522.h del lettore RFID,
- uChipPowerManagement.h dell'impostazione delle alimentazioni di uChip.

```
#include <Wire.h>
```



Fig. 11
Libreria
Adafruit_
SSD1306.

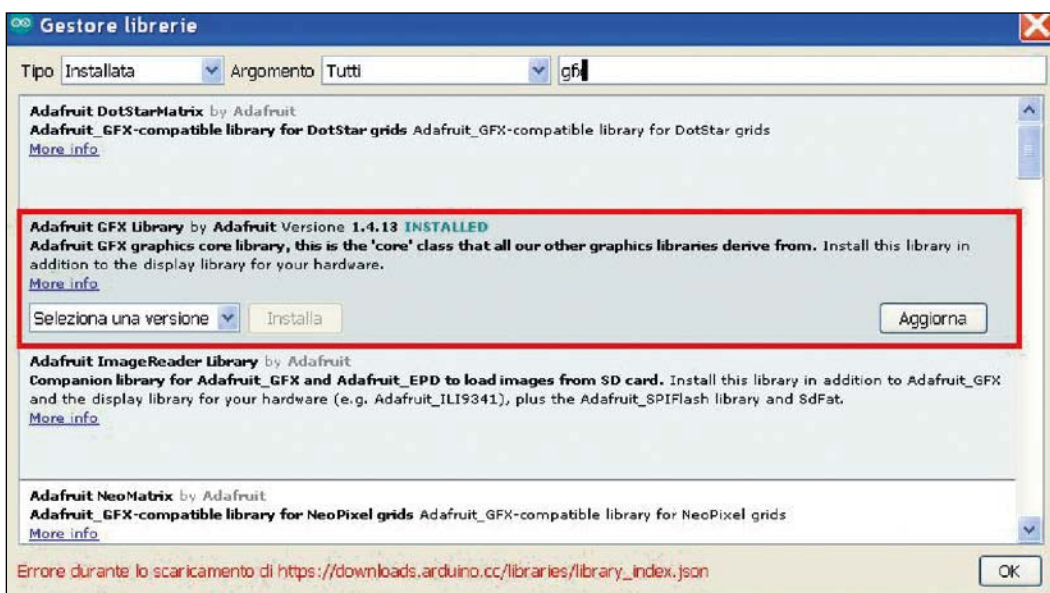
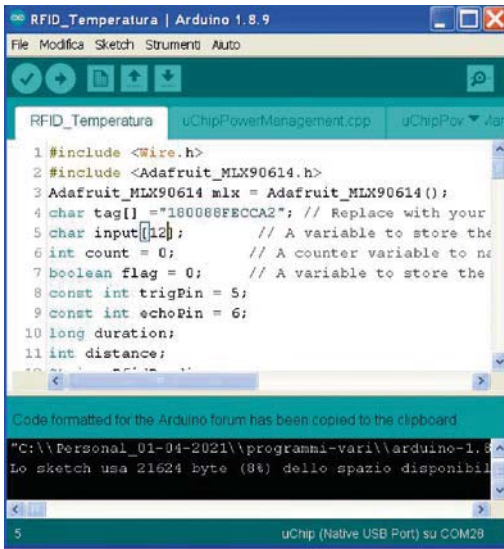


Fig. 12
Libreria
Adafruit_GFX.

→ **Fig. 13**



```
#include <Adafruit_MLX90614.h>
#include <SPI.h>
#include <MFRC522.h>
#include "uChipPowerManagement.h"
```

Vengono definiti i pin D5 e D3 di uChip collegati al sensore a ultrasuoni:

```
const int trigPin = 5;
const int echoPin = 3;
```

Viene creata la variabile stringa contenente l'UID della scheda tag RFID che sostituirete con quello della vostra scheda:

```
char tag[] = "E3 C5 21 15";
```

Sono definite le variabili per memorizzare i dati del sensore a ultrasuoni, del lettore RFID e del sensore di temperatura MLX90614:

```
long duration;  
int distance;  
String RfidReading;  
float TempReading;
```

La funzione void setup () imposta le alimentazioni di uChip e inizializza il display OLED, il monitor seriale per il debug, la comunicazione SPI, il sensore di temperatura MLX90614, il chip RFID MFRC522 e il plugin excel per l'invio dei dati.

Inoltre, imposta la modalità I/O dei pin Trig, Echo e BUZZ (**Listato 1**).

La funzione **void loop ()** calcola continuamente la distanza tra la persona e il sensore e se la distanza

è nel range stabilito, chiama la funzione reader ()
per la lettura della scheda tag:

```
void loop ()
{
    .....
    if (distance <= maxrange){
        if (distance >= minrange)
        {
            .....
        }
        reader();
    }
    delay(200);
} //end loop
```

La funzione **reader()** confronta i dati della scheda tag scansionata con l'UID del tag predefinito nel codice.

Se l'UID del tag corrisponde ai dati della scheda tag scansionata, registra il nome della persona, rileva la sua temperatura corporea, invia questi dati al foglio Excel e visualizza sul display OLED per 3 secondi la temperatura ambiente e del soggetto monitorato:

```

void reader()
{
.....

// Compare UID of the card/cards
if (content.substring(1) == "E3 C5 21 15")
{
..... .

    RfidReading = "Mario Rossi";
    threshold+=((distance-minrange)*deltagrade);
    temp_read();
    Write_streamer();
    Display_Temper();
    delay(3000);
    display.clearDisplay();
    ..... .

}
}

```

La funzione **temp_read ()** legge il valore di temperatura dal sensore MLX90614 in gradi Celsius aggiungendo un offset di calibrazione al valore misurato dal sensore e li memorizza nella variabile "TempReading":

```
void temp_read ()
{
    TempReading = threshold+mlx.readObjectTempC();
    .....
}
```

La funzione **Write_streamer()** si occupa dell'invio dei dati del tag dalla scheda uChip al foglio elettronico:

```
void Write_streamer()
{
    SerialUSB.print("DATA,DATE,TIME,");
    SerialUSB.print(RfidReading);
    //Store date on Excel
    SerialUSB.print(",");
    //Move to next column using a ","
    SerialUSB.print(TempReading); //Store date on Excel
    SerialUSB.print(",");
    //Move to next column using a ","
    SerialUSB.println(); //End of Row move to next row
}
```

La funzione **Display_Temper()** visualizza sul display OLED i valori della temperatura ambiente e del soggetto testato:

```
void Display_Temper() {
    .....
    display.println(mlx.readAmbientTempC(),1);
    .....
    display.println(TempReading,1);
    .....
}
```

Con l'hardware e il software completati, è ora possibile procedere al collaudo del sistema.

COLLAUDO

Per il test occorre prima di tutto rilevare il codice della vostra scheda.

Con il lettore RFID RC522 di Futura Elettronica, in dotazione avrete anche due dispositivi tag RFID, una scheda e un portachiavi che potrete utilizzare per il test previa rilevazione dei rispettivi codici con la procedura di seguito descritta: aprite l'IDE di Arduino e caricate lo sketch **RFID_Temperatura** nella scheda uChip cliccando su "Carica".

Aprire il monitor seriale e impostatelo a 9600 baud. Fate una scansione della scheda che avete scelto come utente autorizzato. Sul monitor seriale apparirà il codice UID con il messaggio: "Accesso autorizzato" e sotto il nome dell'utente e un valore casuale della sua temperatura, dati che non dovrete considerare in questa fase in quanto saranno riportati correttamente in un foglio Excel, come vedremo più avanti.

Ora passate sul lettore RFID il portachiavi che è stato scelto come tag non autorizzato.

Sul monitor seriale apparirà il codice UID del tag portachiavi con il messaggio: "Accesso non autorizzato". In **Fig. 14** viene riportata l'immagine del monitor seriale con i due UID dei tag.

Le informazioni che vedete sono archiviate nella memoria interna del tag in segmenti e blocchi, ovvero, 1.024 byte di memoria dati suddivisi in 16 settori e ogni settore è protetto da due chiavi diverse. Fra i dati visualizzati individuate e annotate l'UID della scheda autorizzata all'accesso.

Rilevati i due UID, aprite lo sketch **RFID_Temperatura** e inserite l'UID del tag che avete scelto come tag autorizzato (ad es. la scheda; il portachiavi verrà utilizzato nel test come tag di utente non autorizzato) sostituendolo all'UID di esempio nel codice (linea 111) dello sketch.

Riavviate l'IDE e caricate nuovamente lo sketch.

↓ Listato 1

```
void setup ()
{
    uChipSetVextValue(VEXT_USB);

    display.begin(SSD1306_SWITCHCAPVCC, 0x3C); //Start the OLED display
    .....
    SerialUSB.begin(9600); // Initialise Communication with the Serial Monitor
    SPI.begin(); // Initiate SPI bus
    mlx.begin(); // Initiate MLX90614
    mfrc522.PCD_Init(); // Initiate MFRC522
    Initialize_streamer();

    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(BUZZ_PIN, OUTPUT);
}
```

INSERIMENTO DATI UTENTE IN EXCEL

Per inviare i dati al foglio Excel utilizzeremo il plugin PLX-DAQ: si tratta di un software che trascrive i dati da uChip (o Arduino) direttamente in un foglio Excel. Il trasferimento dei dati avviene mediante il collegamento seriale fra uChip e il computer. Il file del plugin PLX-DAQ.zip lo scaricate dai file della rivista sul nostro sito. Decomprimete il file PLX-DAQ.zip e cliccate sul file plx_daq_install.exe per installare il plugin nel computer. Si creerà sul desktop una cartella denominata "PLS-DAQ". Aprite questa cartella e il file excel PLX-DAQ Spreadsheet al suo interno. Potrebbe apparire un messaggio di blocco di sicurezza indicante che le macro sono disabilitate, quindi per sbloccarle andate su *Strumenti -> Opzioni -> Protezione -> Protezione macro -> Livello protezione -> Medio -> OK* per abilitare le macro con consenso (questa procedura si riferisce a Windows XP). Abilitate le macro, si otterrà una schermata

excel con sovrapposto un pannello di controllo per la comunicazione seriale del foglio excel con il microcontrollore uChip, come illustrato in **Fig. 15**. Dal pannello di controllo impostate il baud-rate a 9600 e selezionate la porta di comunicazione a cui è connesso il microcontrollore uChip. Con Strumenti -> Porta: COM... verificate la presenza della porta COM di comunicazioni di uChip con il computer e che non sia aperto il monitor seriale che altrimenti andrebbe ad occupare la porta seriale. Quindi cliccate sul pulsante "Connetti" del pannello di controllo per stabilire la connessione di Excel con uChip in attesa di ricevere lo streaming dei dati del tag, ossia del nome del soggetto testato e della sua temperatura corporea. Per misurare la temperatura corporea ponete il dorso del polso parallelamente di fronte al sensore di distanza e quindi al sensore MLX90614. Così la distanza misurata sarà corrispondente a quella dell'MLX90614. Avvici-

Fig. 14
UID dei tag
scansionati.

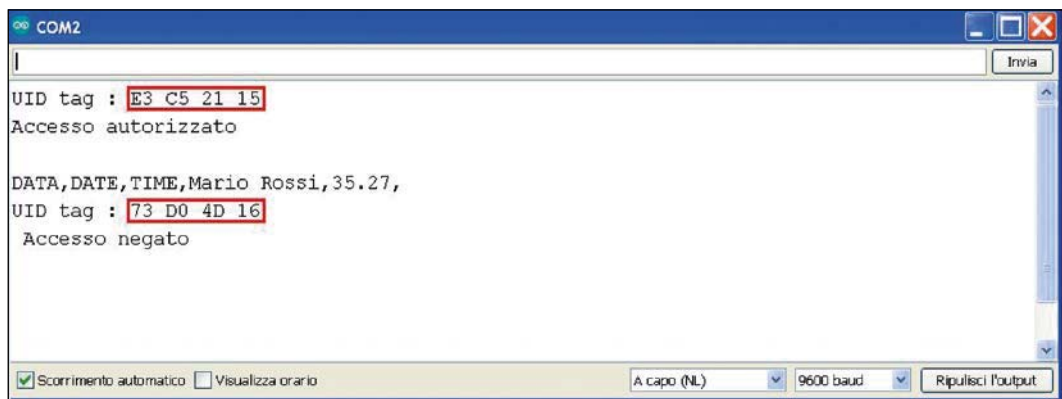
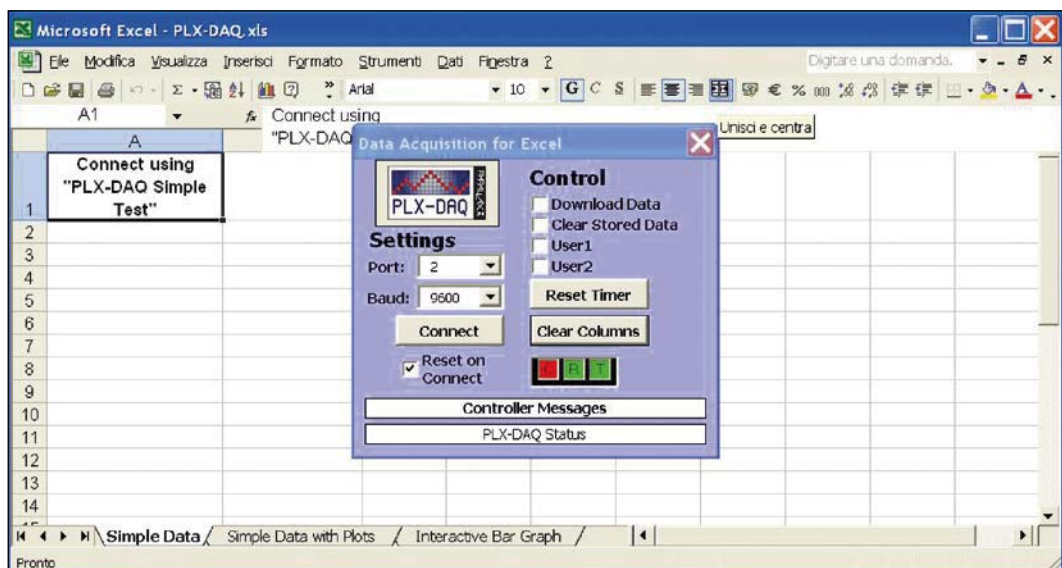


Fig. 15
Foglio Excel con
il pannello di
controllo
PLX-DAQ.



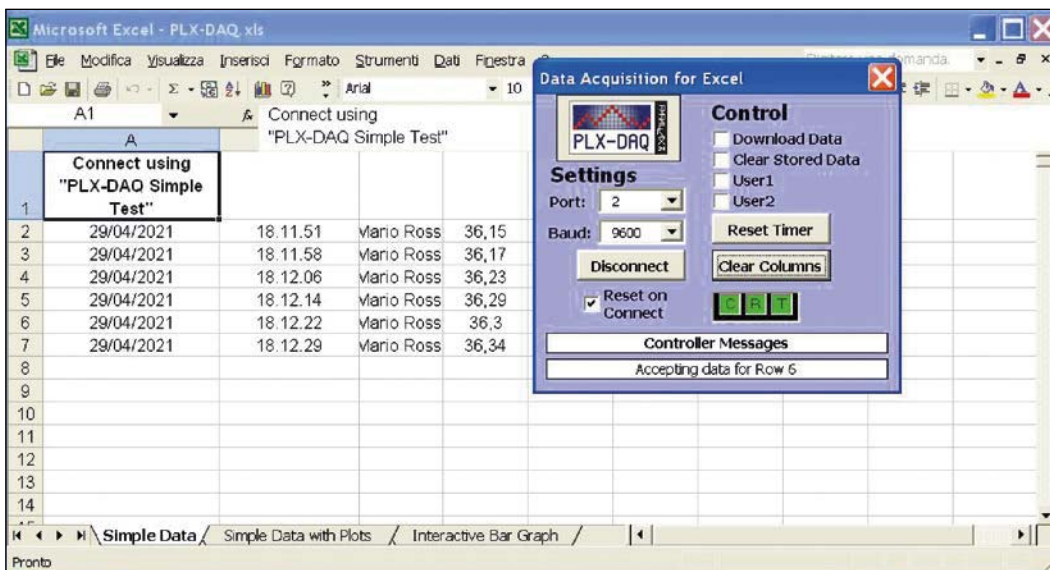


Fig. 16
Registrazione
dei dati di
utente sul
foglio Excel.

nate il polso ai sensori e quando udrete il doppio segnale acustico indicante la distanza dal sensore MLX90614 per la valida misura, mantenendo immobile il polso di fronte al sensore, effettuate la scansione avvicinando la scheda tag al lettore RFID RC522. Si interromperà il doppio beep e un singolo beep indicherà l'acquisizione del tag autorizzato. Contemporaneamente si avvierà il trasferimento dei dati verso il foglio excel su cui si registreranno la data e l'ora di acquisizione, il nome dell'utente titolare della scheda RFID e il suo valore di temperatura corporea misurata in gradi centigradi, come mostrato in **Fig. 16**. Come controprova, si potrà rilevare che, con la stessa procedura, passando il tag portachiavi sul lettore RFID (il cui UID non è inserito nel codice), non si udrà il beep di acquisizione, mentre continuerà il doppio bip attivato dalla vicinanza del polso al sensore della temperatura.

CONCLUSIONI

L'identificazione RFID delle persone può essere fatta singolarmente sostituendo ogni volta l'UID di ogni persona nel codice, quindi, per un'applicazione estesa per l'identificazione di gruppi di persone occorrerebbe creare un apposito database su EPROM o su SD card inserendovi gli UID di ogni persona. Per quanto riguarda la misurazione della temperatura corporea, considerate che il sensore MLX90614 qui utilizzato è una versione a basso costo; per maggior precisione si può utilizzare la versione MLX90614ESF-DCI per uso medico, che però costa circa \$60 (solo il chip). Inoltre, la precisione di misurazione dell'MLX90614 è garantita solo quando il sensore

è in equilibrio termico e in condizioni isotermitiche (non devono esserci differenze di temperatura nel package del sensore). Nella versione ad alta precisione MLX90614xCx i gradienti termici vengono misurati internamente e la temperatura misurata viene automaticamente compensata. In questo modo, MLX90614xCx è molto meno sensibile ai gradienti termici indotti dall'esterno, ma l'effetto non è totalmente eliminato. È quindi importante evitare di introdurre forti fonti di calore in prossimità del sensore o eventualmente schermare il sensore da queste fonti di calore. □



Cosa occorre?

I componenti utilizzati in questo progetto sono disponibili presso Futura Elettronica. La main board uChip (cod. UCHIP) è in vendita a Euro 19,00, il sensore di temperatura senza contatto (cod. MLX90614) è disponibile a Euro 19,00, il lettore RFID (cod. VMA405) costa Euro 12,90, il sensore ad ultrasuoni (cod. MISDIST04) è disponibile a Euro 5,00, il display OLED (cod. OLEDTVSCSD) è in vendita a Euro 11,00.

Il PCB può essere realizzato utilizzando il servizio di realizzazione PCB (cod. PCBPRODUCTION) disponibile sul sito [futurashop.it](http://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>

Temperatura sempre sotto controllo!

Termometri a infrarossi senza contatto



€ 38,00
Cod. TP213N

Termometro IR con puntatore LASER (da -20°C a +400°C)

Termometro a infrarossi UNI-T con puntatore LASER per la misura a distanza della temperatura (da -20°C a +400°C). Dispone di display LCD con retroilluminazione, possibilità di visualizzazione in gradi centigradi o in gradi Fahrenheit, spegnimento automatico. La lettura della temperatura è istantanea. La confezione comprende: il termometro e due batterie da 1,5 V tipo AAA.

Termometro IR con puntatore LASER (da -32°C a +420°C)

Termometro a infrarossi UNI-T con puntatore LASER per la misura a distanza della temperatura (da -32°C a +420°C). Dispone di display LCD con retroilluminazione, possibilità di visualizzazione in gradi centigradi o in gradi Fahrenheit, spegnimento automatico. La lettura della temperatura è istantanea (250 ms) con la memorizzazione della soglia minima e massima in caso di scansione continua e prolungata. Memorizzazione fino a 5 set di valori. La confezione comprende: il termometro, la pila da 9 V e la borsa da cintura per il trasporto.

€ 78,00
Cod. TP224



Prezzi IVA inclusa.



€ 19,00
Cod. FR811

Termometro IR per la lettura della temperatura corporea o di superfici (da +34°C a +42,9°C)

Dispone di ampio display a LED retroilluminato che mostra la lettura in gradi centigradi o Fahrenheit (selezionabile). Gli infrarossi consentono al termometro di leggere la temperatura corporea a una distanza di 5-10 cm in meno di un secondo. L'apparecchio si spegne dopo 8 secondi di standby. Il termometro ha una precisione del display di 0,1°C con un errore massimo di $\pm 0,2^\circ\text{C}$ e un range $34,0^\circ\text{C} \sim 42,9^\circ\text{C}$. Dispone inoltre di segnalazione di allarme con soglia impostabile, cambio di colore del display per indicare le soglie di temperatura misurate. Offre la possibilità di memorizzare fino a 32 misurazioni.



ACCENDERE IL PC DA REMOTO



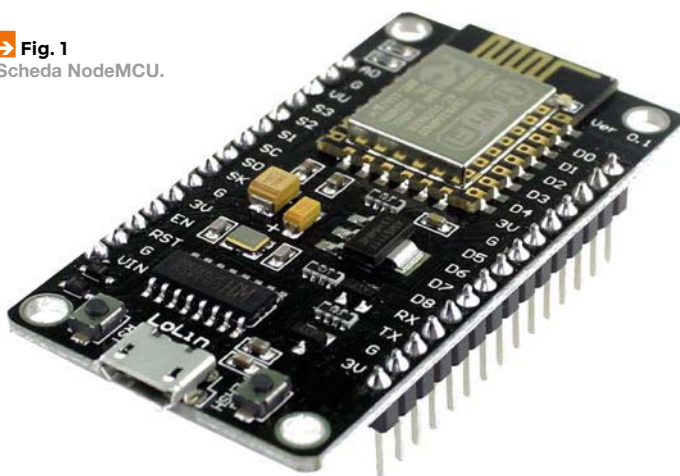
di MASSIMO DIVITO

Accendiamo
il nostro PC
fisso da remoto
o tramite
automazioni
sul nostro hub
domotico.

Di solito quando si parla di domotica il primo pensiero va al controllo di luci o di prese smart cui si collegano i più svariati elettrodomestici e utilizzatori elettrici di qualsiasi tipo. Diciamoci la verità: oggi ci siamo abituati a queste comodità; pensate ad esempio ad accendere da remoto il riscaldamento di casa prima di rientrare per tornare in un ambiente caldo e confortevole, oppure a spegnere le luci della cantina che distrattamente abbiamo lasciato accese ma facendolo dal cellulare, dal primo piano della nostra abitazione, senza dover necessariamente fare quattro rampe di scale! Con l'avvento, poi, degli assistenti vocali ci siamo abituati a comandare comodamente seduti sul divano del nostro salotto luci, musica, tende, condizionamento, riscaldamento e chi più ne ha più ne metta.

In questo articolo descriveremo un comando simile, ma

Fig. 1
Scheda NodeMCU.



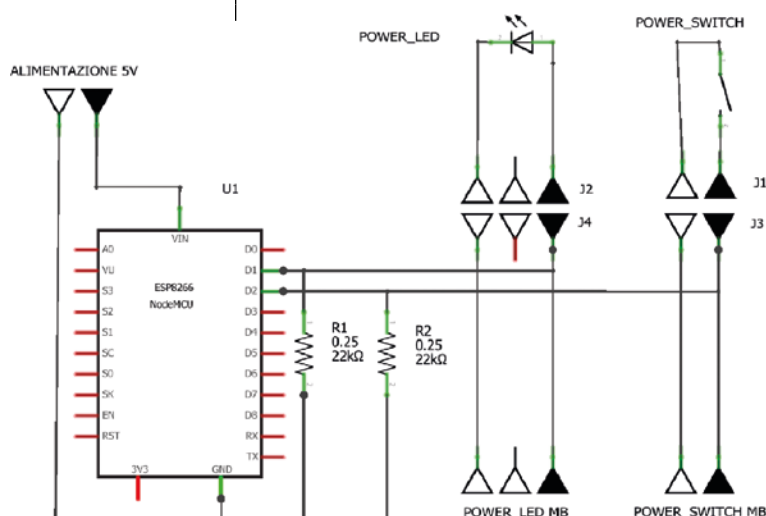
dedicato all'accensione del nostro PC, attraverso l'integrazione in un hub tramite il protocollo MQTT.

IL PROGETTO

Il nostro sistema è nato con lo scopo di riuscire ad integrare in un hub domotico il nostro Personal Computer, in modo da poterne controllare lo stato e di poterlo spegnere e accendere da remoto o a determinati giorni e orari, oppure creare scenari per esigenze specifiche: spengo il pc quando so che in casa non c'è più nessuno oppure lo accendo alle 8:00 di mattina del martedì e del venerdì perché in quei giorni lavoro in smart-working!

Una soluzione potrebbe essere quella di impostare nel BIOS del PC la funzione di accensione automatica al ripristino della corrente e di interporre tra la presa di corrente e l'alimentazione del PC una presa smart. In questo modo, quando vogliamo accendere il computer, comandiamo la presa smart

Fig. 2
Schema elettrico.



che lo alimenta ed esso si avvia in modo automatico. Quando invece vogliamo spegnerlo dobbiamo farlo prima da software e dopo ricordarci di spegnere la presa, facendo attenzione a non eseguire quest'ultimo passaggio troppo presto altrimenti rischieremmo di togliere l'alimentazione quando il processo di spegnimento non è ancora terminato. Chiaramente questa soluzione comporta dei rischi in quanto uno spegnimento non corretto del PC può portare alla corruzione del file system o, ancora peggio, alla rottura del disco rigido. Inoltre con questa soluzione avremmo lo svantaggio di poter conoscere solo lo stato della presa smart, ma non di sapere se il computer è realmente acceso o spento.

Il modo più sicuro è quello di interfacciarsi al pulsante di accensione del PC, che ne permette sia l'accensione che lo spegnimento, nonché al Power LED del computer per conoscere l'effettivo stato, vale a dire per capire se il computer si è effettivamente acceso.

Abbiamo quindi deciso di affidarci a un modulo con a bordo un ESP8266 e connettere due GPIO alla motherboard utilizzandone uno come uscita per comandare l'ingresso POWER_SWITCH ed uno come ingresso per leggere lo stato dell'uscita POWER_LED. Abbiamo previsto l'utilizzo di una scheda di sviluppo NodeMCU (**Fig.1**) perché ha a bordo già un regolatore a 3,3 volt che ci permette di alimentarla direttamente dalla tensione presente sulla motherboard oppure tramite un piccolo alimentatore esterno utilizzando la presa USB che ha a bordo.

LA SCHEDA NODEMCU

NodeMCU è una scheda che si basa sul noto transceiver WiFi ESP8266 della Espressif Systems, una rivoluzione nel settore dell'IoT (Internet of Things) poiché con una spesa di pochi euro si possono connettere velocemente una miriade di oggetti utilizzando la connessione WiFi di casa. La NodeMCU risolve il problema principale della scheda ESP8266, che è la mancanza di un sistema di interfacciamento diretto al PC oltre a velocizzare il flash ed il reset della scheda grazie alla presenza di due pulsanti che assolvono a tale funzione. Inoltre, a differenza dei moduli più compatti come l'ESP-01, dispone di tutti i GPIO accessibili tramite i pinheader della board. In pratica è una ESP8266 alla quale sono stati aggiunti tutta una serie di componenti elettronici che ne facilitano di molto l'utilizzo specie nella fase di prototipazione dell'oggetto.

I vantaggi rispetto all'utilizzo diretto della ESP riassumendo sono:

- facilità di connessione e di programmazione: si collega tramite USB;
- dimensioni e formfactor simili a quelle di un Arduino Nano con il vantaggio di avere una connessione WiFi;
- facilità nella fase di flash della scheda, si preme semplicemente un pulsante senza bisogno di ponticelli tra i pin;
- presenza di tutti i GPIO su pin-header.

IL CIRCUITO

Lo schema elettrico del circuito è molto semplice, come si vede dalla **Fig. 2**: oltre alla NodeMCU vi sono soltanto due resistori, R1 ed R2, del valore di 22K collegati uno tra il pin D1 ed il GND ed uno tra il pin D2 ed il GND. Questi resistori hanno la funzione di pull-down, servono cioè a tenere basso il livello dei pin quando al loro ingresso non è presente una tensione positiva.

Il pin D1 è poi collegato al positivo del connettore POWER_LED ed il D2 al positivo del connettore POWER_SWITCH, entrambi collocati sulla scheda madre del computer.

L'alimentazione della scheda la preleveremo direttamente dall'alimentatore del PC collegandoci direttamente al connettore ATX.

Il pin GND della NodeMCU va collegato ad un qualsiasi cavo negativo dell'alimentatore del PC,

identificato dal colore nero, mentre il pin VIN va invece collegato ad una sorgente di alimentazione fissa (che sia presente anche a PC spento) che nel caso del connettore ATX corrisponde al piedino 9 a cui fa capo filo di colore viola.

Nel caso decidiate di alimentare il circuito tramite un alimentatore esterno, utilizzando ad esempio la presa USB, ricordate che il cavo GND va ugualmente collegato a quello del PC.

Ricordiamo che lo standard ATX definisce le specifiche che riguardano case, scheda madre ed alimentatore di un Personal Computer. Nello specifico, detta le dimensioni della scheda madre, la posizione degli slot, dei fori per il fissaggio della scheda madre al case, dell'alimentatore e del contenitore, le posizioni e il colore dei connettori e il collegamento dell'alimentazione.

Grazie alla definizione di questo standard, elementi di marche diverse possono essere assemblati tra di loro senza problemi di compatibilità.

In **Fig. 3** possiamo vedere quelle che sono le tensioni presenti sul connettore ATX, notiamo subito che ad ogni tensione è associato un diverso colore anch'esso specificato dallo standard.

PROGRAMMAZIONE E FIRMWARE

L'ESP8266 ha un core compatibile con Arduino, la programmazione avviene quindi con lo stesso IDE. Prima però di potere iniziare dobbiamo compiere alcune operazioni preliminari.

Fig. 3
PinOut del
connettore con
standard ATX.

Fig. 4
Aggiunta repository
al gestore schede.

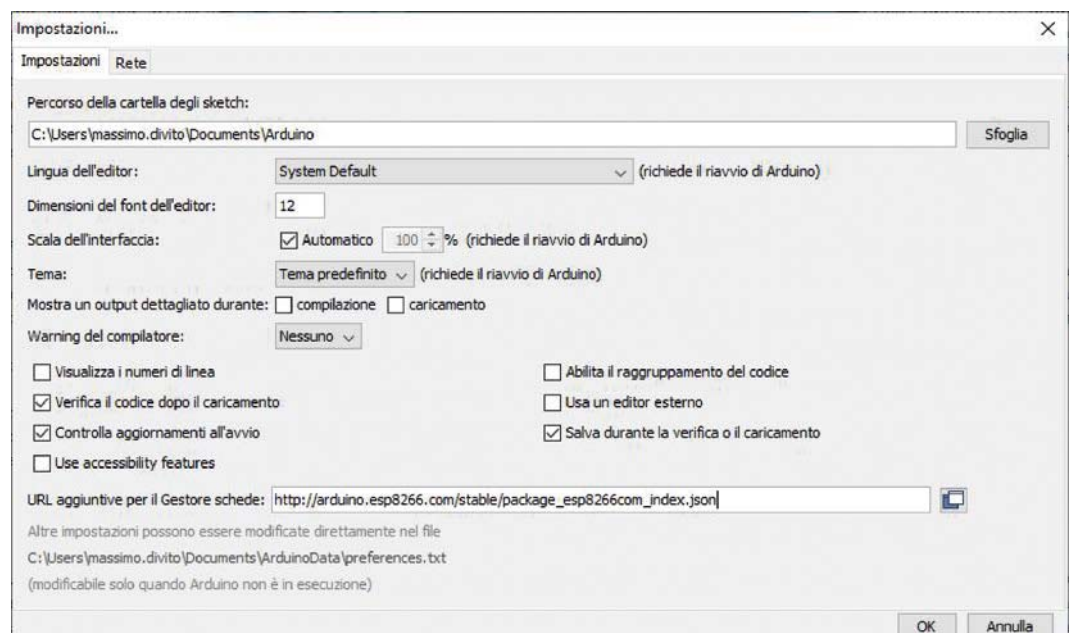
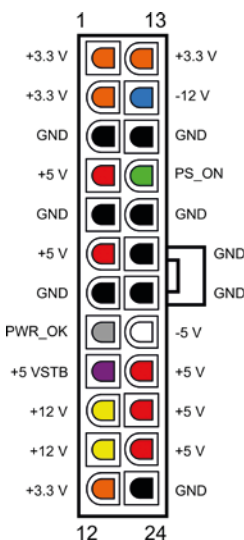


Fig. 5
Installazione
pacchetto ESP8266.



Come prima cosa dobbiamo installare il driver del chip CH340 (in alcuni modelli potrebbe essere presente un chip differente) che si occupa di convertire il segnale USB in seriale, lo possiamo scaricare dal sito del produttore al seguente link: http://www.wch-ic.com/downloads/CH341SER_EXE.html. Una volta installato il driver possiamo collegare la NodeMCU al computer, a questo punto apriamo l'IDE di Arduino e dal menu File andiamo in Impostazioni, procediamo quindi ad incollare il link http://arduino.esp8266.com/stable/package_esp8266com_index.json nel campo "URL aggiuntive per il Gestore di schede" e clicchiamo sul pulsante OK (**Fig. 4**).

Adesso dal menu Strumenti > Scheda > Gestore schede dobbiamo fare una ricerca con la parola chiave "ESP8266" trovando un pacchetto firmato ESP8266 Community, clicchiamo su Installa ed al termine su Chiudi (**Fig. 5**). Fatto ciò andiamo in Strumenti > Scheda > ESP8266 Boards e scegliamo la NodeMCU 1.0 (ESP12E Module) ed iniziamo con la scrittura del codice.

Di seguito vediamo solo alcuni punti più importanti dello sketch, la versione completa la potete scaricare dal nostro sito Internet. Come prima cosa importiamo le librerie:

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
```

La prima serve per la gestione della connettività WiFi, mentre la seconda serve per implementare la comunicazione tramite il protocollo MQTT nel nostro dispositivo.

Le definizioni che seguono indicano il valore di due timer, il primo definisce per quanto tempo mantenere attiva l'uscita collegata al pulsante di accensione mentre il secondo indica ogni quanto tempo pubblicare i dati verso il server MQTT:

```
#define TOGGLE_DURATION 300
// tempo attivazione pin pwr (PWR_SW)
#define PUB_PERIODIC_MS 1000 * 60 * 10
// pubblica i dati ogni 10 minuti
```

Le righe che seguono contengono i dati relativi ai parametri della rete WiFi, quindi SSID e password, ed indirizzo del server MQTT, la porta e la relativa user e password:

```
//Enter your wifi credentials
const char* ssid = "SSID";
const char* password = "password";

//Enter your mqtt server configurations
const char* mqttServer = "192.168.1.50";
//Enter Your mqttServer address
const int mqttPort = 1883; //Port number
const char* mqttUser = "mqtt_user"; //User
const char* mqttPassword = "mqtt_password";
//Password
```

L'ultima parte prima della routine di setup è relativa ai TOPIC ai quali il dispositivo deve sottoscrivere e quelli che invece deve pubblicare:

```
#define TOPIC_OUT_CONN "esp/pcswitch/conn"
#define TOPIC_OUT_PC_STATUS "esp/pcswitch/status"
```

```
#define TOPIC_IN_PC_POWER "esp/pcswitch/power"
#define TOPIC_IN_SYNC "esp/pcswitch/sync"
```

Nel caso si vogliano realizzare più dispositivi di questo genere sarà necessario cambiare il nome (pcswitch) contenuto in essi in modo che i TOPIC siano univoci per ogni scheda, ad esempio:

```
#define TOPIC_OUT_CONN "esp/pcswitch2/conn"
#define TOPIC_OUT_PC_STATUS "esp/pcswitch2/status"
#define TOPIC_IN_PC_POWER "esp/pcswitch2/power"
#define TOPIC_IN_SYNC "esp/pcswitch2/sync"
```

Nella routine di setup vengono impostati i piedini e viene poi inizializzata la connessione WiFi, successivamente viene inizializzata la connessione al server MQTT, in questa fase sulla console seriale sarà possibile verificare l'avvenuta connessione sia alla rete che al server.

Successivamente viene letto lo stato del PC per la prima volta e poi vengono pubblicati due TOPIC, il primo che indica che il dispositivo è online ed il secondo indicante lo stato del computer:

```
current_pc_status = digitalRead(PWR_LED);
client.publish(TOPIC_OUT_CONN, "1");
client.publish(TOPIC_OUT_PC_STATUS, String(current_
pc_status).c_str());
```

Si passa poi alla sottoscrizione dei TOPIC per

leggere il comando di accensione/spegnimento del computer e a quello di richiesta di sincronizzazione:

```
client.subscribe(TOPIC_IN_PC_POWER);
client.subscribe(TOPIC_IN_SYNC);
```

Nel loop dello sketch vi sono solo due controlli che si ripetono ricorsivamente il primo si occupa di leggere lo stato del PC e se differisce da quello impostato viene inviato un TOPIC per aggiornare il server. Subito dopo viene verificato che non sia passato il tempo impostato per la pubblicazione periodica dello stato, se è passato viene appunto pubblicato il TOPIC di stato online.

La funzione forse più importante è la MQTTcallback, la quale viene invocata ogni qualvolta arriva un TOPIC al quale ci siamo sottoscritti; è riportata nel **Listato 1**.

In questa funzione viene verificato che il TOPIC ricevuto sia TOPIC_IN_PC_POWER, in tal caso assegna alla variabile target_state il valore ricevuto, 1 per acceso oppure 0 per spento. Nel caso in cui il valore ricevuto differisca dal current_pc_status richiama la funzione TogglePcState. Nel caso in cui invece il TOPIC ricevuto sia TOPIC_IN_SYNC, con il payload uguale a 1, viene pubblicato il TOPIC_OUT_PC_STATUS con payload 1 che sta ad indicare che il dispositivo è online.

La funzione TogglePcState cambia lo stato del GPIO associato a PWR_SW portandolo a 0, poi

↓ Listato 1

```
void MQTTcallback(char* topic, byte* payload, unsigned int length) {
  if (strcmp (TOPIC_IN_PC_POWER, topic) == 0) {
    int target_state = current_pc_status;
    if ((char)payload[0] == '1')
    {
      target_state = 1;
    }
    else if ((char) payload[0] == '0')
    {
      target_state = 0;
    }
    if (target_state != current_pc_status)
    {
      TogglePcState ();
    }
  } else if (strcmp (TOPIC_IN_SYNC, topic) == 0)
  {
    if ((char) payload[0] == '1')
    {
      client.publish(TOPIC_OUT_CONN, "1");
      client.publish(TOPIC_OUT_PC_STATUS, String(current_pc_status).c_str());
      last_published_ms = millis();
    }
  }
}
```

COSA SONO GLI HUB DOMOTICI

Gli HUB domotici sono delle soluzioni hardware o software che servono per integrare in un unico ambiente e con un'unica interfaccia, dispositivi anche molto diversi tra di loro.

L'adozione di componenti domotiche diverse, di diversi produttori e con diverse tecnologie, richiede l'utilizzo di molteplici app, ognuna dedicata ad un numero limitato di dispositivi e quasi sempre appoggiata a soluzioni cloud proprietarie.

Gli HUB risolvono questo problema perché di solito dotati di

tecnologia modulare e di plugin che gli consentono di dialogare con accessori di diversa natura, indipendentemente dal produttore e dalla tecnologia utilizzata. Inoltre si possono installare su vecchi PC, Single Board Computer come il Raspberry Pi o macchine virtuali, sono quindi raggiungibili nella nostra rete senza ricorrere a nessun servizio di cloud.

Tra i vari HUB software disponibili gratuitamente i principali sono:

- Home Assistant
- OpenHAB
- Domoticz

attende il tempo impostato in TOGGLE_DURATION e lo riporta ad 1. In questo modo viene simulata la pressione del pulsante. Notate che ogni volta che viene pubblicato un TOPIC viene aggiornata la variabile last_published_ms:

```
last_published_ms = millis();
```

Questo serve a resettare il timer non bloccante che conteggia il tempo trascorso dall'ultima pubblicazione di un TOPIC.

REALIZZAZIONE PRATICA

Il montaggio è semplice, si tratta di saldare i due resistori di pull down tra D1 e GND e tra D2 e GND. Bisogna poi collegare a D1 e D2 due cavetti terminati con un connettore maschio ed uno femmina di tipo dupont in modo da collegarli in parallelo al pulsante di accensione ed al led di stato. Potete vedere un esempio di come assemblare il circuito in **Fig. 6**. Se non avete una pinza adatta a crimpare tali connettori potete utilizzare i cavetti (jumper) maschio-femmina che si utilizzano per cablare i circuiti sulle breadboard.

Il circuito va chiaramente protetto inserendolo in

↓ **Fig. 6**
Il nostro
circuito
assemblato.



↑ **Fig. 7**
Morsetto
"rubacorrente".

un contenitore plastico che potrà poi essere alloggiato all'interno del case del PC.

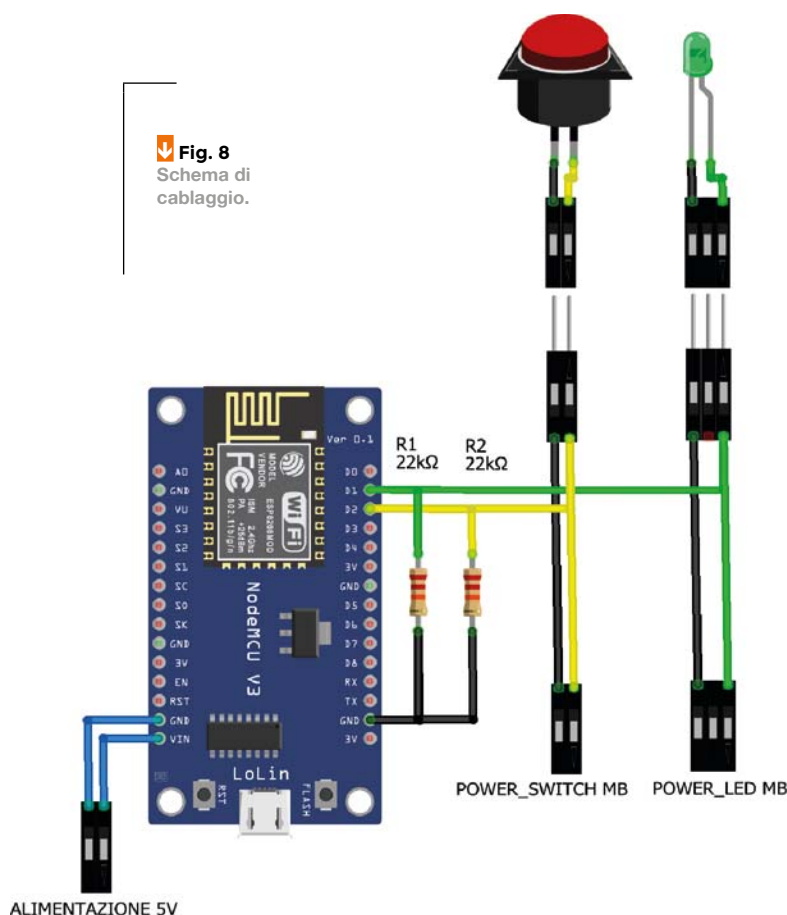
Terminato l'inscatolamento occorre passare al collegamento all'interno del computer, per prima cosa scollegiamo il cavo di alimentazione del PC in modo da lavorare in sicurezza senza il rischio di danneggiare niente. Fatto questo troviamo un posto adatto dove alloggiare il dispositivo e fissiamolo utilizzando delle fascette o del nastro biadesivo in modo da tenerlo ben stabile.

Per alimentarlo direttamente dall'alimentatore del PC dobbiamo collegare il positivo al piedino 9 del connettore ATX, quello con il cavo di colore viola.

Il negativo lo possiamo collegare ad un qualsiasi cavo di colore nero che fuoriesce dall'alimentatore, per comodità lo colleghiamo al piedino 7 del connettore ATX, in questo modo i cavi faranno lo stesso percorso ed avremo un cablaggio più ordinato. Per giuntare i cavi di alimentazione una soluzione veloce e pulita potrebbe essere quella di utilizzare dei morsetti "rubacorrente" (**Fig. 7**) che permettono di unire un cavo libero (quello dell'interfaccia) ad un cavo già intestato da entrambi i capi (quello dell'alimentatore) senza dover tranciare o sguainare quest'ultimo. Adesso dobbiamo procedere a collegare il dispositivo, seguendo lo schema di cablaggio proposto nella **Fig. 8**, al connettore FRONT_PANEL della scheda madre e più precisamente ai contatti del pulsante di accensione e del LED di stato. Sulla scheda madre sono presenti dei pinheader che di solito riportano la serigrafia con tutte le indicazioni, nel caso non sia chiara controllare sul manuale della scheda madre.

Il collegamento andrà fatto, in entrambi i casi, sul pin che riporta il simbolo +; in caso di inversione della polarità il dispositivo non funzionerà.

Fig. 8
Schema di cablaggio.



Per ciò che riguarda questi cavi, dato che dovremo collegarci in parallelo al led di stato ed al pulsante di accensione, intersteremo ognuno di essi con due connettori dupont, uno femmina che andrà collegato alla scheda madre ed uno maschio che andrà al cavo del LED o del pulsante (**Fig. 9**).

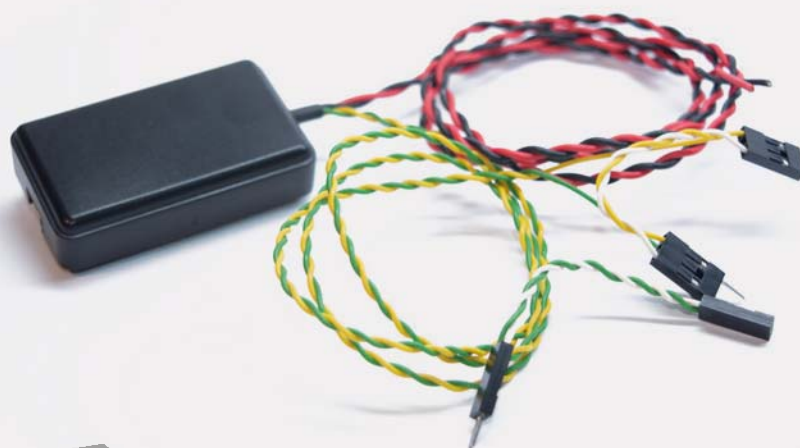
INTEGRAZIONE IN HOME ASSISTANT

Terminata l'installazione all'interno del PC è necessario integrare il dispositivo all'interno dell'HUB domotico, nel nostro caso prediamo come esempio Home Assistant che è probabilmente uno dei più diffusi. Per prima cosa andiamo ad editare il file switches.yaml oppure nel caso abbiate la configurazione in un file unico editiamo configuration.yaml:

```
- platform: mqtt
  name: "Computer"
  icon: mdi:desktop-classic
  state_topic: "esp/pcswitch/status"
```

```
command_topic: "esp/pcswitch/power"
availability_topic: "esp/pcswitch/conn"
payload_on: "1"
payload_off: "0"
payload_available: "1"
payload_not_available: "0"
qos: 1
retain: false
```

Fig. 9
Circuito incascolato e cavi terminati con connettori maschio-femmina dupont.



MQTT (Message Queue Telemetry Transport) è un protocollo di messaggistica nato nel 1999, con l'obiettivo di gestire le connessioni machine to machine.

Le caratteristiche principali di MQTT sono quelle di essere un protocollo semplice e leggero, nonché di minimizzare al massimo il traffico e richiedere risorse minime ai dispositivi che lo utilizzano.

Semplificando al massimo il pattern publish-subscribe su cui si basa: quando il nodo A vuole comunicare con il nodo B, non lo fa in modo sincrono o diretto. Al

contrario, nel protocollo MQTT il messaggio viene pubblicato dal nodo A (publish) e viene ricevuto dai nodi che sottoscrivono la ricezione del messaggio stesso (subscribe).

A voler essere più precisi, il protocollo MQTT prevede lo scambio di messaggi tramite un apposito broker: un broker MQTT non è altro che un software, che si occupa di ricevere i messaggi dai vari dispositivi e renderli disponibili a chi ne fa richiesta.

Per saperne di più potete leggere i nostri articoli presentati nei fascicoli n° 254, n° 255 e n° 256.

Salviamo il file e riavviamo il server.

In queste righe abbiamo definito quali sono i TOPIC di stato, di comando e di disponibilità del dispositivo. Riguardo a quest'ultimo ricordiamoci che il dispositivo trasmette lo stato ogni PUB_PERIODIC_MS millisecondi. Pertanto se avviamo il server dopo aver alimentato il dispositivo, per vederlo raggiungibile e poterlo comandare, dobbiamo aspettare tale tempo.

Per ovviare a questo problema possiamo creare un'automazione che parte all'avvio di Home Assistant in modo che ad ogni riavvio del server venga richiesto immediatamente lo stato dell'interfaccia. Editiamo quindi il file automations.yaml aggiungendo:

```
- alias: "Censimento stato PCswitch"
  trigger:
    - event: start
```

```
platform: homeassistant
condition: []
action:
- service: mqtt.publish
  data:
    topic: " esp/pcswitch/sync "
    payload: "1"
```

Terminata questa fase accediamo all'interfaccia web di Home Assistant, andiamo in modifica plan- cia ed aggiungiamo l'entità Computer. Se abbiamo fatto tutto correttamente dovremmo ritrovarci lo switch che permette di controllare l'accensione e lo spegnimento del nostro PC dal nostro HUB domotico (**Fig. 10**).

Questo però è solo l'inizio, perché da questo momento potremo impostare programmazioni per far sì che il PC si accenda o si spenga in determinati orari oppure impostare scenari che ci permettano ad esempio di spegnere il PC quando non siamo in casa o magari di accenderlo quando accendiamo le luci dello studio. □



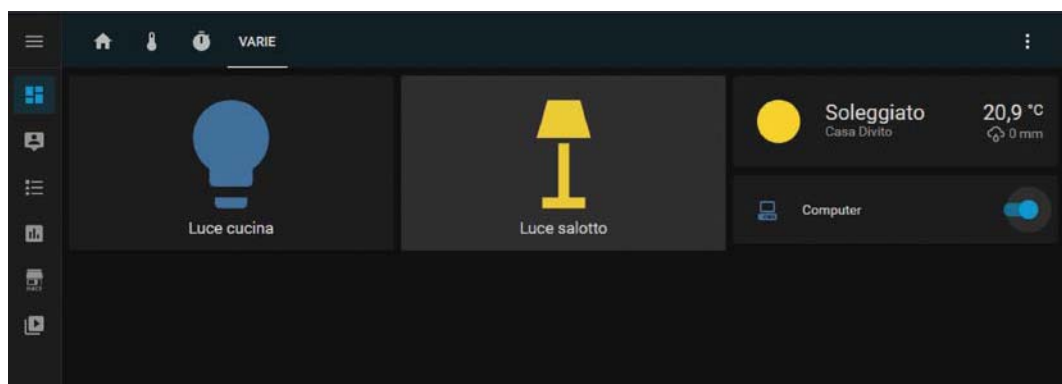
Cosa occorre?

I componenti utilizzati in questo progetto sono disponibili presso Futura Elettronica. La breakout board con ESP8266 (cod. NODEMCUESP) è in vendita a Euro 9,50, il contenitore utilizzato (cod. SC-703) è disponibile a Euro 1,50. 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>

 **Fig. 10**
Switch integrato in Home Assistant.



Nuovo allarme evoluto WiFi-GSM/GPRS



€119,00

Cod. FR783



Controllabile con:



Smart Life



Tuya Smart



Amazon Alexa



Google Home

Centralina antifurto wireless programmabile, controllabile da remoto tramite Smartphone sfruttando la rete GSM/GPRS o Wi-Fi e direttamente con Amazon Alexa o Google Home. Le App "Smart Life" o "Tuya Smart" permettono di configurare e gestire il funzionamento dell'allarme in ogni suo dettaglio. È possibile aggiungere sensori ed accessori opzionali per coprire fino a 99 zone. La confezione comprende: la centralina, due radiocomandi, un sensore wireless per porte/finestre, un sensore PIR wireless, due chiavi TAG RFID, la batteria di backup e l'alimentatore di rete. Manuale di utilizzo in italiano.

Prezzi IVA inclusa



Accessori
opzionali:

Sensore PIR
wireless



€14,00

Cod. FR783PIR

Sensore
porte/finestre
wireless



€11,00

Cod. FR783MAGN

Radiocomando
supplementare



€9,00

Cod. FR783RAD

Transponder
a portachiavi



€5,00

Cod. FR783TAG

Sirena wireless
da interno



€29,00

Cod. FR783SIR

Sirena wireless
da esterno con
pannello solare



€65,00

Cod. FR783SIREXT

PER I TUOI PROGETTI UTILIZZA I SET DI COMPONENTI

Vasta scelta e grande risparmio!

Euro 14,90
cod. YM397



**SET 1000 PEZZI DI LED
3 MM - VARI COLORI**

Euro 5,90
cod. SETCONDELE



**SET CONDENSATORI ELETTROLITICI
12 VALORI**

Euro 8,00
cod. YM403



**SET 100 CONDENSATORI
POLIESTERE - VARIE CAPACITÀ**

Euro 7,50
cod. KITFUS5X30



**KIT FUSIBILI 82 PEZZI 6X30
ASSORTITI**

Euro 7,50
cod. YM406



**SET 300 PEZZI LED 3-5 MM
VARI COLORI**

Euro 9,00
cod. YM374



**SET 50 TRIMMER DI PRECISIONE
MULTIGIRI - MONTAGGIO VERTICALE**

Euro 9,00
YM398



**SET 500 PEZZI LED 5 MM
VARI COLORI**

Euro 6,00
cod. YM379



**SET COMPONENTI ELETTRONICI
ASSORTITI**

Euro 12,00
cod. YM377



**SET 300 TRANSISTOR ASSORTITI
TO-92**

Euro 21,90
cod. SETELEC



**SET COMPONENTI ELETTRONICI
VARI CON PIASTRA E ALIMENTATORE**

Euro 8,00
cod. SET24INDU



**SET INDUTTANZE 24 VALORI
10 PEZZI PER VALORE**

Euro 14,00
cod. SMDRES3K610M



**SET RESISTENZE SMD 0805
DA 3,6 KOHM A 10 MOHM**

Euro 6,00
cod. 140ZENER



**SET 140 DIODI ZENER
10 PEZZI PER VALORE**

Euro 14,00
cod. SMDRES03K6



**SET RESISTENZE SMD 0805
DA 0 A 3,6 KOHM**

Euro 3,90
cod. SETCERAMICI



**SET CONDENSATORI CERAMICI
22 VALORI - 10 PEZZI PER VALORE**

Euro 6,50
cod. SETDIODI



**SET DIODI RADDRIZZATORI
100 PEZZI**

Euro 5,50
cod. SETTRIMMERMO



**SET TRIMMER MONTAGGIO
ORIZZONTALE - ASSORTITI**

Euro 7,00
cod. SETMULTIST



**SET CONDENSATORI MULTISTRATO
VARI VALORI - 10 PEZZI PER VALORE**

Euro 16,00
cod. KRES-0603



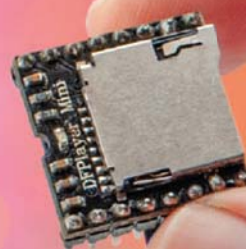
**SET 1700 RESISTENZE SMD
(SERIE 0603) - ASSORTITE**

Euro 6,50
cod. HS2133



**SET 66 PZ ZOCCOLI A LAMELLA
ASSORTITI**

MP3 CON IL MODULO DFR0299



del Prof. MICHELE MENNITI

Impariamo ad integrare un lettore MP3 nei nostri progetti, per riprodurre file audio vocali e musicali.

Il protocollo di compressione audio MP3 ha consentito di realizzare svariati dispositivi un tempo impensabili e di registrare e riprodurre, in formato numerico, parlato e brani musicali, garantendo un livello di fedeltà sonora quasi paragonabile a quello del Compact Disc. Nella sua forma tradizionale, con campionamento a 44,1 kHz, lo standard consente di comprimere 1 minuto di musica in circa 1 MB.

Proprio per la sua importanza, in diverse occasioni abbiamo presentato progetti basati sull'utilizzo di file MP3 e, negli ultimi anni, ci siamo dedicati a un dispositivo disponibile sotto forma di breakout board, che è il modulo MP3 DFPlayer Mini DFR0299 della DFRobot; con esso abbiamo realizzato progetti come l'Albero di Natale WiFi (n. 231) il Lettore MP3 per auto (n. 212) il Presepio (n. 201) e la Demoboard MP3 (presentata nel fascicolo 209) che con

CARATTERISTICHE DEL DFR0299

- Frequenze di campionamento supportate (kHz): 8 – 11,025 – 12 – 16 – 22,05 – 24 – 32 – 44,1 – 48.
- Uscita DAC a 24 bit, gamma dinamica 90 dB e rapporto segnale-rumore (SNR) 85 dB.
- Supporto completo file system FAT16 e FAT32, capacità massima SD Card o Pen drive USB 32 Gb.
- Flash memory: 64 Mb.
- Tipologie di controllo: I/O, seriale, pulsanti AD.
- Funzione di sospensione a scopo pubblicitario.
- File audio organizzabili in 100 cartelle di 255 brani l'una.
- Volume regolabile in 30 livelli, equalizzatore a 6 livelli.

20 pulsanti ci permette di gestire le principali funzioni del modulo, nonché di eseguire direttamente e singolarmente 14 file MP3 presenti nel supporto di memoria collegato, a scelta tra una Pen drive USB o una SD Card.

La diffusione e la versatilità di tale modulo ed anche la gestione ottimale tramite le schede Arduino, ci ha spinto a ad analizzare tutti gli aspetti rilevanti e a spiegare quel che ogni sperimentatore vorrebbe sapere per dotare il proprio progetto di un riproduttore MP3.

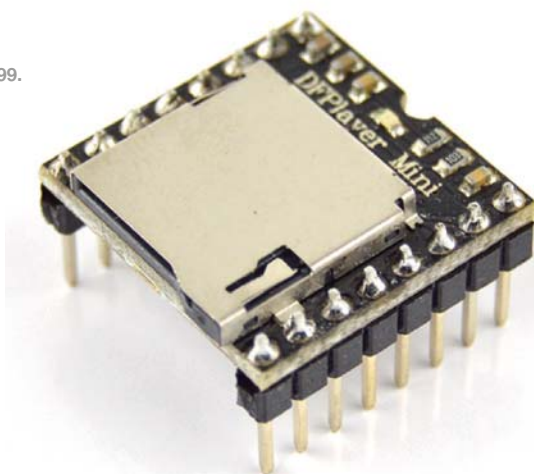
Nello studio del modulo MP3 ci siamo avvalsi di una piattaforma come la demoboard suaccennata, disponibile sul sito www.futurashop.it con il codice FT1280K.

IL MODULO DFR0299

Il nostro riproduttore è prodotto dalla DFRobot, la quale ci mette a disposizione una pagina wiki, al link https://wiki.dfrobot.com/DFPlayer_Mini_SKU_DFR0299#target_0, sulla quale si possono reperire tutte le necessarie informazioni, le più

Fig. 1

Il modulo DFR0299.



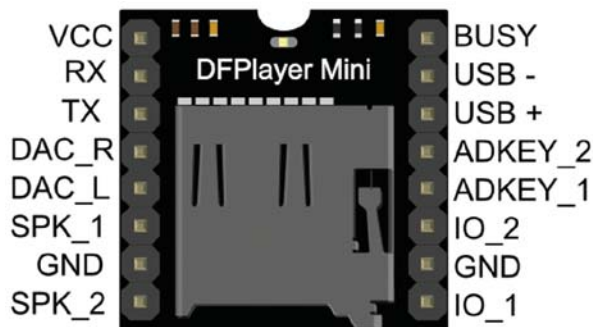
importanti le riporteremo in questo articolo; inoltre dall'indirizzo <https://github.com/DFRobot/DFRobotDFPlayerMini> è possibile scaricare l'indispensabile libreria per Arduino, DFRobotDFPlayerMini-1.0.3.zip, che comunque abbiamo caricato nella sezione download del sito www.elettronica.it.

Il modulo DFR0299 (Fig. 1) si presenta sotto forma di un piccolo PCB con uno slot per una micro SD card e due file di 8 pin strip, che lo rendono facilmente innestabile su uno zoccolo costituito da due header.

La piedinatura del modulo, con i riferimenti dei singoli pin, è dettagliata nella Fig. 2.

Si può notare come il modulo sia ricco di possibilità di interfacciamento, sia per la sua gestione (tramite porta USB o seriale UART o collegamento I/O o ancora mediante pulsanti esterni) che per l'audio (uscita stereo preamplificata per auricolari o amplificatore esterno, oppure mono per altoparlante fino a 3W, sfruttando l'amplificatore interno).

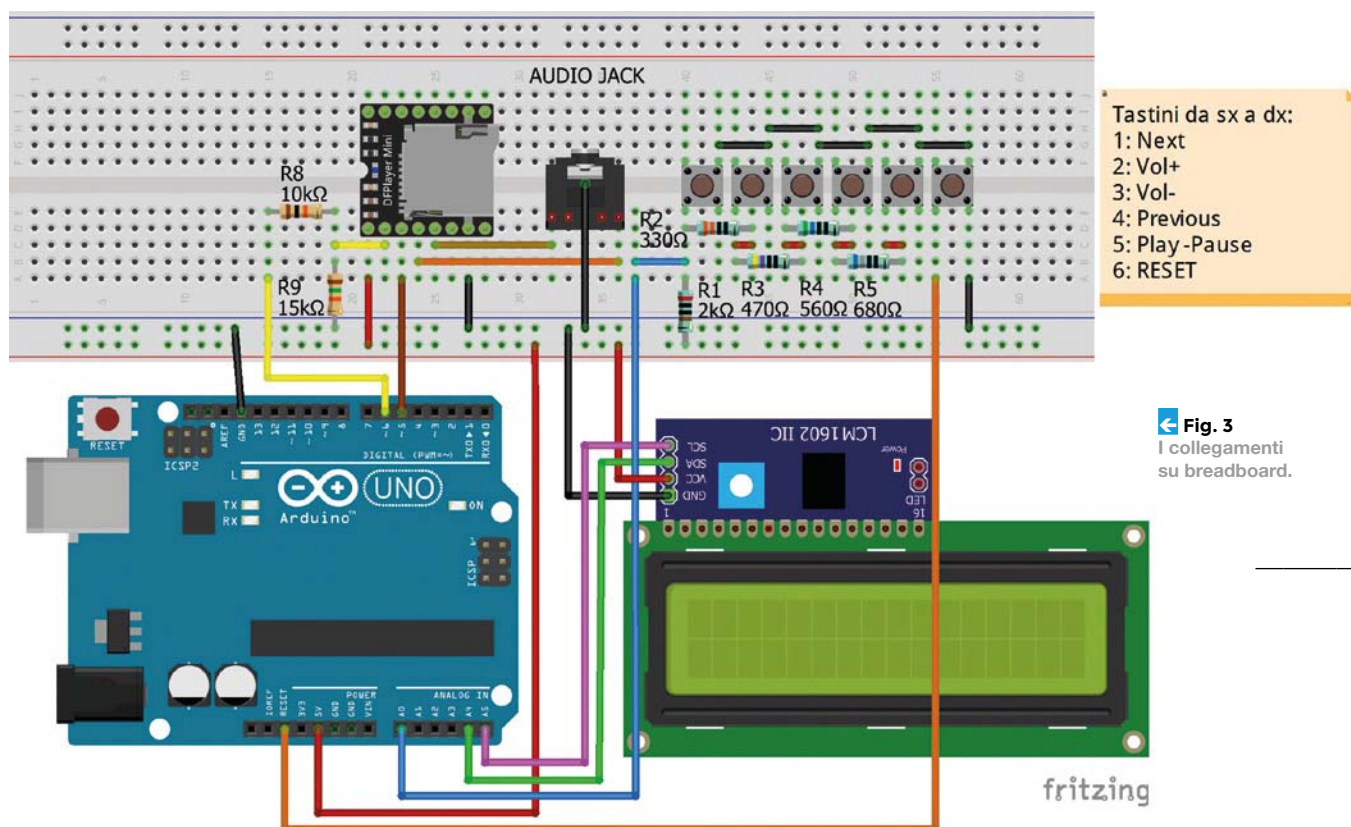
Alle due porte ADKEY1 e ADKEY2 è possibile collegare una serie di resistori che, assieme ai due



PIN	Descrizione	Note
VCC	Positivo alimentazione	3,2 ÷ 5 Vdc
RX	Ingresso seriale UART	
TX	Uscita seriale UART	Livello TTL 3,3 V
DAC_R	Uscita audio canale destro	
DAC_L	Uscita audio canale sinistro	Per auricolare o amplificatore
SPK2	Altoparlante -	Max 3 W
GND	Massa	Negativo alimentazione e DAC
SPK1	Altoparlante +	Max 3 W
IO1	Porta 1 Trigger	Pressione breve: brano precedente Pressione lunga: decremento volume
GND	Massa	Negativo alimentazione e DAC
IO2	Porta 2 Trigger	Pressione breve: brano successivo Pressione lunga: aumento volume
ADKEY1	Porta AD 1	Attiva l'esecuzione del primo segmento
ADKEY2	Porta AD 2	Attiva l'esecuzione del quinto segmento
USB+	USB Dati +	Porta USB per il collegamento ad un PC o ad una flash memory USB
USB-	USB Dati -	
BUSY	Stato esecuzione	Low=Play; High=stop/pausa

Fig. 2

Piedinatura del modulo DFR0299.



localizzati all'interno del modulo, costituiscono un partitore resistivo in grado di gestire 10 pulsanti per porta (il principio utilizzato nella nostra Demo-board FT1280K).

In alternativa è possibile realizzare una configurazione minimal collegando 4 pulsanti alle porte ADKEY1, ADKEY2, IO1 e IO2 per la gestione del volume e per l'esecuzione di due brani demo. La modalità che ci interessa è quella del collegamento seriale UART ad una board Arduino, in tal modo potremo gestire il modulo via software, con l'ausilio di un keypad analogico per l'invio dei comandi e di un display LCD per visualizzarne lo stato.

REALIZZAZIONE DEL CIRCUITO

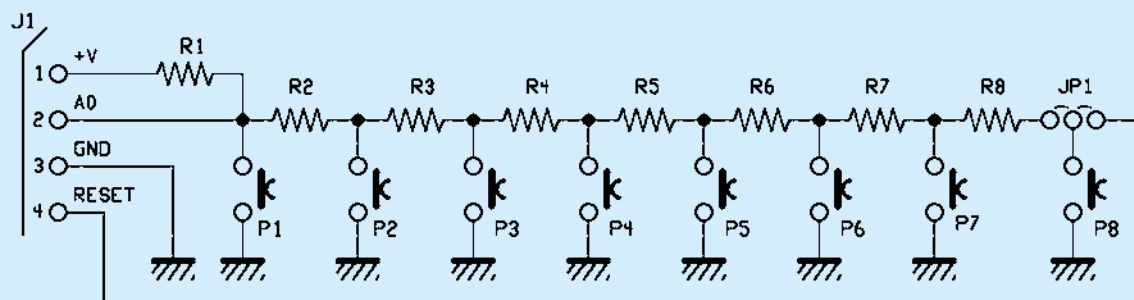
Per realizzare il nostro circuito test faremo ricorso ad una breadboard, che useremo per interfacciare una board Arduino UNO, il modulo DFR0299 originale, un display LCD 16x2 con interfaccia I²C ed un keypad analogico; gli unici componenti aggiuntivi sono due resistori, necessari per realizzare un adattatore di livello tra Arduino e modulo. In **Fig. 3** la rappresentazione grafica dei collegamenti.

Confrontando il circuito con la **Fig. 2** possia-

mo notare che abbiamo usato questi pin del modulo: 1-VCC collegato ai 5 V, 2-RX e 3-TX per la comunicazione seriale, 4-DAC_R e 5-DAC_L per l'uscita audio, 7-GND per la massa. Riguardo il collegamento seriale si vede come il pin TX (uscita seriale del modulo) sia collegato direttamente al pin 5 di Arduino UNO, mentre l'ingresso seriale RX necessita di un partitore. Il motivo è legato al fatto che, nonostante l'alimentazione a 5V, il modulo opera con segnali di livello TTL a 3,3 V; poiché Arduino UNO genera segnali a 5 V il modulo si potrebbe danneggiare. Sulla wiki consigliano di mettere un resistore da 1 kOhm in serie, ma tecnicamente non è corretto in quanto così si ottiene una limitazione di corrente e non un abbassamento del livello di tensione. Il sistema adottato da noi invece riduce il segnale in arrivo al modulo secondo la formula

$$V_{rx} = V * \frac{R_9}{R_8 + R_9} = 5 * \frac{15000}{10000 + 25000} = 3V$$

in cui V_{rx} è il livello del segnale in uscita da Arduino UNO (5 V) mentre V è il livello in uscita dal partitore. È importante aver chiaro che lo sketch



che abbiamo preparato per il test del modulo DFR0299 richiede espressamente che quest'ultimo sia il modello originale della DFRobot, perché in caso contrario tutta la gestione dello stato (Play/Pause, Stop, cambio brano) non potrà funzionare, in quanto i codici generati dai moduli compatibili (almeno quelli testati da noi) sono diversi. Approfondiremo la questione più avanti. Altra cosa da specificare riguarda il keypad, ora illustreremo la versione completa a 8 pulsanti,

in realtà ne abbiamo usati solo sei, quindi nella **Fig. 3** abbiamo ommesso due pulsanti e due resistori; per non creare confusione con le sigle dei resistori abbiamo evitato di usare "R6" e "R7". Nello schema di cablaggio, abbiamo rappresentato il keypad sotto forma di 6 pulsanti e 5 resistori; potete replicare esattamente questo schema sulla breadboard, oppure realizzare un piccolo modulo su millefori o PCB dotato di 8 pulsanti, 7 resistori e due connettori (**Fig. 4**). In quest'ultimo caso potete far riferimento al piano di montaggio ed all'elenco dei componenti. Inoltre potete scaricare i file gerber dalla nostra sezione download. Come si vede chiaramente dallo schema elettrico, si tratta di un sistema a partitore multiplo in cui il riferimento è il resistore R1, che ha la funzione di pull-up (collegata al positivo di alimentazione), quindi in condizioni normali l'uscita del circuito (A0) è allo stesso livello della tensione di alimentazione. Premendo il pulsante P1 l'uscita A0 viene collegata direttamente a massa, ossia al negativo di alimentazione, quindi rappresenta la condizione completamente opposta a quella precedente. Premendo P2 invece si viene a formare il partitore resistivo con R1-R2, con quest'ultima in funzione di pull-down (collegata al negativo dell'alimentazione). Supponendo di operare con una alimentazione di 5 V, il risultato è quello di avere su A0 una tensione data dalla formula.

Fig. 4
prototipi di keypad.

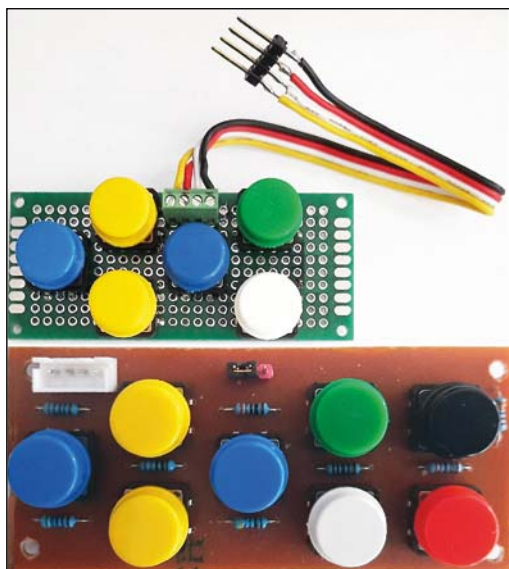


Fig. 5
Tabella
corrispondenze
valori keypad.

R	Ohm	P	Volt	ADC
R1	2000	1	0	0
R2	330	2	0,71	145
R3	470	3	1,43	293
R4	560	4	2,02	414
R5	680	5	2,52	517
R6	1000	6	3,02	618
R7	2200	7	3,62	741
R8	3300	8	4,05	830

$$V_{a0} = V \cdot \frac{R2}{R1+R2} = 5 \cdot \frac{330}{2000+330} = 0,71V$$

Premendo P3 dovremo considerare la R2 della formula appena vista come la somma di R2 ed R3, in quanto esse si vengono a trovare in serie verso la massa; quindi la formula precedente diventa

$$V_{a0} = V \cdot \frac{R2+R3}{R1+R2+R3} = 5 \cdot \frac{330+470}{2000+330+470} = 1,43V$$

e così via per gli altri pulsanti. Quindi diventa chiaro che, in base al pulsante premuto, cambia il rapporto di partitore e cambia di conseguenza il valore in tensione sull'uscita A0.

Collegando tale uscita ad un pin analogico di Arduino (nel nostro caso proprio A0) e leggendo il valore in tensione con l'ADC, otterremo una conversione in un numero tra 0 e 1023, che potremo poi usare nello sketch per stabilire quale funzione svolgere alla pressione dei singoli pulsanti. Con questa semplice tecnica in pratica si possono gestire molti pulsanti con un solo pin analogico, con notevole precisione. L'ultimo pulsante, P8, ha una caratteristica funzionale in più, rispetto agli altri 7, perché la presenza del jumper JP1 permette di stabilire a priori se deve essere l'ottavo pulsante del keypad, oppure se può essere utilizzato separatamente, collegandolo ad un pin digitale o, come nel nostro caso, al pin RESET di Arduino; in questo caso la pressione del pulsante porterà il pin RESET a livello logico 0, provocando appunto il reset della board. Questa è la ragione per la quale il connettore J1 porta all'esterno l'alimentazione, il pin A0 ed un altro pin che corrisponde appunto a P8, quando il

cappuccio del jumper è posizionato tra i suoi pin 1 e 2. Se invece il cappuccio mette in cortocircuito i pin 2 e 3, P8 opererà come gli altri 7 pulsanti.

La **Fig. 5** riporta una tabella con la corrispondenza tra partitore, pulsante premuto, tensione generata e valore ADC.

I valori riportati nelle colonne Volt e ADC sono strettamente correlati ai valori reali dei resistori che compongono il partitore, per tale ragione è importante utilizzare resistori di precisione con tolleranza massima del 2%; in ogni caso sarà difficile che ci sia una perfetta corrispondenza dei valori, ma a questo si rimedia con il software, come vedremo. Per comprendere bene il funzionamento del nostro keypad, che potrà tornare utile in altre occasioni, abbiamo messo a punto lo sketch "keypad_7_1.ino", scaricabile sempre dalla sezione download del sito www.elettronica.in. it sotto forma di file compresso; per testarlo sarà sufficiente collegare i pin A0, V+ e GND ai corrispondenti pin di Arduino, e settare JP1 su "P8" in modo da usare tutti i pulsanti col partitore, quindi basterà caricare lo sketch su Arduino e aprire il Monitor Seriale.

La schermata resterà vuota finché non si

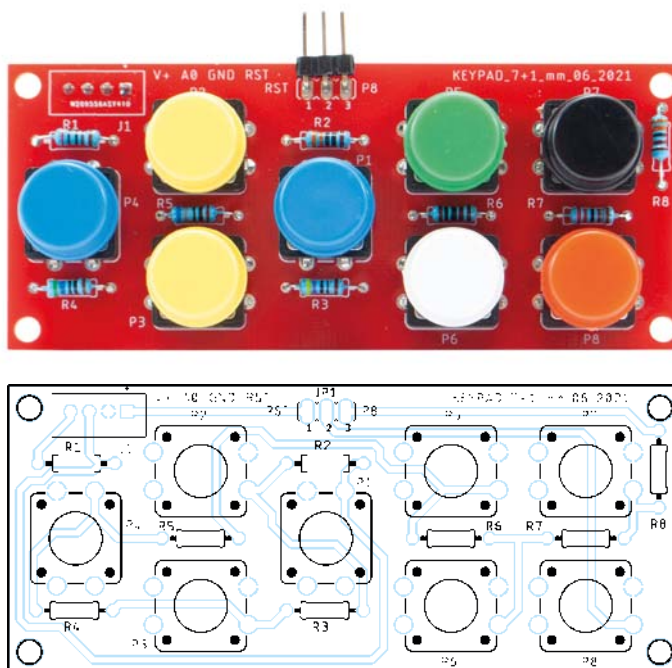
piano di MONTAGGIO

Elenco Componenti:

- R1: 2 kohm 1%
- R2: 330 ohm 1%
- R3: 470 ohm 1%
- R4: 560 ohm 1%
- R5: 680 ohm 1%
- R6: 1 kohm 1%
- R7: 2,2 kohm 1%
- R8: 3,3 kohm 1%
- P1÷P8: Pulsanti switch p.12mm con cappuccio colorato
- J1: Connettore JST 4 poli p. 2,54
- JP1: Strip maschio 3 poli a 90°

Varie:

- Mini jumper
- Stampato S1579 (90x38)



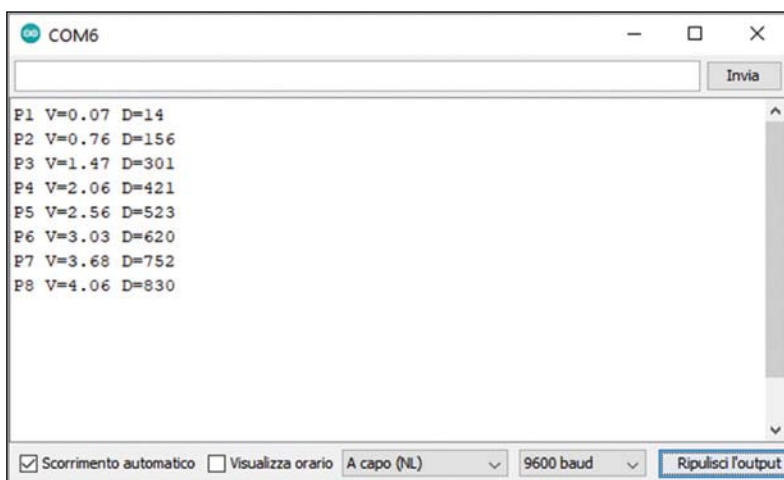


Fig. 6
Test del
keypad a 8
pulsanti con
Arduino UNO.

Fig. 7
Lo switch
case usato
per la lettura
dei pulsanti.

premerà uno dei pulsanti, a quel punto saranno visualizzati: il numero del pulsante P, la tensione letta V e la conversione decimale D (**Fig. 6**). Si può facilmente notare che i valori sono tutti diversi da quelli della nostra tabella, eccetto l'ultimo; peraltro riprovando dopo una decina di minuti in cui Arduino è rimasto acceso, facilmente questi valori saranno ancora leggermente diversi da quelli attuali; ciò è dovuto non solo alle tolleranze dei resistori ma anche alla temperatu-

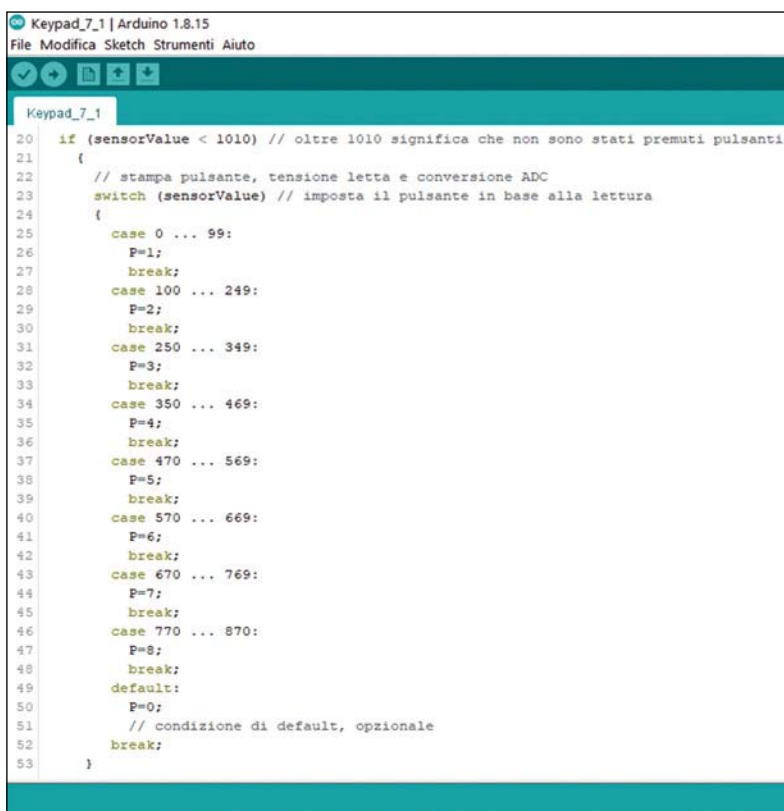
ra di lavoro del circuito, che influenza sia il valore dei resistori che, seppur di poco, il funzionamento dell'ADC integrato nel microcontrollore ATmega328P, il cuore di Arduino UNO. Diventa chiaro che impostando nel software i valori delle tabelle, molto raramente verrebbe riconosciuta la pressione dei singoli pulsanti, abbiamo quindi risolto il problema assegnando ad ogni pulsante un ampio range di valori decimali. In **Fig. 7** abbiamo riportato lo spezzone di sketch con il "trucco" utilizzato per risolvere il problema della tolleranza. Questa interessante sintassi permette di associare ad ogni "caso" un range di valori invece di un singolo valore, in pratica bisogna inserire il valore minimo e quello massimo da considerare validi ed interporre tra di essi i caratteri "..." aggiungendo uno spazio prima e dopo i tre puntini; per esempio il primo caso "0 ... 99" significa che qualsiasi valore decimale rientri in tra 0 e 99 sarà associato al pulsante P1, quindi che la lettura sia 0, come da tabella in **Fig. 5**, oppure 14, come nel test della **Fig. 6**, sarà sempre riconosciuta come pressione del pulsante P1.

La realizzazione del keypad è molto semplice, basta inserire i pochi componenti nel PCB, seguendo la serigrafia e saldarli; l'unica accortezza dovrà essere quella di non invertire i resistori tra loro, altrimenti il funzionamento risulterà non coerente con quanto abbiamo detto finora; se non ricordate la codifica a colori, consigliamo di misurarli con un multimetro digitale prima di saldarli.

IL DISPLAY LCD I²C-BUS

Per visualizzare i dati abbiamo utilizzato un display LCD 16x2 (LCD1602ABLU sul sito futurashop.it) al quale abbiamo saldato un'interfaccia I²C (LCDI2CINTERFACE). Il posizionamento dell'interfaccia sul retro del display permette il montaggio a pannello, ma bisogna tenerla sufficientemente distante dal display per evitare pericolosi cortocircuiti (**Fig. 8**).

Il collegamento con Arduino andrà effettuato come rappresentato in **Fig. 3**, quindi: VCC ai 5 V, GND al negativo di alimentazione, SDA al pin A4, SCL al pin A5. Per il suo funzionamento occorrerà utilizzare la libreria "LiquidCrystal_I2C.h", la cui ultima versione è scaricabile dal link <https://www.arduino.cc/reference/en/libraries/liquidcrystal-i2c/>; nel momento in cui scriviamo questo articolo è la versione 1.1.2, che abbiamo messo nella sezione download del sito www.elettronica.in sotto forma di file compresso.



COMANDI SOFTWARE DEL DFR0299

Poiché gestiremo il nostro modulo esclusivamente via software è necessario fare una panoramica delle istruzioni previste dalla libreria specifica. Nell'apposito riquadro abbiamo riportato l'elenco presente sulla wiki. Osservando questa lista è chiaro che, via software, è possibile gestire il modulo senza alcuna necessità di usare pulsanti esterni. Due note importanti per chi si cimerà a realizzare uno sketch: nei punti in cui si abbia necessità di eseguire due comandi in sequenza, dopo il primo occorre inserire una pausa di 100 ms, mediante un `"delay(100);"`, altrimenti la risposta del modulo non sarà coerente e si potrebbero ottenere risultati inattesi. Per quanto riguarda i nomi dei brani, la libreria gestisce solo valori numerici, però tollera la presenza del testo dopo il valore numerico, per esempio `"0001_nel_blu_dipinto_di_blu.mp3"`; solo che non c'è modo di leggerlo e visualizzarlo, quindi l'unica possibilità resterebbe quella di un database al quale accedere una volta selezionato il brano; noi abbiamo rinominato 35 file mp3 e poi abbiamo scattato una snapshot dell'elenco, forse una modalità non troppo ortodossa, ma sicuramente funzionale, per i nostri scopi. Per utilizzare l'hardware, che abbiamo realizzato in base alla **Fig. 3**, abbiamo scritto uno sketch dimostrativo, ricorrendo ad alcuni dei comandi elencati precedentemente, ed associandoli ai pulsanti del keypad, quindi visualizzando le informazioni sul display LCD. Lo sketch è denominato `"MP3_LCD_I2C_k5-1T.ino"` ed è scaricabile dalla sezione download del sito www.elettronica.in sotto forma di file compresso.

Prima di eseguire il nostro test dobbiamo inserire un certo numero di file mp3 in una micro SD card con capacità massima di 32GB, formattata FAT32. Affinché il test abbia successo occorre seguire rigorosamente questi passaggi:

- a) Creare una cartella "mp3" sulla micro SD card; il nome della cartella è obbligato, usandone un altro lo sketch non funzionerà;
- b) Selezionare sul PC (o sullo smartphone) i brani mp3 da usare per il test e rinominarli anteponendo nel nome la codifica xxxx, iniziando da 0001, senza saltare numeri, per esempio: 0001_Il cielo in una stanza.mp3, 0002_Piazza Grande.mp3, ecc.;
- c) Caricare i file sulla micro SD card, uno per volta, seguendo l'ordine numerico; un errore in questa fase significa perdere la corrispondenza tra la numerazione reale e quella

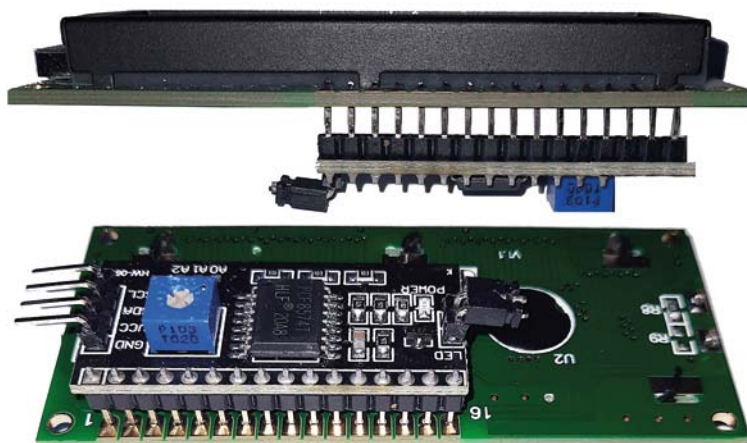
visualizzata sul display, durante l'esecuzione del brano. Questa cosa è specificata nella wiki: in pratica la numerazione garantisce l'esecuzione del file quando viene chiamato direttamente con l'apposito comando, mentre l'esecuzione in sequenza dei file segue l'ordine di caricamento;

- d) Inserire la micro SD card nello slot del modulo DFR0299;
- e) In qualsiasi momento sarà possibile, successivamente, aggiungere altri brani alla cartella, ma sempre seguendo queste due semplici regole.

Alla presa jack stereo 3,5 mm si può collegare una cuffia o degli auricolari, o ancora un amplificatore audio esterno, con le proprie casse. A questo punto disponiamo di tutto il necessario per eseguire il nostro test finale, vediamo intanto le fasi iniziali.

- 1 – Collegare Arduino UNO (o altra board basata su ATmega328) al PC, aprire l'IDE e settare correttamente la board e la porta COM;
- 2 – Scaricare, decomprimere e posizionare lo sketch `"MP3_LCD_I2C_k5-1T.ino"` nella cartella degli sketch di Arduino, quindi aprirlo nell'IDE;
- 3 – Scaricare le due librerie `"DFRobotDFPlayerMini-1.0.3.zip"` e `"LiquidCrystal_I2C-1.1.2.zip"` e posizionarle in un percorso facilmente raggiungibile (ad esempio la cartella Documenti o il Desktop);
- 4 – Usare il comando Sketch → #include libreria → Aggiungi libreria da file .ZIP..., quindi localizzare ed aggiungere le due librerie all'IDE (**Fig. 9**);
- 5 – Se le operazioni sono state eseguite correttamente, ora sarà possibile caricare lo sketch nell'Arduino UNO.

Fig. 8
Il display LCD 16x2 ad interfaccia I²C.



6 – A caricamento avvenuto, dopo alcuni istanti, sul display LCD appariranno in sequenza le due immagini visibili in **Fig. 10**.

La seconda schermata indica sulla prima riga il numero di brani caricati nella cartella mp3 (nel nostro caso sono 35), e sulla seconda riga la modalità di equalizzazione impostata via software ed il volume di partenza.

FUNZIONAMENTO DELLO SKETCH DI TEST

Il modulo, inizialmente (ossia dopo essere stato alimentato) in modalità "Stop"; da qui, per eseguire un brano si può ricorrere al pulsante "Play-Pause" che eseguirà sempre il brano 0001, oppure al pulsante "Next" che inizialmente partirà comunque dal brano 0001, o ancora dal pulsante "Previous", che partirà dall'ultimo brano dell'elenco (nel nostro caso 0035); subito dopo la pressione del pulsante prescelto il lettore inizierà l'esecuzione del brano ed il display visualizzerà il messaggio "PLAY XXXX" con l'indicazione del brano in esecuzione, ma senza gli zeri iniziali. Se

I COMANDI SOFTWARE DEL MODULO DFR0299

- `myDFPlayer.next();` //brano successivo
- `myDFPlayer.previous();` //brano precedente
- `myDFPlayer.play(X);` //X=primo brano da eseguire
- `myDFPlayer.loop(X);` //ripetizione continua del brano X
- `myDFPlayer.pause();` //pausa
- `myDFPlayer.start();` //play dopo la pausa
- `myDFPlayer.playFolder(X, Y);` //esegue il brano localizzato nella posizione X,Y (X=cartella (1÷99), Y=nome brano (001÷255) es: (15,4) SD:/15/004.mp3
- `myDFPlayer.enableLoopAll();` //esegue in loop tutti i brani
- `myDFPlayer.disableLoopAll();` // interrompe il loop di tutti i brani
- `myDFPlayer.playMp3Folder(X);` //esegue il brano X (0000÷3000) della cartella "mp3"; es. SD:/MP3/0004.mp3
- `myDFPlayer.advertise(X);` // esegue il brano X (0000÷3000) della cartella "advert"; es. SD:/ADVERT/0003.mp3
- `myDFPlayer.stopAdvertise();` //interrompe l'esecuzione del comando advertise
- `myDFPlayer.playLargeFolder(X, Y);` //esegue il brano localizzato nella posizione X,Y (X = cartella (1÷10), Y = nome brano (0001÷1000) es: SD:/02/950.mp3, cartella n.2, brano n.950)
- `myDFPlayer.loopFolder(X);` // esegue in loop tutti i brani della cartella X; es. cartella 5: X=05
- `myDFPlayer.randomAll();` // esegue in random tutti i brani
- `myDFPlayer.enableLoop();` //abilita il loop per il singolo brano
- `myDFPlayer.disableLoop();` //disabilita il loop del singolo brano
- `myDFPlayer.readCurrentFileNumber();` //legge il numero del file in esecuzione
- `myDFPlayer.readFileCounts();` //legge il numero di file presenti nell'SD
- `myDFPlayer.readFileCountsInFolder(X);` //legge il numero di file presenti nella cartella X (solo valori numerici)
- `myDFPlayer.readState();` //legge lo stato del

lettore: 512=Stop; 513/529=Play; 514=Pause; -1=passaggio al brano successivo – NB: questi sono i valori del modulo originale DFRobot, quelli compatibili forniscono risultati completamente differenti

- `myDFPlayer.volume(X);` //impostazione volume iniziale (X=0÷30)
- `myDFPlayer.volumeUp();` //aumenta volume
- `myDFPlayer.volumeDown();` //diminuisce volume
- `myDFPlayer.readVolume();` //legge il valore del volume attuale (0÷30)

//Controllo MP3

- `myDFPlayer.enableDAC();` //Abilita il DAC integrato
- `myDFPlayer.disableDAC();` //Disabilita il DAC integrato
- `myDFPlayer.sleep();` //attiva la modalità di risparmio energetico, spegne il lettore, per riattivarlo bisogna togliere l'alimentazione
- `myDFPlayer.outputSetting(true, 15);` // abilita l'uscita e setta il guadagno a 15
- `myDFPlayer.reset();` // resetta il modulo

//Imposta il dispositivo che contiene i file (SD è il default)

- `myDFPlayer.outputDevice(DFPLAYER_DEVICE_U_DISK);`
- `myDFPlayer.outputDevice(DFPLAYER_DEVICE_SD);`
- `myDFPlayer.outputDevice(DFPLAYER_DEVICE_AUX);`
- `myDFPlayer.outputDevice(DFPLAYER_DEVICE_SLEEP);`
- `myDFPlayer.outputDevice(DFPLAYER_DEVICE_FLASH);`

// Gestione Equalizzatore

- `myDFPlayer.readEQ();` //legge lo stato dell'equalizzatore (0=normal, 1= pop, 2=rock, 3=jazz, 4=classic, 0=bass[normal e bass restituiscono 0])
- Elenco modalità impostabili:
 - `myDFPlayer.EQ(DFPLAYER_EQ_NORMAL);`
 - `myDFPlayer.EQ(DFPLAYER_EQ_POP);`
 - `myDFPlayer.EQ(DFPLAYER_EQ_ROCK);`
 - `myDFPlayer.EQ(DFPLAYER_EQ_JAZZ);`
 - `myDFPlayer.EQ(DFPLAYER_EQ_CLASSIC);`
 - `myDFPlayer.EQ(DFPLAYER_EQ_BASS);`

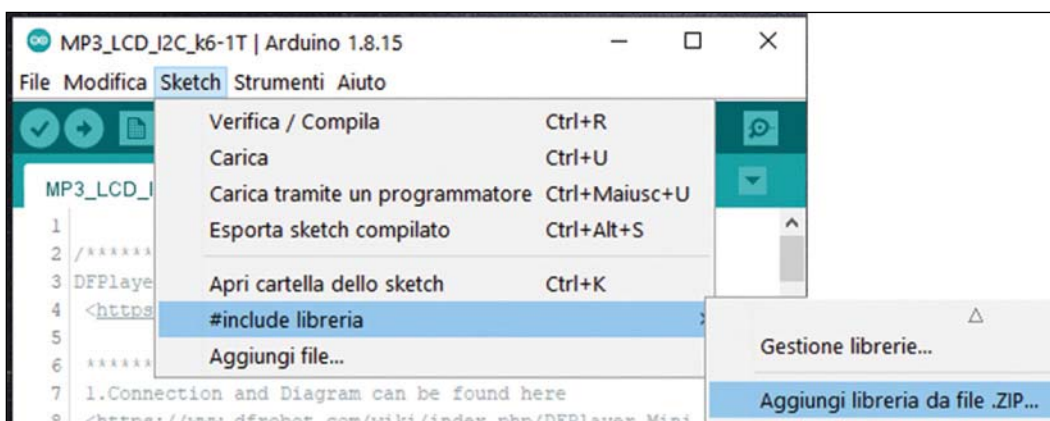


Fig. 9
Il caricamento
delle due
librerie.

in fase di esecuzione si dovesse premere il pulsante "Play-Pause" il modulo andrà in pausa, con la relativa indicazione sul display; per riprendere l'esecuzione del brano occorrerà ripremere il pulsante "Play-Pause". Premendo invece il pulsante "Next" si passerà immediatamente al brano numericamente successivo, così come il pulsante "Previous" farà eseguire il brano numericamente precedente. A fine brano il modulo andrà in modalità "Stop" (**Fig. 11**). I file vengono letti ciclicamente per cui, se si sta eseguendo l'ultimo brano, con "Next" si passa al primo file; se invece si sta eseguendo il primo brano, con "Previous" si passa all'ultimo. I pulsanti "Vol+" e "Vol-" agiscono sul volume del modulo, partendo dal valore iniziale settato nello sketch (20 nel nostro caso), l'escursione completa va da 0 a 30 (**Fig. 12**). Se la micro SD card non è stata inserita correttamente o è assente, all'avvio dello sketch il display mostrerà un avviso che inviterà ad inserirla correttamente e premere poi il pulsante abbinato al RESET di Arduino. Se invece essa viene tolta e/o reinserita durante il normale funzionamento il display mostrerà un messaggio che ne indica lo stato (**Fig. 13**).

DETTAGLI SOFTWARE

Vedremo ora i punti essenziali del nostro sketch, in modo da chiarire quali sono i comandi che abbiamo usato per gestire il modulo DFR0299, ma daremo anche alcune indicazioni sull'uso del keypad.

Nella parte iniziale dello sketch, quella riservata alle librerie ed alle dichiarazioni delle variabili e/o costanti, occorre inserire queste righe:

```
// gestione modulo DFR0299
#include "Arduino.h"
```

```
#include "SoftwareSerial.h"
#include "DFRobotDFPlayerMini.h"
SoftwareSerial mySoftwareSerial(5, 6); // RX, TX
DFRobotDFPlayerMini myDFPlayer;
```

Si tratta delle librerie essenziali per il funzionamento del modulo; la SoftwareSerial viene utilizzata per la comunicazione con il modulo,



Fig. 10
Avvio dello
sketch di
test.



Fig. 11
I messaggi di
esecuzione
dei brani.

Fig. 12
I messaggi relativi
al volume.



Fig. 13
I messaggi relativi
all'SDcard.



lasciando liberi i pin TX/RX di Arduino per l'eventuale attivazione del Monitor Seriale.

Il gruppo di righe riportato nel **Listato 1** serve per avvisare l'utente in caso di inserimento mancante o non corretto della micro SD card nello slot del modulo.

Serve inoltre per impostare il tipo di equalizzazione preferita nel modulo; sono sei righe, ma bisogna attivarne solo una, commentando le altre; nel nostro caso è attiva la modalità "JAZZ", corrispondente al n. 3, valore che, come abbiamo visto, viene visualizzato sul display all'accensione

del circuito. In ogni caso alla fine viene eseguito un "delay(100);" per le ragioni già spiegate in precedenza.

All'interno del LOOP vengono eseguite due funzioni: la lettura del keypad ed il controllo dello stato del modulo e dello slot per la micro SD-Card.

Per leggere il keypad, in questo caso, allo scopo di offrire una soluzione alternativa a quella già esposta sopra, abbiamo fatto ricorso ad una tecnica diversa. Nella sezione delle dichiarazioni abbiamo inserito un array di valori ed alcune variabili e costanti:

```
int adc_key_val[5] = {99, 249, 349, 469, 569};
int NUM_KEYS = 5;
int adc_key_in;
int key=-1;
int oldkey=-1;
```

In pratica abbiamo predisposto la lettura dei nostri cinque pulsanti, inserendo nell'array il valore massimo dei primi 5 range, visti nella parte dedicata alla descrizione del nostro keypad. Poi abbiamo aggiunto allo sketch una scheda "ADC_to_Number" con queste righe:

```
int get_key(unsigned int input)
{
    int k;
    for (k = 0; k < NUM_KEYS; k++)
    {
        if (input < adc_key_val[k]) // tasto riconosciuto
        {
```

↓ Listato 1

```
if (!myDFPlayer.begin(mySoftwareSerial))
{
    //Use softwareSerial to communicate with mp3
    lcd.print("Inserire SD card");
    lcd.setCursor(0, 1); //x=0, line=1
    lcd.print("e premere RESET");
    while(true);
}
Il seguente gruppo di righe:
//----Impostazione Equalizzatore interno----
// attivare una sola voce
// myDFPlayer.EQ(DFPLAYER_EQ_NORMAL); //0-default
// myDFPlayer.EQ(DFPLAYER_EQ_POP); //1
// myDFPlayer.EQ(DFPLAYER_EQ_ROCK); //2
// myDFPlayer.EQ(DFPLAYER_EQ_JAZZ); //3
// myDFPlayer.EQ(DFPLAYER_EQ_CLASSIC); //4
// myDFPlayer.EQ(DFPLAYER_EQ_BASS); //0 (uguale a NORMAL ma le impostazioni sono diverse)
delay(100);
```

```

        return k;
    }
}
if (k >= NUM_KEYS)
{
    k = -1;    // valore non valido
    return k;
}
}

```

Il funzionamento è molto semplice: ogni volta che si preme un tasto viene eseguita questa routine, che confronta il valore letto con quelli della matrice, iniziando dal più basso: se il valore letto è maggiore di quello col quale è confrontato la routine passa al valore successivo, se invece il valore è uguale o inferiore si ferma e restituisce il numero del pulsante associato.

Un piccolo esempio con i valori reali visti prima chiarirà le idee: immaginiamo di premere il pulsante P3, abbiamo visto (**Fig. 6**) che la lettura restituisce il valore 301; la routine confronta il valore 301 con il primo dell'array (0: 99), poiché $301 > 99$, passa al secondo valore (1: 249). Per la stessa ragione passa al terzo valore (2: 349), ma siccome $301 < 349$, significa che è stato premuto il pulsante P3 e quindi la variabile "k" assume il valore "2", che sarà poi utilizzato per eseguire le funzioni associate al pulsante P3; da notare che l'array va da 0 a 4, quindi $0 = P1 \dots 4 = P5$.

Nel LOOP viene effettuata una doppia lettura in seguito alla pressione di ogni pulsante, per garantire un corretto anti-debounce (anti-rimbalzo), ed eliminare quindi i "rimbalzi" dati dai contatti (che potrebbero essere interpretati come pressioni multiple del pulsante).

Vediamo ora i comandi associati ai cinque pulsanti:

- $0 = P1 = \text{myDFPlayer.next()};$ // passa al brano successivo
- $1 = P2 = \text{myDFPlayer.volumeUp()};$ // aumenta il volume
- $2 = P3 = \text{myDFPlayer.volumeDown()};$ // diminuisce il volume
- $3 = P4 = \text{myDFPlayer.previous()};$ // passa al brano precedente
- $4 = P5 =$ in questo caso il comando cambia in base allo stato attuale del modulo, che viene letto con il comando " $\text{myDFPlayer.readState()};$ " il quale restituisce uno tra i seguenti valori numerici:

-513 o 529 (è in Play) = $\text{myDFPlayer.pause()};$ // va in PAUSA

-514 (è in Pausa) = $\text{myDFPlayer.start()};$ //dopo la PAUSA riparte dal punto di arresto
 -512 (è in Stop) = $\text{myDFPlayer.playMp3Folder(1)};$
 // dopo lo stop esegue sempre il brano 0001.

La libreria del modulo DFR0299 è dotata di un sistema di debug che invia informazioni utili per la gestione di errori o condizioni di stato; ne abbiamo utilizzate tre, per controllare l'eventuale rimozione (*DFPlayerCardRemoved*) e/o reinserimento (*DFPlayerCardInserted*) della micro SDcard nello slot, e per sapere quando il lettore termina l'esecuzione di un brano e quindi va in modalità Stop (*DFPlayerPlayFinished*).

Abbiamo inserito queste tre informazioni in una routine collocata nella scheda "Debug". Tutte le altre tipologie di informazioni disponibili sono presenti negli esempi inseriti nella libreria del modulo, una volta compreso il meccanismo di uso visto in questo sketch, sarà facile utilizzarle.

CONCLUSIONI

Bene, con questo abbiamo terminato la nostra trattazione del modulo MP3 DFR0299 della DFRobot; speriamo di essere stati esaustivi e di avervi fornito sufficienti informazioni da permettervi, se lo desiderate, di utilizzarlo agevolmente nei vostri progetti futuri.

A tal proposito vi anticipiamo che abbiamo realizzato un progetto che sfrutterà tutte le potenzialità del modulo mp3, ma notevolmente ampliato rispetto al circuito di test, usato in questo articolo. □



Cosa occorre?

I componenti utilizzati in questo progetto sono disponibili presso Futura Elettronica. Il modulo lettore MP3 (cod. DFR0299) è in vendita a Euro 12,00, la demoboard (cod. FT1280K) è disponibile a Euro 16,00. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

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





IMPARA L'ELETTRONICA FACILMENTE CON TOTEM MINI LAB



€ 112,00
cod. TOTEMMINILAB

Piccolo ma potente laboratorio di elettronica ideale per sperimentare e apprendere le basi dell'elettronica. Pienamente compatibile con l'ambiente di programmazione Arduino, Totem Mini Lab comprende una varietà di strumenti di misura e blocchi elettronici pronti all'uso per realizzare esperimenti. Pensato per essere utilizzato insieme al sistema di costruzione Totem consente agli utenti di costruire solidi banchi da lavoro con breadboard integrate. Inoltre, il laboratorio è compatibile con schede di espansione I/O, sensori e audio che possono espandere ulteriormente le capacità del Mini Lab. L'utilizzo di Totem Mini Lab è semplice e divertente, ideale sia per ragazzi sia per adulti!

COMPLETALO CON LE SCHEDE DI ESPANSIONE!



ESPANSIONE AUDIO

Con questo pannello sarà possibile avere disponibili e facilmente accessibili i seguenti componenti: amplificatore audio in classe AB con 1 W di potenza, altoparlante integrato e alimentazione a 12 volt, VU-meter con barra a 10 LED, generatore di funzioni in grado di generare segnali fino a 1 MHz.

€ 45,00
Cod. TOTEMAUDIOSIDE



ESPANSIONE I/O

Contiene componenti elettronici che solitamente sono difficili da inserire sulle breadboard. Con questo pannello sarà possibile avere disponibili e facilmente accessibili i seguenti componenti: 3 interruttori a pulsante con indicatori LED, 3 potenziometri da 10 kOhm, 1 encoder rotativo, 3 pulsanti, 1 LED RGB, 1 presa Jack audio da 3,5 mm, 1 Relè da 10 A / 250 V, con LED di stato.

€ 37,00
Cod. TOTEMIOSIDE



ESPANSIONE SENSORI

Con questo pannello sarà possibile avere disponibili e facilmente accessibili i seguenti componenti: un microfono preamplificato, un display OLED 128x64 con interfaccia I²C, un sensore umidità DHT11, un termistore NTC da 10 kohm, un buzzer e un piccolo driver per motori DC.

€ 39,00
Cod. TOTEMSENSORSIDE

SMART DIMMER

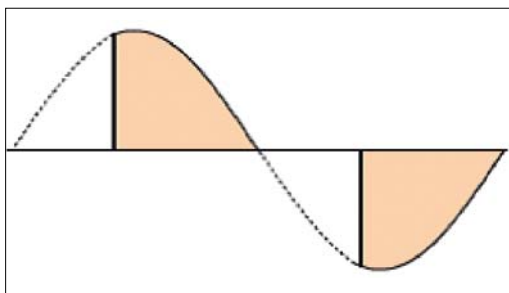
di MAURIZIO ŠKERLIČ



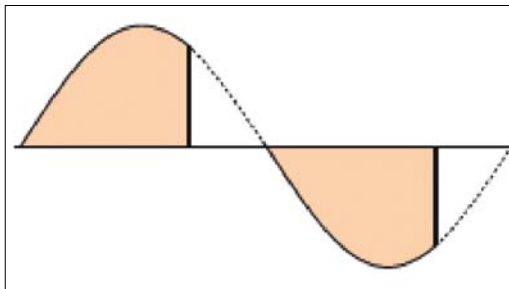
Regola l'intensità luminosa di lampade a incandescenza, CFL e a LED dimmerabili, controllato da smartphone, localmente o tramite assistente vocale.

La rapida diffusione degli assistenti vocali, come ad esempio Amazon Alexa, ha visto nascere parallelamente una serie di dispositivi interfacciabili ad essi, capaci di trasformare la nostra abitazione in una casa Smart. In questo articolo vi spiegheremo come realizzare uno smart dimmer e in che modo interfacciarlo con l'assistente vocale Amazon Alexa oppure Google assistant, per regolare la luminosità delle lampade con semplici comandi vocali. Immaginate di rientrare a casa dopo aver fatto la spesa con le mani piene di borse accendere la luce diventa un'impresa ardua; con l'assistente vocale vi basta semplicemente pronunciare un comando vocale non solo per accendere e regolare l'intensità della luce, ma anche aprire la porta del garage, accendere la televisione, impostare la temperatura del climatizzatore ecc. Praticamente è possibile comandare

→ **Fig. 1**
Funzionamento della tecnologia leading edge, conosciuta anche come forward phase.



→ **Fig. 2**
Funzionamento della tecnologia trailing edge, conosciuta anche come reverse.



qualsiasi dispositivo o elettrodomestico. In passato, regolare la luminosità di lampade a incandescenza era molto semplice e conveniente, utilizzando la tecnologia leading edge con TRIAC (acronimo di TRIode for Alternating Current). Con l'avvento delle lampade fluorescenti compatte CFL (Compact Fluorescent Lamp) e successivamente con quelle a LED, utilizzare tale tecnologia non è più possibile e bisogna ricorrere alla regolazione trailing edge, più complessa e più costosa da realizzare, che fornisce un controllo della luminosità più fluido senza generare fastidiosi ronzii tipici dei dimmer a TRIAC.

FUNZIONAMENTO DEI DIMMER

I dimmer per lampade ad incandescenza spengono l'alimentazione al corpo illuminante durante una parte del ciclo di alimentazione. Accendendo e spegnendo rapidamente il corpo illuminante due volte per ogni ciclo della tensione di alimentazione, 100 volte al secondo per gli impianti a 50Hz e 120 volte per quelli a 60 Hz, riducendo l'energia totale erogata. Maggiore è la porzione del ciclo durante

la quale la corrente è interrotta, più debole sarà la luminosità. Nelle lampade a incandescenza, la persistenza termica del filamento attenua questi impulsi, assicurando che lo sfarfallio dovuto dalla rapida commutazione sia impercettibile all'occhio umano.

I dimmer possono essere suddivisi in due categorie: quelli che funzionano con angolo positivo cioè taglio inizio fase (forward phase), noto come regolazione "leading edge", e quelli a funzionamento con angolo negativo, cioè con taglio fine fase (reverse phase), noto come regolazione "trailing edge".

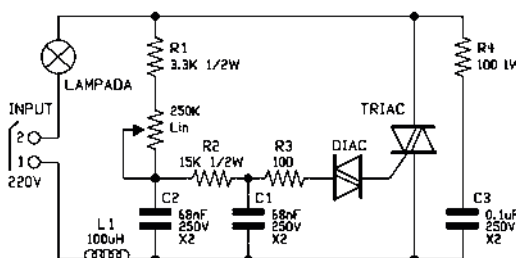
I dimmer "leading edge" funzionano variando il punto di accensione della corrente di alimentazione del circuito di illuminazione, nel senso che rilevano l'inizio, cioè il punto di passaggio per lo zero di ogni semi ciclo della tensione di alimentazione, quindi attendono un intervallo di tempo prima di applicare la corrente. Viene parzializzata la parte iniziale cioè la parte ascendente della semionda e spenta al passaggio per lo zero, vedi **Fig. 1**. Poiché la corrente non viene fornita quando la tensione passa per lo zero, l'accensione del corpo illuminante non è graduale ma brusca, con picchi di corrente che purtroppo ne riducono la vita.

I dimmer "trailing edge" accendono la corrente al passaggio della semionda per lo zero e variano il punto di spegnimento, interrompendo la corrente di alimentazione al circuito di illuminazione ad intervalli predeterminati. Viene parzializzata la parte finale discendente della semionda, per cui la corrente elettrica viene tolta prima del passaggio per lo zero, vedi **Fig. 2**. Dato che il corpo illuminante viene acceso con il passaggio della semionda per lo zero, l'accensione sarà graduale e comporterà meno stress, prolungando la vita della lampada. I dimmer di tipo leading edge sono quelli più semplici da realizzare, perché si ottengono utilizzando un TRIAC e alcuni componenti passivi (un esempio è lo schema di **Fig. 3**); sono più economici rispetto a quelli di tipo trailing edge, che risultano più complessi da realizzare.

SCHEMA ELETTRICO

Incominciamo con l'analisi dello schema elettrico, il cui elemento principale è il modulo ESP03, siglato U4 e basato sul microcontrollore Espressif ESP8266, equipaggiato con una memoria Flash da 1MB. La regolazione dell'intensità luminosa è affidata al circuito integrato FL5150 (in commercio esiste la variante FL5160, progettata per funzionare con correnti alternate a 60Hz). Il chip FL5150 può funzionare sia in modalità leading che trailing

→ **Fig. 3**
Semplice dimmer con Triac di tipo leading Edge.



edge, è protetto da sovracorrenti ed è capace di pilotare transistor sia IGBT sia MOSFET. L'integrato va alimentato dal piedino 6 con una tensione non superiore ai 17 V (al suo interno è presente un diodo Zener che va in conduzione se superiamo questo valore) che serve ai driver per riuscire a pilotare i gate dei due transistor in uscita. Al suo interno troviamo pure un regolatore di tensione da 5 V, la cui uscita fa capo al piedino 3. La regolazione dell'intensità luminosa avviene tramite un potenziometro da collegare tra il piedino DIM e il GND. Per poter regolare la luminosità della lampadina con l'assistente vocale, all'inizio abbiamo pensato di sostituire il potenziometro analogico con uno digitale, però ci siamo accorti che quasi tutti quelli reperibili in commercio hanno una risoluzione bassa a 8 bit che permette una regolazione di soli $2^8=255$ livelli, rendendo la regolazione luminosa a scatti. Per ottenere una regolazione più fluida e lineare, abbiamo optato per un convertitore analogico digitale DAC (*digital to analog converter*) a 12 bit MCP4725, nello schema elettrico siglato U2, con il quale siamo riusciti ad ottenere una risoluzione della regolazione luminosa di $2^{12}=4096$ livelli. Sostituendo il potenziometro analogico (meccanico) con uno digitale possiamo regolare la luminosità mediante un pulsante, semplificando l'impianto elettrico domestico, soprattutto nei corridoi o nelle trombe delle scale, dove il corpo illuminante deve essere comandato da più punti. In questo modo il costo dell'impianto sarà inferiore poiché non si utilizzeranno più i deviatori e tantomeno i costosi invertitori.

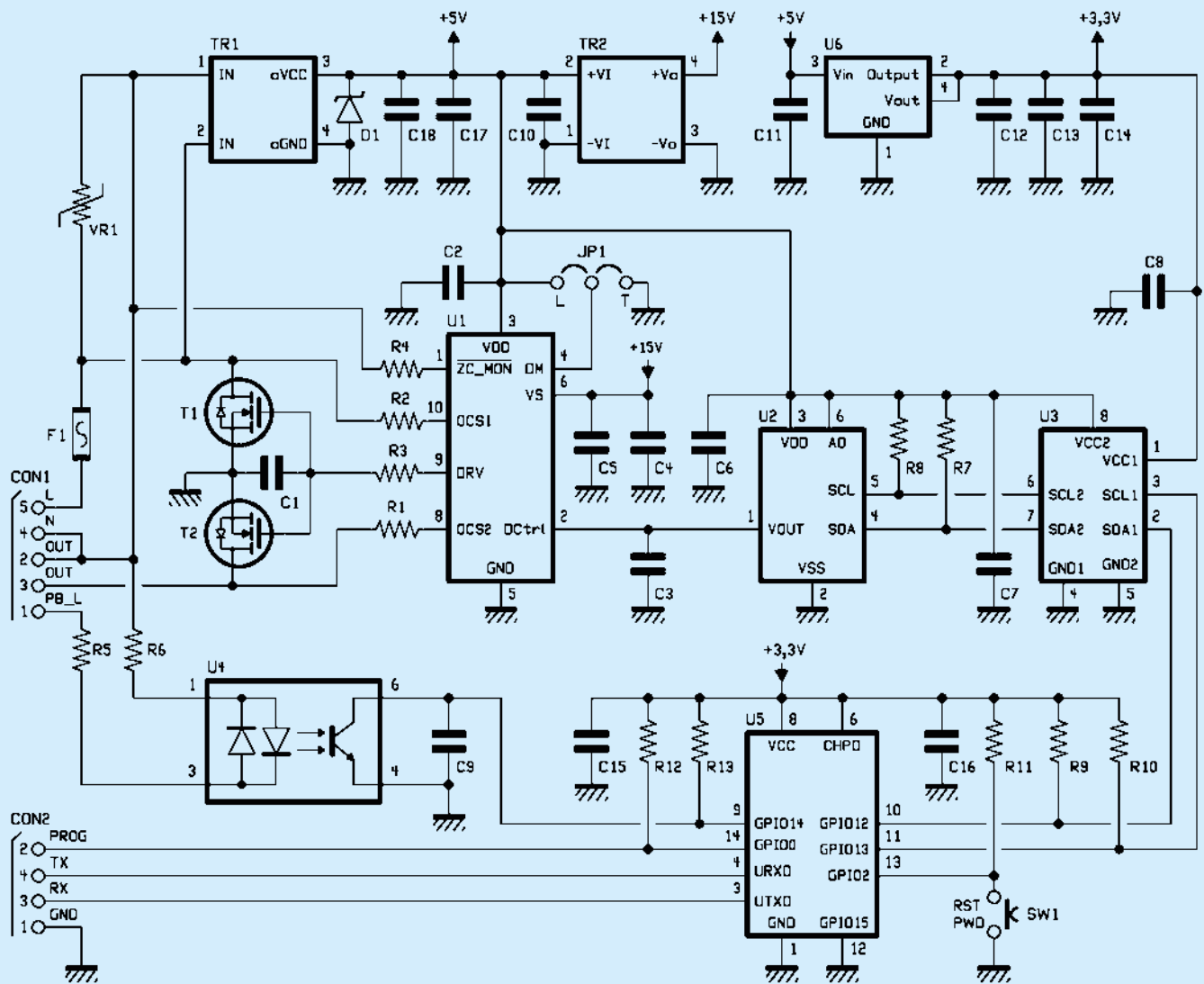
L'integrato MCP4725 preleva l'alimentazione a 5V direttamente dal piedino 3 del chip FL5150 fornita dal regolatore di tensione lineare interno. Poiché entrambe i chip MCP4725 e FL5150 sono collegati direttamente alla tensione di rete a 220 V, per scongiurare eventuali folgorazioni, soprattutto in fase di programmazione, abbiamo dovuto separarli galvanicamente dal resto del circuito. Per isolare il bus I²C abbiamo utilizzato il circuito integrato ISO1541D, nello schema elettrico siglato U3, mentre per alimentare l'FL5150 abbiamo utilizzato un alimentatore step-up galvanicamente isolato da 15 V, nel circuito elettrico siglato TR2. Anche se l'FL5150 viene alimentato con una tensione inferiore a quella consigliata dal costruttore, i driver riusciranno comunque a pilotare i moderni MOSFET e IGBT. Da prove effettuate, sono sufficienti addirittura 12 V. Quindi chi non riesce a reperire in commercio un alimentatore step-up galvanicamente isolato da 15 V può sostituirlo con uno da

12 V. Il circuito viene alimentato da uno switching da 5 V 1 A e poiché il modulo ESP03 lavora a 3,3 V il voltaggio viene abbassato mediante un regolatore di tensione lineare, nello schema elettrico siglato U6. Potevamo evitare di utilizzare il regolatore di tensione U6 e usare direttamente un alimentatore switching da 3,3 V, però ci siamo accorti che il ripple generato dall'alimentatore switching oppure piccole fluttuazioni di tensione, provocano malfunzionamenti del modulo ESP03, in particolar modo quando il modulo WiFi trasmette alla massima potenza e il consumo raggiunge picchi di corrente alti. Partendo con una tensione più alta e utilizzando un regolatore da 3,3 V la tensione di alimentazione risulta priva di disturbi e rimane stabile in tutte le condizioni, anche in quelle più gravose. Poiché i microprocessori Espressivo sono conosciuti per la loro sensibilità alle variazioni di tensione e ai disturbi EMI, per evitare reset casuali provocati dal watchdog o crash di sistema, consigliamo sempre di aggiungere un condensatore minimo 100 uF e uno da 100 nF posti il più vicino possibile ai piedini di alimentazione del modulo ESP e mantenere il collegamento del piedino reset il più corto possibile. Il piedino 1 del circuito integrato FL5150 tramite la resistenza R4 capta il passaggio per lo zero della semionda, mentre la modalità di funzionamento viene impostata con il piedino 4. Se posto a livello basso viene impostato il funzionamento trailing edge, mentre a livello alto si sceglie il funzionamento leading edge. I gate dei due Transistor T1 e T2 sono pilotati dal piedino 9 tramite la resistenza R3 e dal condensatore C1. I piedini 8 e 10 sono collegati mediante le resistenze R1 e R2 rispettivamente ai Drain di T2 e T1, per monitorare l'estensione massima di corrente di carico e di conseguenza limitare la potenza dissipata. Se vengono rilevati 16 impulsi consecutivi di sovracorrente, l'FL5150 va in protezione e disabilita l'uscita DRV piedino 9. Per ristabilire il normale funzionamento bisognerà scollegare fisicamente il circuito dalla corrente elettrica. Il pulsante SW1, collegato al piedino GPIO2 del modulo ESP03 serve a reimpostare la password predefinita di sistema, "123456789", in caso dimenticaste quella da voi impostata. Il foto accoppiatore U4 grazie al doppio diodo LED in entrata assieme alle due resistenze R1 e R2, ci permette di disaccoppiare la tensione alternata a 230 V fornita dal pulsante collegato al piedino 1 del connettore CON1 e la invia al piedino GPIO14 del modulo ESP03. La resistenza R13 serve da pullup e il condensatore C8 filtra il segnale in entrata, migliorando anche la funzione anti rimbalzo

CARATTERISTICHE TECNICHE

- Carico commutabile:
200 W
- Modalità di funzionamento:
leading edge o trailing edge
- Carico resistivi e induttivo.
- Compatibile con lampadine CFL e LED dimmerabili.
- Protezione termica e da sovracorrente.
- Memoria dell'intensità luminosa impostata.
- Accensione e spegnimento in dissolvenza.
- Timer spegnimento automatico.
- Impostazione luminosità minima.
- Eliminazione sfarfallii o tremolii della luce.
- Regolazione mediante pulsante o smartphone.
- Compatibile con assistenti vocali Amazon Alexa e Google Assistant.
- Configurazione tramite interfaccia web.
- Connessione WiFi.
- Protezione con password.





software. Il carico viene collegato tra i piedini 2 e 3 del connettore CON1 e la tensione di rete viene collegata ai piedini 4 e 5 rispettivamente neutro e fase. Il circuito è protetto dalle sovratensioni e sovracorrenti di rete mediante il varistore VR1 e il fusibile F1, mentre il diodo D1 protegge il circuito da una tensione eccessiva in caso di guasto del trasformatore.

REALIZZAZIONE PRATICA

Passiamo ora alla realizzazione pratica che richiederà una conoscenza minima nel montaggio SMD. Per chi fosse interessato a realizzare da solo lo stampato, facciamo presente che sul nostro sito web www.elettronica.in.it trovate disponibili per il download i file Gerber del circuito stampato a doppia faccia ottenibile per fotoincisione, nonché il firmware. 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 inoltre di usare un fluxante di ottima qualità che non lascia scorie residue, utile soprattutto se usate leghe saldanti senza piombo che notoriamente aderiscono con difficoltà e possono richiedere del fluxante aggiuntivo per ottenere buone saldature. Per posizionare i componenti aiutatevi con delle pinzette a punta fine possibilmente amagnetiche di acciaio inox oppure con le punte in plastica o ceramica, altrimenti magnetizzandosi i componenti saranno attratti dalle punte metalliche complicandovi il lavoro. Consigliamo sempre di iniziare a saldare i semiconduttori a montaggio superficiale come U1, U2, U3, U4, U6 e successivamente tutti i componenti passivi.

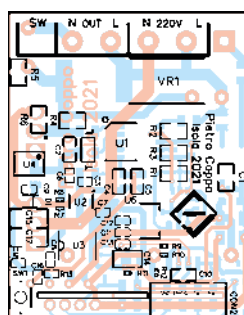
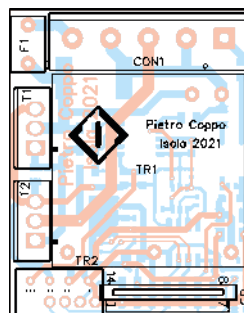
| piano di montaggio SMART DIMMER

Elenco Componenti:

R1, R2, R4: 2 Mohm 250 V (0805)
 R3: 1 kohm 250 V (0805)
 R5, R6: 150kohm 250 V (0805)
 R7, R8, R9, R10: 2,2 kohm (0402)
 R11, R12: 10 kohm (0402)
 R13: 470 kohm (0402)
 C1: 22 pF 250 V (0805)
 C2: 100 nF 25 V (0603)
 C4, C10: 4,7 μ F 25 V (0603)
 C5: 100 nF (0603)
 C3, C6, C7, C8, C13, C16, C18:
 100 nF (0402)
 C9: 1 μ F 6,3 V (0402)
 C11, C12: 10 μ F 10 V (0402)
 C14: 47 μ F 6,3 V (0805)
 C15: 100 μ F 6,3 V (1206)
 C17: 220 μ F 6,3 V (1206)
 D1: XBP1013
 T1, T2: SPI08N80C3XKSA1
 VR1: V430CH8
 TR1: RAC05-05SK
 TR2: ROE-0515S
 F1: F 1 A
 U1: FL5150
 U2: MCP4724A0
 U3: ISO1541D
 U4: TLP184
 U5: ESP-03
 U6: NCP1117ST33T3G
 CON1: Morsettiera 5 poli P - 5 mm

CON2: DIP-1x4P-2 mm
 SW1: pulsante SMD
 4,6 X 2,8 mm

Varie
 - stampato S1569 (41,5x33 mm)



Fate attenzione al corretto posizionamento dei semiconduttori nonché al corretto orientamento dei componenti polarizzati come visibile nel piano di montaggio. Nel saldare i componenti passivi consigliamo di stagnare prima quelli più piccoli con

case 0402 e successivamente passare a quelli con case 0603 e finire con quelli più grandi come il variatore VR1 e il pulsante SW1. Il modulo ESP03, che nello schema elettrico è siglato U5, andrà prima stagnato sul PCB adattatore (scheda adapter) e

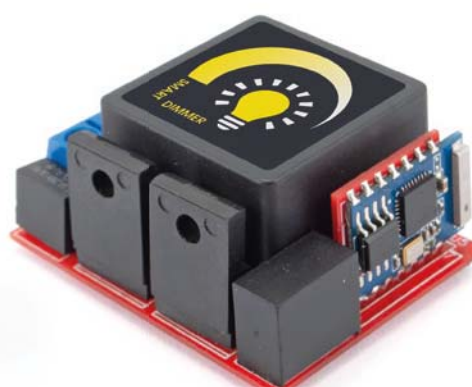
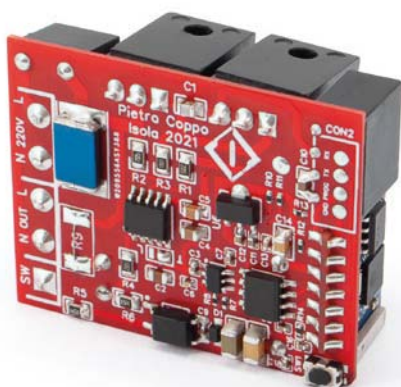
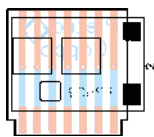


Fig. 4
 Il prototipo visto dalle due facce del PCB: notare l'ESP03 montato in verticale.



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

successivamente montato verticalmente nella fessura presente sul PCB dello Smart dimmer, come visibile in **Fig. 5**. Abbiamo optato per questo tipo di montaggio per ottimizzare e risparmiare lo spazio miniaturizzando il circuito. Passate ora al montaggio dei componenti a foro passante come il connettore CON1, il fusibile F1, i trasformatori TR1 e TR2 per finire con i due transistor T1 e T2. Poiché il connettore CON2 serve solo in fase di programmazione verrà saldato provvisoriamente per poi toglierlo definitivamente, oppure potete optare di saldare i fili direttamente sul circuito stampato. Lasciamo a voi la scelta. Se il carico in uscita assorbirà troppa corrente provocherà il surriscaldamento dei transistor T1 e T2 che dovranno essere raffreddati adeguatamente con un dissipatore di calore. Per il controllo delle comuni lampadine a LED, dove l'assorbimento di corrente è basso, il montaggio del dissipatore di calore non sarà necessario.

Se non usate transistor con case T0220F isolato in plastica, non toccate con le dita T1 e T2 quando il circuito è collegato alla rete elettrica, altrimenti rischiate di rimanere folgorati. Il nostro consiglio è di non toccare mai il circuito prima di scollegarlo dalla rete.

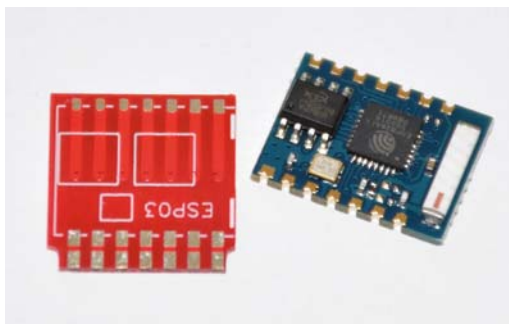
Una volta terminato il montaggio pulite bene il PCB con alcol isopropilico o in alternativa con un altro solvente, in modo da togliere ogni residuo di fluxante. Controllate attentamente, aiutandovi con una lente di ingrandimento, che non si siano formati ponticelli di stagno tra i piedini dei semiconduttori. Terminato questo passaggio potete alimentare il circuito collegando la fase ed il neutro della corrente elettrica rispettivamente ai piedini 5

e 4, una lampadina tra i piedini 2 e 3 e un pulsante sul piedino 1 del connettore CON1, come visibile nello schema elettrico.

PROGRAMMAZIONE

Il modulo ESP03 deve essere programmato mediante un adattatore USB per ESP8266 come quello a catalogo Futura Elettronica codice prodotto 8300-YM203, basato sul chip di conversione USB-TTL CH340G, oppure il modello *ESP8266 Debugger* codice prodotto 3085-ESP8266DEB, basato sul chip di conversione USB-TTL CP2014. In alternativa potete usare anche un qualsiasi convertitore USB TTL, purché compatibile con logica a 3,3 V, usato per programmare i moduli Arduino mini come quello a catalogo Futura Elettronica codice prodotto 1686-USBTOUART oppure quello usato da noi 2850-FT232USBQUART 1686-UC00A. Per il collegamento fate riferimento alla **Fig. 6**. Ricordatevi di collegare l'uscita TX del convertitore all'entrata RX dello Smart dimmer e l'entrata RX del convertitore all'uscita TX dello Smart dimmer. Per programmarlo dovete metterlo in modalità di programmazione premendo il pulsante *Prog* prima di allacciarlo alla corrente elettrica e rilasciarlo dopo un secondo. A differenza degli altri moduli ESP, il modulo ESP03 è sprovvisto del piedino reset, quindi terminata la programmazione dovete staccarlo e riattaccarlo alla corrente elettrica. In realtà un metodo alternativo ci sarebbe, collegando una resistenza di pullup tra positivo e il piedino CH_PD (*chip power down*), e portando il piedino a basso livello si ottiene il reset del microprocessore. Questo stratagemma potete usarlo nei vostri progetti solo in fase di programmazione, nel progetto definitivo collegate il piedino CH_PD direttamente al positivo, come indicato nel Datasheet, altrimenti il microprocessore diventa molto sensibile ai disturbi EMI e tende a riavviarsi spontaneamente come spiegato precedentemente nell'articolo. Poiché il modulo ESP03 monta un microprocessore ESP8266 potete programmarlo direttamente dall'ambiente di sviluppo Arduino Ide scaricando le definizioni della scheda dal sito http://arduino.esp8266.com/stable/package_esp8266com_index.json, indirizzo che dovete aggiungere tra gli "URL aggiuntivi per il gestore di schede" nel menu "impostazioni". Dal menu "sketch" selezionate "#include libreria" e cliccate su "Gestione libreria". Dalla finestra Gestione libreria cercate e installate le librerie "IoTWebConf" "SinricPro" e "Adafruit MCP4725", necessarie per compilare lo sketch "smartdimmer.ino" che andrete a scaricare dal sito

Fig. 5
Modulo ESP03
e adattatore per
il montaggio in
verticale sul PCB
dello Smart dimmer.



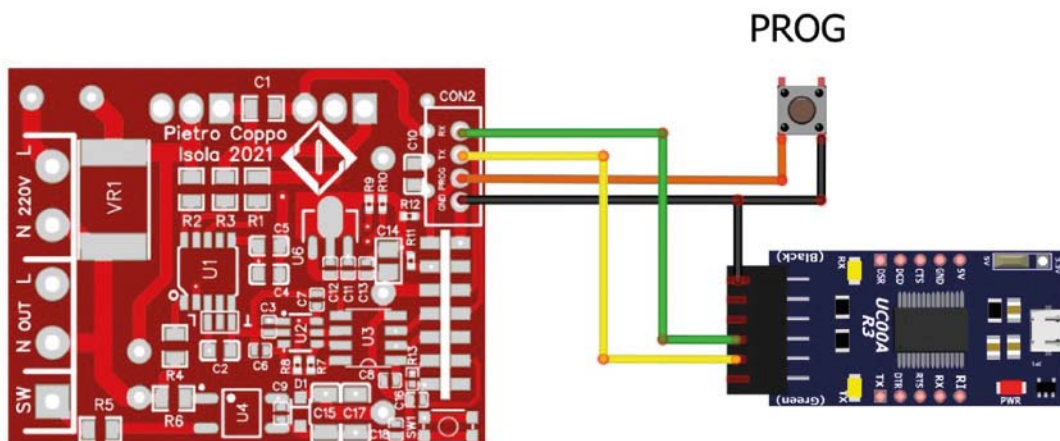


Fig. 6
Cablaggio dei pulsanti e del convertitore USB TTL allo Smart dimmer. Ricordate di impostare il voltaggio dell'adattatore a 3,3 V.

fritzing

web www.elettronica.in.it. Facciamo presente che la versione del package core ESP8266 caricato in Arduino Ide sia aggiornato alla 2.7.4, con le versioni successive, al momento della progettazione siamo arrivati alla 3.0.2, abbiamo riscontrato problemi con la connessione alla rete WiFi. A causa del continuo aggiornamento delle librerie e delle incompatibilità riscontrate facciamo presente che il sorgente del prototipo è stato compilato e testato con le librerie "lotwebconf" versione 3.1.0, "SinricPro" versione 2.9.7, "Adafruit MCP4725" versione 2.0.0 e "ArduinoJson" versione 6.18.3. Una volta compilato lo sketch potete caricarlo sullo Smart dimmer ricordando, prima di dare tensione al circuito, di premere il pulsante collegato al piedino GPIO0 (pin 3 sul connettore CON2) e rilasciarlo dopo un secondo. Per il corretto cablaggio del pulsante al connettore CON2 fate riferimento al cablaggio di **Fig. 6**. Una volta programmato il modulo ESP03 bisognerà resettarlo staccandolo dalla corrente elettrica. Con un computer o altro dispositivo provvisto di scheda WiFi fate una scansione delle reti wireless e selezionate quella con SSID *FuturaElettronica*, la password predefinita è "123456789". Lo Smart dimmer attiverà il "captive portal" all'indirizzo 192.168.4.1, e lancerà il browser predefinito di sistema visualizzando la pagina web di **Fig. 7**. Facciamo presente che negli smartphone e tablet, una volta scelta la rete SSID, il browser non si avvierà in automatico ma bisognerà cliccare sul pulsante *accedi alla rete*, oppure immettere nella barra degli indirizzi del browser l'indirizzo IP 192.168.4.1. Detto questo, per i meno esperti, consigliamo di effettuare questa configurazione mediante pc. Nella pagina iniziale viene visualizzato l'indirizzo fisico ossia l'indirizzo MAC (Media Access Control)

dello Smart dimmer. L'indirizzo MAC è un numero a 48 bit composto da 6 numeri esadecimali da 8 bit separati dai due punti. I primi 24 bit dell'indirizzo MAC rappresentano il numero ID del produttore dell'adattatore di rete. I secondi 24 bit rappresentano il numero di serie assegnato all'adattatore di rete dal produttore. Prendete nota di questo numero perché vi servirà in seguito. Cliccando sul pulsante di configurazione, nella sezione "System configuration" potete modificare il nome del dispositivo, modificare la password di sistema, inserire il nome e la password della rete SSID della rete WiFi di casa vostra e il "Startup delay". Quest'ultimo è il tempo in secondi che, ad ogni accensione o riavvio, lo Smart dimmer diventa un punto di accesso per permettere all'utente di raggiungere la home page, visibile in **Fig. 7**. Nella sezione "Device IP" selezionando la casella di spunta "Use static IP" potete disabilitare l'assegnazione dell'IP dinamico mediante il DHCP server e impostare quello statico. Nella casella "IP address" scrivete il numero IP che volete assegnare allo Smart dimmer, nella casella "Subnet mask" inserite



Fig. 7
Home page dello Smart dimmer, con la funzione web control attivata.



Fig. 8
Schermate della
pagina web di
configurazione
dello Smart dimmer.

System configuration

Thing name
FuturaElettronica

AP password

WiFi SSID

WiFi password

Startup delay (seconds)
30

Device IP

Use 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

Api Key

Api Secret

Settings

DNS
0.0.0.0

Iot Credentials

Api Key

Api Secret

Device Id

Auto off timer (0 - 1440 min)
0

Min brightness (0 - 20 %)
0

Fade delay (0 - 10)
0

Turn on at startup: ☐

Enable external pushbutton: ☒

Enable assistant control: ☒

Enable web control: ☐

Reboot device on save: ☐

Apply

Firmware config version '1.1 Pietro Coppo Isola 2021'

il numero IP della maschera di sottorete della vostra rete, nella casella "Gateway" inserite il numero IP del vostro router e nella casella "DNS" inserite il numero IP del server DNS (*domain name system*)

del vostro provider oppure il numero IP del vostro router. In alternativa potete utilizzare il numero IP di un servizio DNS pubblico come quello di Google, numero primario 8.8.8.8 e numero secondario 8.8.4.4, oppure quello di CloudFlare numero primario 1.1.1.1 e numero secondario 1.0.0.1, che tra i due risulta essere pure il servizio DNS pubblico gratuito con un ping medio migliore. Terminata la configurazione di sistema salvatela cliccando sul pulsante *Apply* e verificate se lo Smart dimmer si connette correttamente alla rete. Se avete scelto di ottenere il numero IP in modo dinamico, per conoscere quello che vi ha assegnato il server DHCP della vostra rete, avete bisogno del numero MAC che avete precedentemente annotato. Aprite il terminale "cmd" con i permessi di amministratore e digitate il comando "arp -d *" per cancellare tutti gli host memorizzati, poi fate un ping di broadcasting della vostra rete ad esempio con il comando "ping 192.168.1.255" e infine lanciate il comando "arp -a". A terminale comparirà l'elenco dei numeri MAC di tutti i dispositivi presenti nella vostra rete, cercate quello corrispondente allo Smart dimmer, e annotate l'indirizzo IP che vi servirà per accedere alla pagina web del dispositivo. Lo Smart dimmer per funzionare con l'assistente vocale necessita delle credenziali Iot.

Fig. 9
Pagina di login
del sito sinric.pro.

Sinric Pro

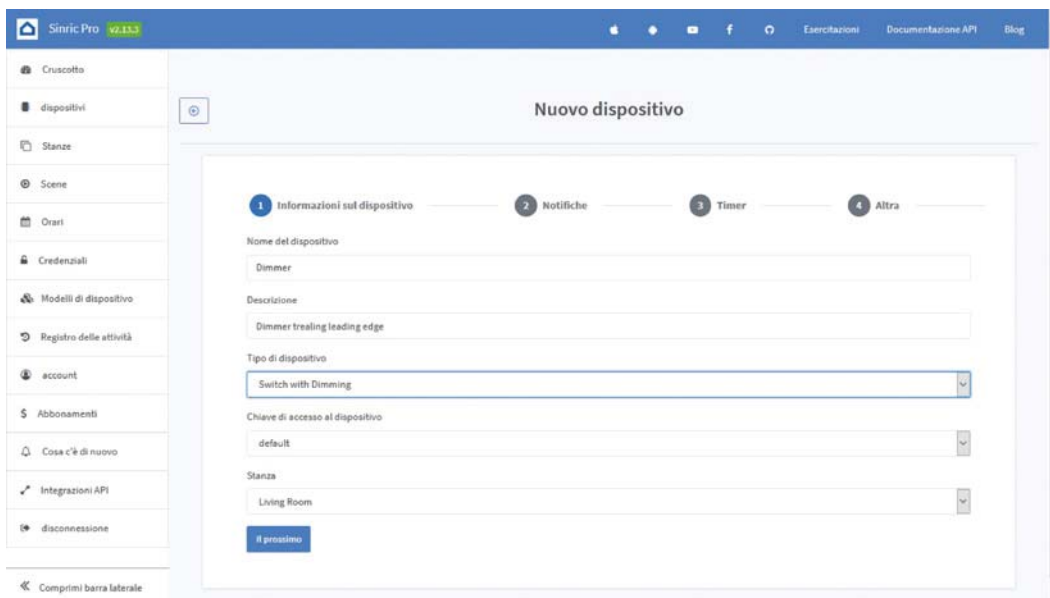
Email address
[redacted]@gmail.com

Password
[redacted] [I forgot password](#)

☒ Non sono un robot

reCAPTCHA
Privacy - Terms

Log in



Qui spiegheremo come ottenere le credenziali lot per il servizio Sinric pro e per dell'assistente vocale Amazon Alexa. Dal vostro browser, aprite la pagina web <https://sinric.pro> dove vi registrate e fate il login, come visibile in **Fig. 9**. Una volta entrati, dal menu dispositivi cliccate sul pulsante *"Aggiungi dispositivo"*, come visibile in **Fig. 10**: si aprirà una nuova schermata nella quale inserirete il nome del dispositivo e una breve descrizione, dal menu a tendina *"Tipo di dispositivo"* scegliete *"Switch with Dimming"*, dal menu a tendina *"Chiave di accesso al dispositivo"* scegliete *"default"* e dal menu a tendina *"Stanza"* scegliete ad esempio *"Living room"*,

come visibile in **Fig. 10**. Cliccando sulla pagina successiva potete attivare l'invio della notifica allo smartphone ad ogni accensione o spegnimento della luce. Continuando, potete impostare un timer di accensione o spegnimento e nell'ultima schermata potete attivare il calcolo approssimativo del consumo energetico.

Dal menu "*Credenziali*" della pagina web *Sinric.pro* ricopiate le chiavi "*App key*" e "*App Secret*", come visibile in **Fig. 11** e dal menu "*Dispositivi*" ricopiate l'*ID* del dispositivo che volete comandare, come visibile in **Fig. 12**. Queste chiavi dovete inserirle nella pagina di configurazione dello *Smart dimmer* alla

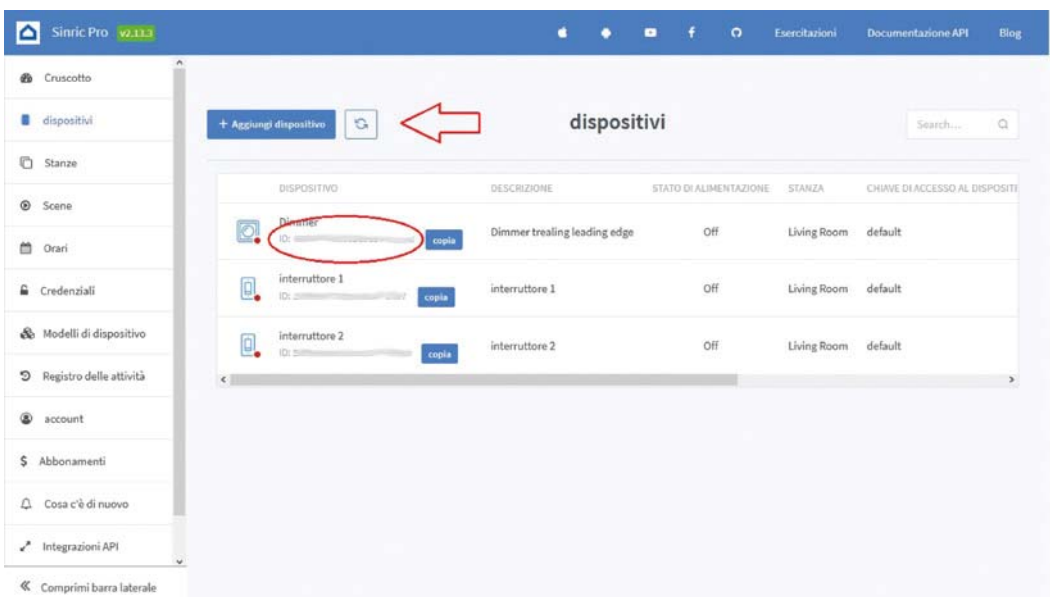
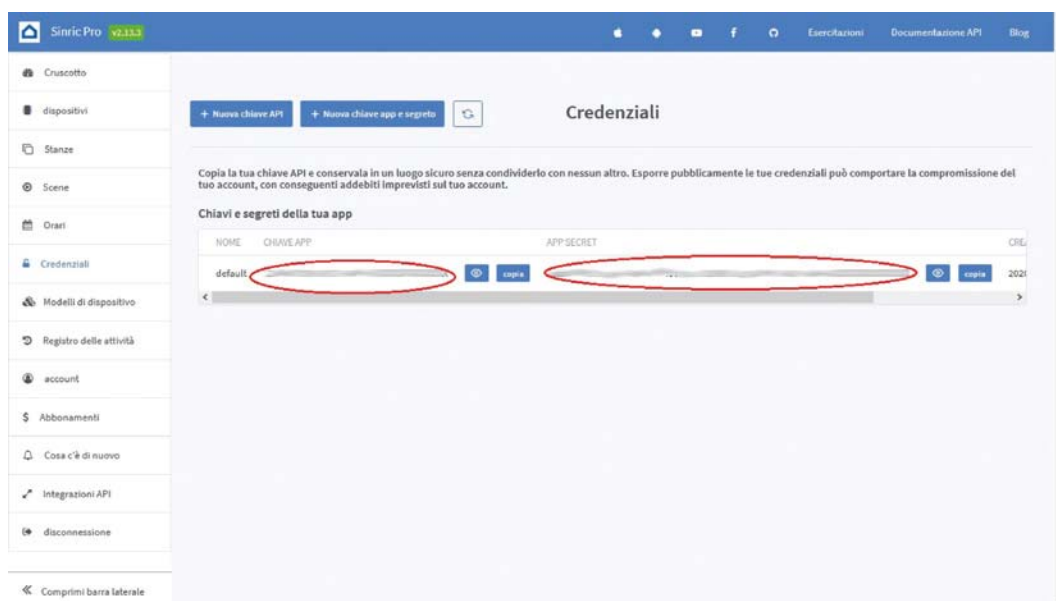


Fig. 12
Pagina delle
credenziali sul sito
(sinric.pro).



sezione "I tuoi dispositivi", rispettivamente nei campi "API Key", "API Secrets" e "Device ID" (fare riferimento alla Fig. 8).

A questo punto nel vostro smartphone o tablet dovete installare l'App Amazon Alexa, rammentando che per poterla scaricare e installare dovete disporre di un account Amazon, quindi se non lo avete, createlo.

Una volta entrati nell'App Amazon Alexa andate nel menu *home*, cercate e quindi installate la skill *Sinric Pro*. Spostatevi nel menu dispositivi e

aggiungete un nuovo dispositivo cliccando sull'icona in alto a destra come visibile in Fig. 13. Nella schermata successiva scegliere come dispositivo "altro" e successivamente cliccate sul tasto "trova dispositivi", come visibile in Fig. 13, quindi attendete la rilevazione del dispositivo.

A questo punto selezionate la voce "Configura dispositivo" e, se desiderate tenere i dispositivi organizzati per gruppo, scegliete l'opzione "Seleziona gruppo": potete scegliere tra *Salotto*, *Camera da letto*, *Cucina*, *Sala da pranzo*, *Ufficio*, ecc. oppure

Fig. 13
Procedura per
aggiungere un
dispositivo nell'app
Amazon Alexa.



creare un gruppo personalizzato. Per terminare la configurazione del vostro sistema basato su Alexa di Amazon, seguite le istruzioni che appaiono sullo schermo (fate riferimento alla **Fig. 14**, che propone quelle d'interesse).

Notate che non siete vincolati all'app Amazon Alexa, perché in alternativa a quest'ultima potete utilizzare convenientemente quella di Google Home e l'assistente vocale Google assistant; riferitevi alla **Fig. 15** ed alla **Fig. 16**, dove sono mostrate le schermate di configurazione e di utilizzo, rispettivamente per la prima e per la seconda soluzione offerte da "Big G".

Come alternativa al prodotto di Amazon e a quello di Google, potete impiegare l'app Sinric Pro, che è quella le cui schermate sono proposte dalla **Fig. 17**. A prescindere dallo strumento di interfaccia remota (l'app, in sostanza...) prima di utilizzare lo Smart dimmer non dimenticate di impostare il tipo di regolazione della luminosità desiderato, fra trailing edge e leading edge. L'impostazione del caso dev'essere effettuata a livello hardware, vale a dire sul circuito stampato del dimmer, ponticellando con una goccia di stagno la piazzola centrale con quella a destra o sinistra, come visibile in **Fig. 18**. Se volete regolare carichi resistivi come lampadine a incandescenza o alogene utilizzate la tecnologia leading edge; se invece volete regolare carichi induttivi come trasformatori, lampade CFL oppure lampadine LED impostate la tecnologia trailing edge. Inserite ora lo Smart dimmer in una scatola

di derivazione oppure direttamente nella scatola da incasso dell'impianto domestico sostituite l'interruttore con un pulsante seguendo il cablaggio sul connettore CON1, come riportato nello schema elettrico. Facciamo presente che in base alla norma CEI 64/8 questa prevede una classificazione degli impianti elettrici secondo tre livelli:

- Livello 1 - Base: livello minimo previsto;
- Livello 2 - Standard: per unità immobiliari con una maggiore fruibilità degli impianti, tenendo conto delle altre dotazioni impiantistiche presenti;
- Livello 3 - Domotico: per unità immobiliari con dotazioni impiantistiche ampie e innovative, ad esempio per sistema bus, e in particolare, la domotica.

Se il vostro impianto civile elettrico non è a norma CEI 64/8 livello tre, nelle scatole da incasso rettangolari portafrutto sicuramente manca il cavo conduttore blu del neutro, indispensabile negli impianti domotici. Quindi per far funzionare il nostro dispositivo Smart dovete far arrivare questo conduttore dal quadro elettrico o dal punto luce che volete comandare alla scatola da incasso portafrutto. Oltre ad essere molto pericoloso la legge italiana vieta al proprietario dell'impianto, se non possiede la qualifica professionale per farlo, di rifare l'impianto per conto proprio. Nel caso ristrutturare la vostra abitazione e volete rendere l'impianto elettrico a norma CEI 64/8 livello tre, vi consigliamo di far installare all'elettricista in ogni

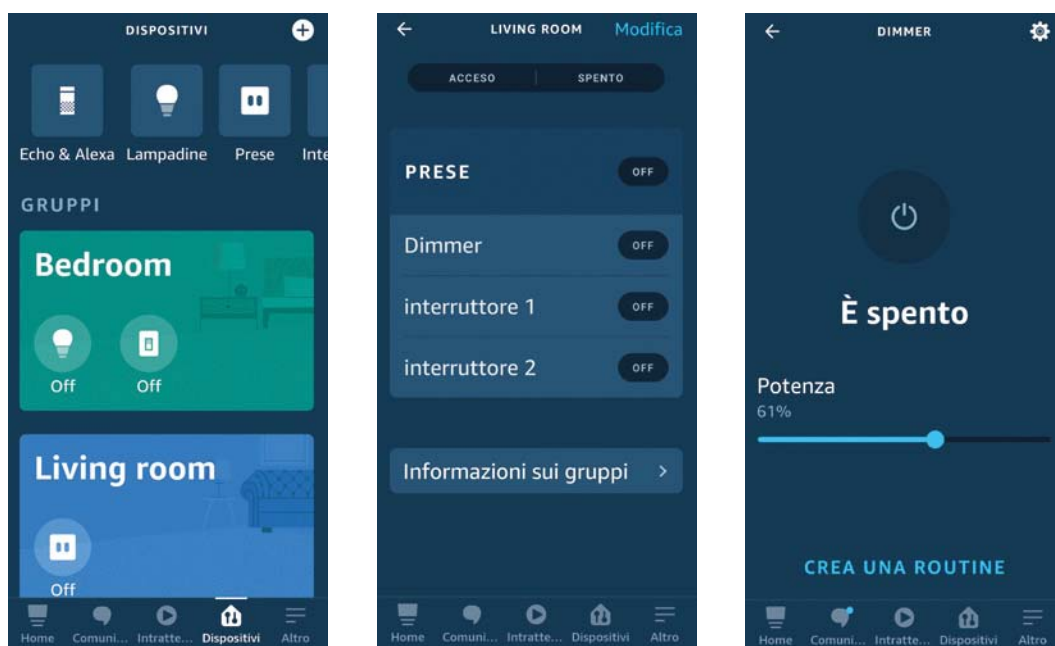


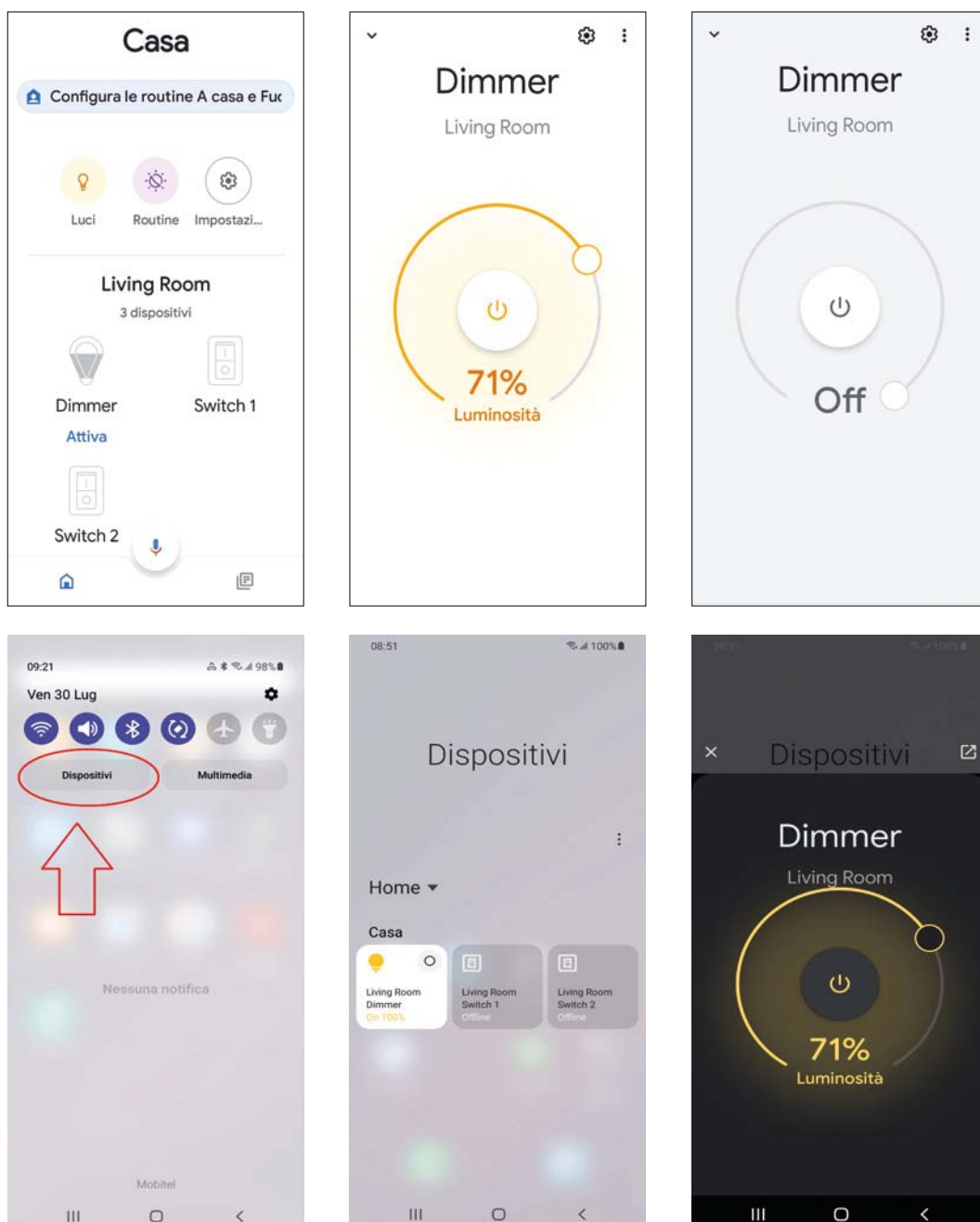
Fig. 14
Interfaccia per il controllo dello Smart dimmer con l'app Amazon Alexa.

scatola da incasso anche il cavo conduttore del neutro, cioè quello di colore blu, così con una spesa irrisoria aumenterete anche il valore dell'immobile investendo per il futuro.

Tra i file scaricabili dal sito web www.elettronica.in.it nella cartella STL trovate i file per la stampa 3D degli adattatori e frutti per i supporti Vimar Plana, Vimar Idea, Gewiss System, Gewiss PlayBus e bTicino visibile in **Fig. 19**. Questo faciliterà l'installazione dello Smart dimmer nel vostro impianto

elettrico domestico. Tra i file STL troverete pure un contenitore per il montaggio nelle scatole di derivazione oppure direttamente nelle scatole rettangolari da incasso. Facciamo presente che l'adattatore Vimar Idea viene montato su un copri foro originale al quale devono essere asportati precedentemente i segmenti di rinforzo come visibile in **Fig. 20**. Aiutatevi con le tronchesine per tagliare i reofori dei componenti oppure un paio di pinze. Fate attenzione al corretto orientamento

Fig. 15
Interfaccia dell'app
Google Home.



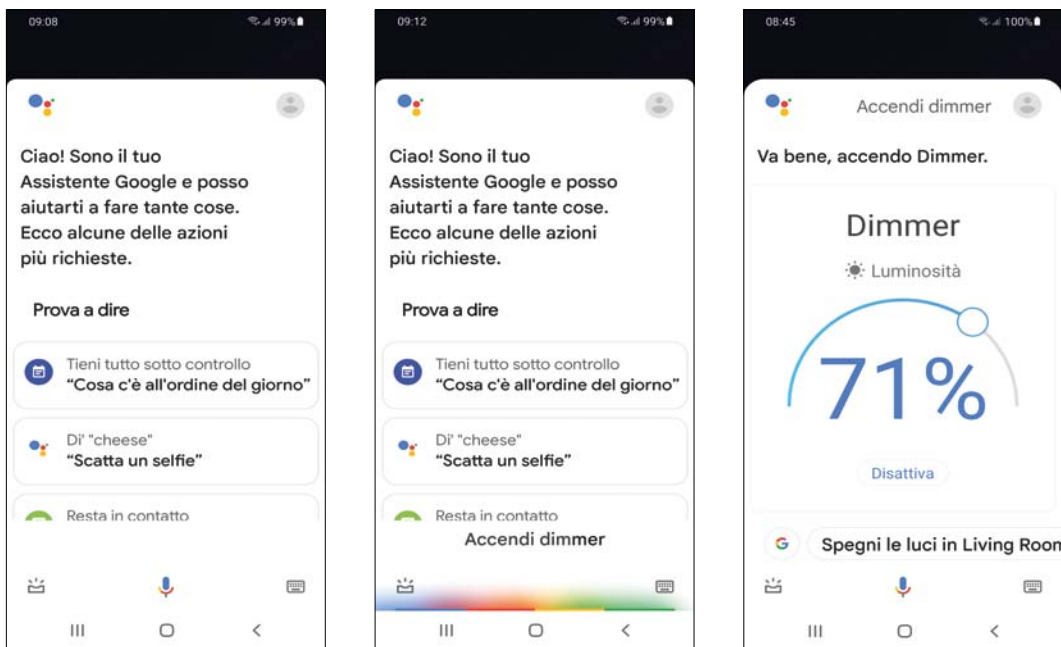


Fig. 16
Interfaccia
Google
assistant.

del coperchietto, che viene montato a incastro, in modo che il forellino laterale sia posizionato sopra al pulsante SW1 per permettere di schiacciare in caso vogliate reimpostare la password. Tenendo premuto il pulsante per più di cinque secondi, come conferma la lampadina collegato all'uscita lampeggerà tre volte. Il modulo si riavvierà con le impostazioni predefinite, impostando la password di sistema a "123456789" e cancellando l'SSID della rete memorizzata nonché le credenziali IoT.

A questo proposito vi informiamo che la libreria `lotwebconf` di default visualizza il log sulla seriale in chiaro, compresi i parametri del vostro account `lot`. Per motivi di sicurezza vi consigliamo di disabilitare tale opzione inserendo all'inizio del file `lotWebConfSettings.h`, che si trova nella cartella `SRC`, la riga `"#define IOTWEBCONF_DEBUG_DISABLED"`. Per quelli che vogliono realizzare da soli il contenitore con la stampante 3D, consigliamo di impostare l'altezza del layer a un valore non superiore ai

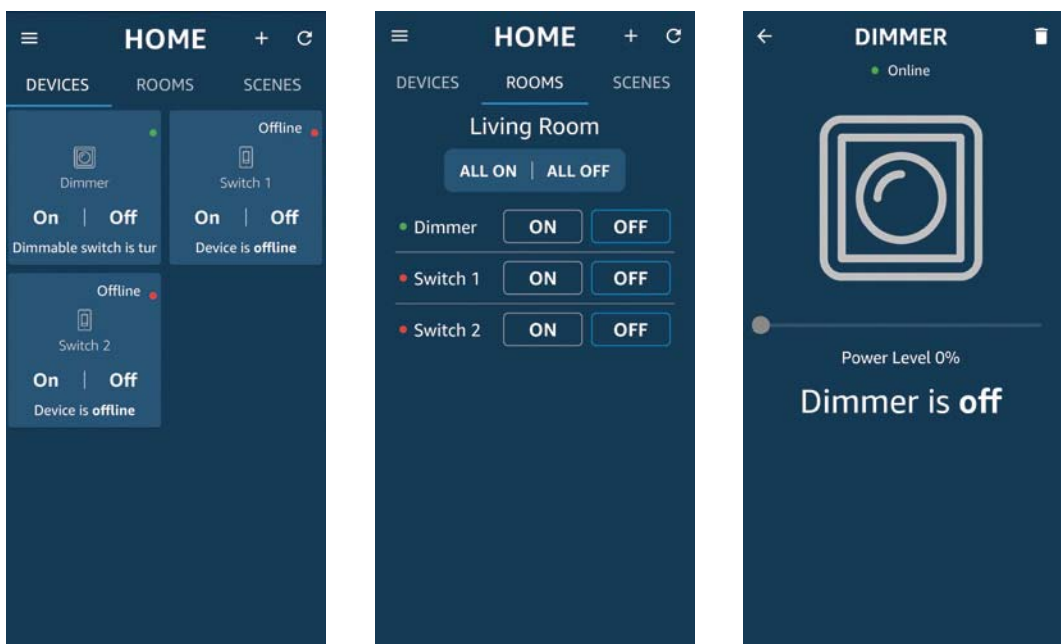
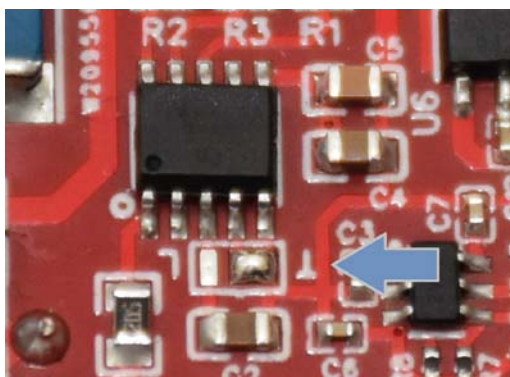


Fig. 17
Interfaccia
dell'app
Sinric Pro.

Fig. 18
Ponticello leading/
trailing edge.



0,2 mm e, a seconda della versione del software di slicing, attivare l'utilizzo del riempimento a linee nel riempimento a griglia ad alta densità, altrimenti la parete del contenitore, spessa appena 1,5 mm, non verrà riempita correttamente presentando uno spazio vuoto. Benché con il filamento PLA si ottengono ottimi risultati, consigliamo comunque l'utilizzo di filamento in ABS che sopporta temperature di fusione più alte, oppure quello ABS VO autoestinguente certificato dallo standard UL94 VO. In alternativa, per chi ha difficoltà a realizzare i contenitori, oppure non possiede una stampante 3D, è disponibile il servizio di stampa 3D su <https://www.futurashop.it/index.php?route=stampa3d/stampa3d>.

UTILIZZO DELLO SMART DIMMER

Per accedere alla pagina di configurazione dello Smart dimmer 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 nella **Fig. 8**: in "Auto off timer" potete impostare il valore in minuti del temporizzatore dell'autospegnimento della luce. Questa funzione è utile soprattutto per i più sbadati che dimenticano di spegnere la luce in garage o in cantina, rimanendo accesa anche per più giorni o settema-

ne. Ad esempio impostando il timer a 240 minuti (4 ore) la luce si spegnerà da sola; impostando il timer a 5 minuti si potrebbe usare lo Smart dimmer per controllare l'accensione delle luci in un corridoio oppure sulle scale, rimpiazzando il vecchio temporizzatore elettromeccanico. Nella casella "Min brightness" potete impostare il livello minimo di luminosità della lampadina. A differenza delle lampadine a incandescenza o alogene, quelle CFL e quelle LED hanno una soglia di accensione sotto la quale non riescono ad accendersi oppure hanno difficoltà a mantenere stabile la luminosità. Questa soglia varia da lampadina a lampadina e dal numero di lampadine collegate al dimmer, quindi questo valore dovrà essere impostato empiricamente fino a raggiungere per tentativi un funzionamento soddisfacente. Noi consigliamo di impostare la luminosità al minimo, se la lampadina non si accende aumentate il valore con passi del 3 - 5 %. Quando la luce si accenderà istantaneamente senza indecisioni avete raggiunto la soglia di regolazione minima per quella determinata lampadina. Nella casella "Fade delay" potete impostare l'enfasi dell'effetto dissolvenza cioè prolungare il tempo di accensione o spegnimento graduale. Impostando il valore a 10 si ha il massimo effetto, mentre il valore zero disabilita l'effetto dissolvenza. La casella di spunta "Enable external pushbutton" vi permette di abilitare o disabilitare la regolazione mediante il pulsante collegato allo Smart dimmer. Disabilitare tale funzione potrebbe essere utile durante un evento importante, per evitare che qualche maldestro ospite per sbaglio appoggiandosi sul pulsante spenga la luce, oppure evitare che venga accesa se nella sala viene riprodotto un film. Viceversa l'attivazione della regolazione mediante il pulsante è utile soprattutto nelle installazioni senza connessione internet oppure, quando la connessione viene a mancare, permettendo la regolazione e lo spegnere della luce che altrimenti rimarrebbe accesa.

Fig. 19
Contenitori,
adattatori e frutti
stampati in 3D
per placche Vimar
Plana, Vimar Idea,
Gewiss System,
Gewiss Playbus
e bTicino.



La casella di spunta "Enable assistant control" vi permette di abilitare o disabilitare il controllo mediante assistente vocale. La casella di spunta "Enable web control" vi permette di abilitare il controllo della regolazione della luminosità da pagina web. Poiché questa pagina è accessibile da chiunque anche senza credenziali, se il dispositivo è installato in una rete pubblica consigliamo di disabilitare questa funzione. Vi chiederete perché non abbiamo protetto la pagina con password. Generalmente lo Smart dimmer verrà installato in un ambiente domestico, collegato ad una rete WiFi privata e non pubblica, probabilmente risulterebbe scomodo e frustrante immettere ogni volta le credenziali di accesso della pagina web, per comandare la luce della stanza con lo smartphone. Abbiamo preferito lasciar decidere all'utente di abilitare questa funzione a seconda delle necessità e del tipo di installazione. Nella casella di spunta "Turn on at startup", potete scegliere se la luce deve rimanere accesa o spenta all'accensione o al riavvio del dispositivo. Se volete illuminare il cortile di casa e desiderate che la luce rimanga accesa di notte anche dopo un black out abilitate questa funzione, se invece volete che la luce rimanga spenta disabilitatela. L'ultima casella di spunta che ci rimane da analizzare è "Reboot device on save", che permette di abilitare il riavvio del dispositivo al salvataggio della configurazione con il tasto "Apply".

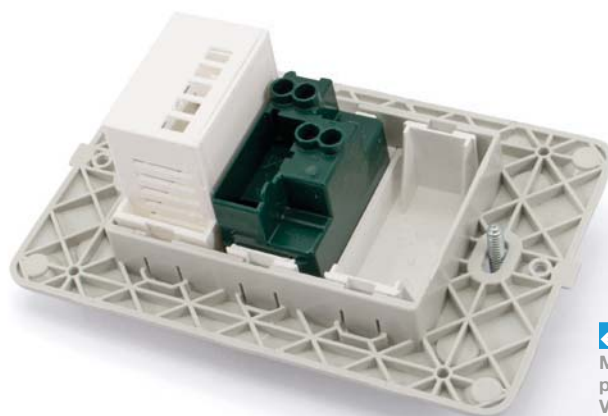
Analizziamo ora le funzionalità di controllo del pulsante collegato al connettore CON1. A luce spenta, una breve pressione del pulsante fa accendere la luce ripristinando la luminosità impostata prima dello spegnimento, mentre a luce accesa, una pressione breve del pulsante la fa spegnere. A luce spenta, con una pressione più lunga, compresa tra 250 e 900 ms, accende la luce alla massima luminosità. A luce spenta, tenendo premuto il pulsante la luminosità varierà dal minimo al massimo. Raggiunto il massimo invertirà la direzione, continuando finché il pulsante è premuto. Impostata la luminosità, se il pulsante verrà tenuto premuto nuovamente, la direzione di regolazione si invertirà. All'accensione della luce viene sempre ripristinata l'intensità luminosa precedentemente impostata (effetto memoria), tranne dopo un blackout dove la luce si accenderà alla massima potenza.

Vediamo assieme alcuni comandi riconosciuti dall'assistente vocale. Supponiamo che in *sinric.pro* al nostro dispositivo abbiamo assegnato il nome **Dimmer** (Fig. 10). Per accendere la luce pronunciate: "Alexa, accendi Dimmer" oppure "Hey Google, accendi Dimmer".

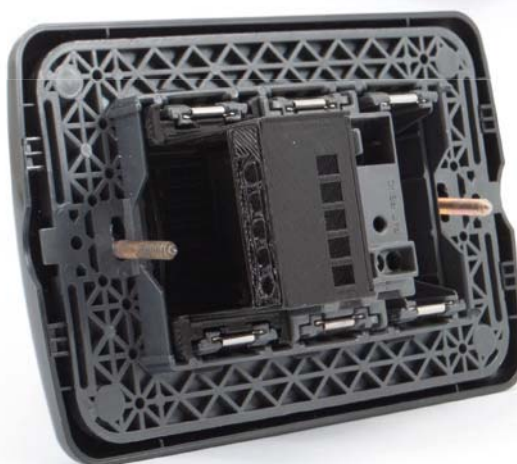


Fig. 20
Le frecce indicano i segmenti di plastica da eliminare sul copriferro.

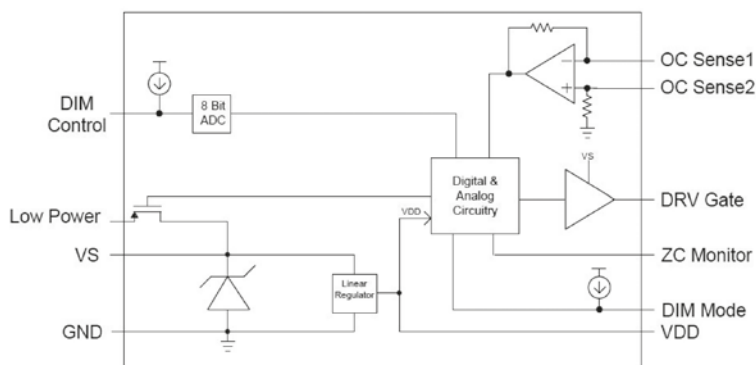
Per spegnere la luce pronunciate: "Alexa, spegni Dimmer" oppure "Hey Google, spegni Dimmer".
Per regolare la luminosità al 30% pronunciate: "Alexa, intensità Dimmer 30%" oppure "Hey Google, intensità Dimmer 30%".
Per aumentare la luminosità pronunciate: "Alexa, incrementa luminosità Dimmer" oppure "Hey Goo-



←
Montaggio nel portafrutto VIMAR Plana



←
Montaggio nel portafrutto VIMAR Idea.



Schema a blocchi interno e piedinatura dell'integrato utilizzato nel nostro dimmer per regolare la luminosità delle lampade; il chip contiene un completo regolatore a parzializzazione d'onda impostabile come trailing-edge o leading-edge mediante l'apposito piedino di controllo (4) secondo quanto spiegato nella tabella in basso in questo riquadro.

N° Pin	Nome	Descrizione
1	ZC monitor	Tensione d'ingresso per rilevamento dello zero crossing.
2	DIM control	Ingresso dell'ADC a 8 bit con tensione di riferimento a 2,5 V. Il segnale viene utilizzato per controllare l'ampiezza del segnale dim.
3	VDD	Uscita del regolatore interno a 5V.
4	DIM mode	Modalità di controllo: GND imposta trailing edge e VDD seleziona leading edge. All'avvio la modalità DIM è bloccata per 60 ms fino a quando non si verifichi un segnale POR (Power-On-Reset).
5	GND	Polo negativo di alimentazione.
6	VS	Ingresso di alimentazione. Un regolatore interno limita l'alimentazione a 17V.
7	Low power	Questo pin fa capo a un transistor PMOS. Quando si imposta a zero volt il piedino 4 DIM questo pin viene cortocircuitato al pin 6 VS.
8	OC Sense 2	Collegato al Drain di Q2 regola la massima differenza di voltaggio tra Q1 e Q2 per entrambe le semionde positiva e negativa.
9	DRV gate	Uscita per il pilotaggio di transistor tipo IGBT o MOSFET.
10	OC sense 1	Collegato al Drain di Q1 regola la massima differenza di voltaggio tra Q1 e Q2 per entrambe le semionde positiva e negativa.

gle, incrementa luminosità *Dimmer*".

Per diminuire la luminosità pronunciate: "Alexa, decrementa luminosità *Dimmer*" oppure "Hey Google, decrementa luminosità *Dimmer*". Per saperne di più, nell'app Amazon Alexa selezionate l'icona in alto a sinistra, scegliete la voce "*Cose da provare*" e cliccate l'opzione "*Casa intelligente*" per conoscere tutti i comandi vocali riconosciuti da Amazon Alexa, e se in casa avete installato un dispositivo Amazon echo dot, comandare lo Smart dimmer sarà ancora più comodo e facile.

Ricordiamo che il Dimmer funziona solo con lampadine CFL o LED di tipo dimmerabile; poiché la maggior parte delle lampadine in commercio non sono dimmerabili, fate attenzione a quale scegliete.

CONCLUSIONI

Con questo progetto speriamo di aver soddisfatto coloro che cercano un regolatore di luminosità affi-

dabile, versatile, gestibile e comandabile da remoto con lo smartphone o con l'assistente vocale, capace di regolare carichi resistivi, induttivi, lampadine a incandescenza, CFL e a LED nonché essere di dimensioni contenute per facilitare l'installazione negli impianti domestici preesistenti. □



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 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>

DOMOTICA CON SCADA ED ESP8266



di **ANTONIO ALOISIO**

Realizziamo
un dispositivo
IoT capace di
comandare carichi a
220V 10A, usando
come interfaccia
uno SCADA
Open Source.

I modi per implementare dei controlli domotici sono svariati ed ognuno si basa sull'utilizzo di tecnologie e di strumenti differenziati; in queste pagine vorremmo descriverne uno davvero originale, giacché vi proponiamo un progetto nato da un approccio non convenzionale, utilizzando un software SCADA (*Supervisory Control And Data Acquisition*) nato per l'automazione industriale, per implementare il comando e controllo via WiFi di un dispositivo IoT, basato sul collaudato ESP8266 e integrato in una scheda con uscita di potenza a TRIAC. Ci rendiamo conto che con l'avvento degli smart speaker corredati e supportati da vari assistenti (Google Home, Alexa, ecc.), utilizzare un sistema industriale può sembrare fuori luogo, ma vi renderete conto, nel corso della descrizione progetto, dell'ampia personalizzazione che permette uno SCADA e della conseguente possibilità di svincolarsi da cloud e connessione Internet. Di contro, utilizzare una soluzione SCADA richiede delle conoscenze

informatiche leggermente più alte, ad esempio per creare degli script che gestiscano processi automatici lato server. Questa è la profonda differenza tra la domotica e il processo industriale: in quest'ultimo l'automazione è demandata ai PLC (Controllori Logici Programmabili dispositivi molto solidi e stabili) a bordo macchina e la supervisione ai PC (lato server), mentre nella domotica si può integrare automazione e supervisione in un semplice computer (tutto lato server). Il motivo è facilmente spiegabile: non si può permettere ad un blocco di un PC o del suo sistema operativo di fermare un processo industriale.

La **Fig. 1** propone un esempio di interfaccia realizzata con uno SCADA.

Lo sviluppo del progetto qui proposto avverrà toccando tre argomenti principali:

- il software SCADA Open Source Rapid Scada;
- il protocollo di comunicazione ModBus TCP;
- il dispositivo IoT basato su ESP8266.

SCADA E RAPID SCADA

I sistemi SCADA (acronimo di "*Supervisory Control And Data Acquisition*", cioè "controllo di supervisione e acquisizione dati") sono software, generalmente installati su PC o server, che permettono di interfacciare un sistema di automazione anche molto complesso con gli operatori umani, facilitandoli nella condotta mediante l'utilizzo anche di pagine grafiche riepilogative di semplice utilizzo e con ampia visione di insieme.

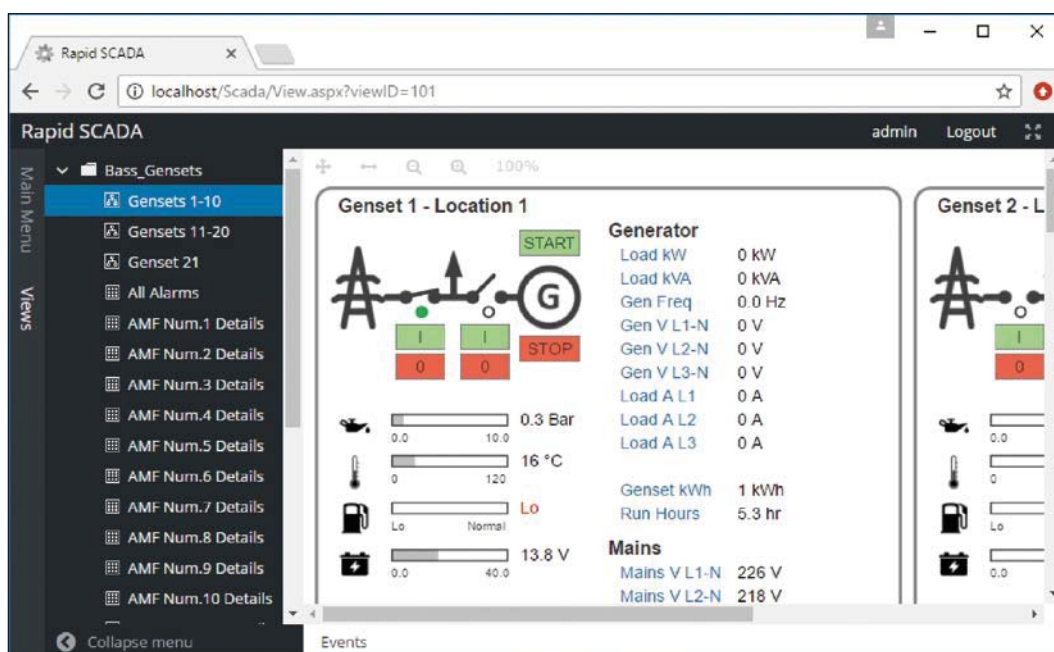
Il software legge dati o comanda azioni, verso qualsiasi dispositivo "parli" il suo stesso "linguaggio", cioè deve avere la capacità di scambiare dati mediante reti informatiche utilizzando protocolli standard. Tali dati vengono immagazzinati in database interni allo SCADA stesso, per poter essere manipolati e gestiti.

Nell'automazione industriale esistono svariati protocolli di comunicazione, tra i più usati dei quali c'è il ModBus TCP. **Rapid Scada** (abbreviato da ora con RS) è il software SCADA che abbiamo utilizzato.

Nel progetto è usato nella sua versione base free, ma è possibile aggiungere plug-in con funzionalità aggiuntive a pagamento. È un programma open source, che nella sua versione gratuita disponibile per gli usi non commerciali, è l'ideale per la realizzazione di una interfaccia grafica semplice ed intuitiva come quella del nostro progetto. Tali interfacce grafiche, in automazione industriale, vengono chiamate "mimici". Obiettivo principale di un mimico deve essere quello di fornire all'operatore un adeguato numero di comandi, feedback e misure nel più semplice ed ergonomico modo possibile. Saper configurare anche a livello grafico uno SCADA vi permette di creare per esempio un mimico della vostra abitazione, con numerosi dispositivi IoT per comandare e supervisionare ciò che volete. In vari articoli della rivista abbiamo presentato dispositivi IoT, facilmente adattabili al protocollo ModBus con piccole modifiche del loro codice.

La maggior parte degli inverter per impianti

Fig. 1
Esempio di
interfaccia
realizzata con uno
SCADA.



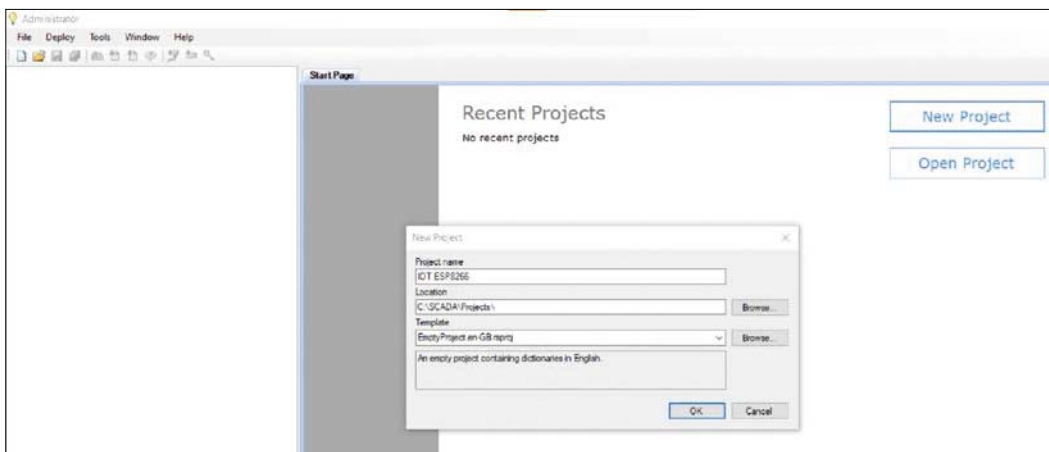


Fig. 2
Impostazioni
di base per
un nuovo
progetto.

fotovoltaici, di pompe di calore e di numerosi altri dispositivi, ha la possibilità di comunicare con il ModBus TCP, essendo questo uno dei protocolli più usati al mondo (o con il più semplice ModBus RTU adattabile al TCP con economici convertitori di protocollo), il che permette di evolvere il progetto presentato in modo enorme.

Non andremo oltre nell'argomentare gli SCADA in generale, essendo un mondo ampio e complesso che svierebbe dagli obiettivi di questo articolo. In commercio ne esistono parecchi, spesso con licenze proprietarie dei marchi più importanti dell'automazione industriale e ovviamente con numerose funzionalità aggiuntive rispetto ad Rapid Scala. RS è un software moderno e viene installato su PC o server, ma la sua interfaccia è raggiungibile mediante browser da qualsiasi dispositivo collegato alla rete a cui appartiene. Ciò permette di avere svariate postazioni operatore, anche su dispositivi mobili, senza dover installare niente. Quindi nessuna app e nessuna incompatibilità di sistemi operativi, solo il semplice browser. Basta immettere l'indirizzo IP del server nella forma **192.168.X.X/scada** per accedere all'interfaccia operatore. Se si usa direttamente il computer dove è installato RS bisogna inserire IP del local host, cioè **127.0.0.1/scada** o direttamente **localhost/scada**. Chi ha dimestichezza con i server web, avrà intuito che RS utilizza le potenzialità di IIS (Internet Information Services) cioè il server web di Microsoft per sistemi Windows, che deve quindi essere presente sul PC (o server) ove avverrà l'installazione.

INSTALLAZIONE DI RAPID SCADA SU PC

Per sviluppare questo progetto abbiamo scelto come "server" un mini-PC compatto con Sistema Operativo Windows 10; si può anche optare per una scheda Raspberry Pi, che rappresenta una

soluzione comoda e più economica.

Dopo aver scaricato l'ultima versione disponibile dal sito <https://rapidscada.org/download-all-files/download-rapid-scada> e la relativa guida all'utilizzo <https://rapidscada.net/doc/content/latest/en/> prestare molto attenzione alla configurazione dell'IIS e di Microsoft .NET Framework. Occorre seguire scrupolosamente le configurazioni iniziali (*Before Installation* nella guida) altrimenti il successivo avvio dell'installazione mediante il file *ScadaSetup.exe* non avverrà correttamente.

Se l'installazione è andata a buon fine, sul PC sarà presente il programma di configurazione di RS (*ScadaAdmin.exe* una lampadina come icona) ed nella directory principale la cartella **c:\SCADA** con una serie di sottocartelle ove vi sono tutte le funzionalità del programma.

CREAZIONE DELL'INTERFACCIA

Per creare l'interfaccia aprite *ScadaAdmin.exe* e creare un "nuovo progetto" dandogli un nome arbitrario (**Fig. 2**); il nostro è **IOT ESP8266**.

Il programma si presenterà con un layout semplice, che vede un menu in alto nella schermata di lavoro e una finestra dinamica a destra, nella quale si possono aprire contemporaneamente varie pagine delle impostazioni scelte dall'albero del progetto, che rimane sempre fisso a sinistra (**Fig. 3**).

Le fasi della creazione dell'interfaccia si basa su tre elementi base di partenza:

- le linee di comunicazione (Communication line);
- i dispositivi (Device);
- i canali (Channels).

LINEA DI COMUNICAZIONE MODBUS

Le linee di comunicazione nel MODBUS hanno lo scopo di instaurare la comunicazione con il dispositivo da supervisionare o comandare.



Modbus è un protocollo di comunicazione open source definito da una struttura gerarchica di tipo master - slave. L'accesso fisico si basa su una trasmissione di tipo seriale Half-duplex. Fu sviluppato nel 1979 dalla Schneider Automation Inc. per la creazione di reti industriali, allo scopo di consentire l'interscambio di informazioni tra dispositivi di controllo in ambito industriale.

A oggi il protocollo è libero da diritti di autore dell'inventore, il che permette di conoscerne a pieno tutte le sue caratteristiche, come la struttura dei pacchetti dati, il che ad esempio ha dato la possibilità ad André Sarmento Barbosa di creare la libreria che utilizzeremo nel progetto (dettagli su <http://github.com/andresarmiento/modbus-arduino> dove vi sono una serie di link utili per approfondire il funzionamento del protocollo). Come vedremo nel progetto, nel paradigma di questa comunicazione, vi deve sempre essere un dispositivo che interroga (Master), e uno o numerosi dispositivi (Slave) che rispondono ciclicamente (polling) alle richieste. L'interrogazione avviene verso quattro tipi di registri (tipi di memorie dei dispositivi slave) di seguito riepilogati:

- uscite discrete (Coil register): registro delle uscite digitali del dispositivo slave;
- ingressi discreti (Discrete input): registro degli ingressi digitali del dispositivo slave;
- uscite numeriche (Holding register): registro delle uscite analogiche del dispositivo slave;
- ingresso numerico (Input register): registro degli ingressi analogici del dispositivo slave.

Nel progetto useremo un Coil register per comandare l'uscita digitale dell'ESP01, ed un Holding register (e non un Discrete Input) per il feedback perché è un elaborato interno del software, non lo stato di un ingresso digitale.

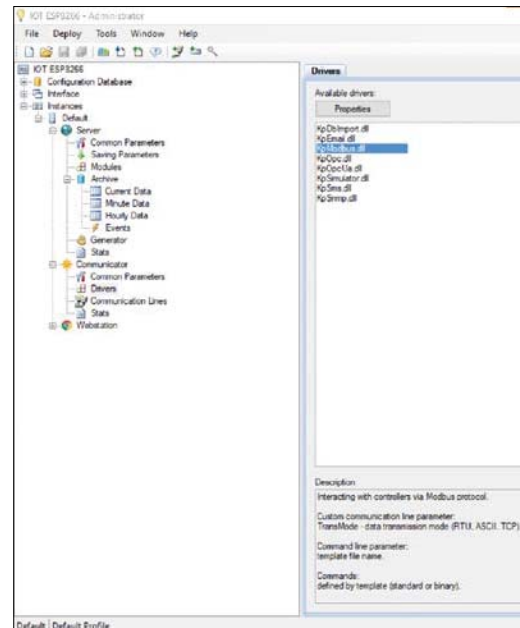


Fig. 4
Finestra delle librerie, nel programma chiamati Driver.

protocollo ModBus TCP, e la libreria utilizzata sarà KpModbus.dll (in ambiente Rapid Scada si chiama Driver). Cliccando su menu "tools → add line" verrà creata una nuova linea di comunicazione da impostare come in **Fig. 5**.

Nome e Descrizione potete sceglierli a piacimento. Nell'albero del progetto alla voce "Instances → Default → Communicator → Communicator line".

Per funzionare, Rapid Scada, si sfruttano delle librerie dinamiche (dll) appositamente create per far comunicare il dispositivo ed RS con un determinato protocollo. Come anticipato, noi abbiamo scelto il

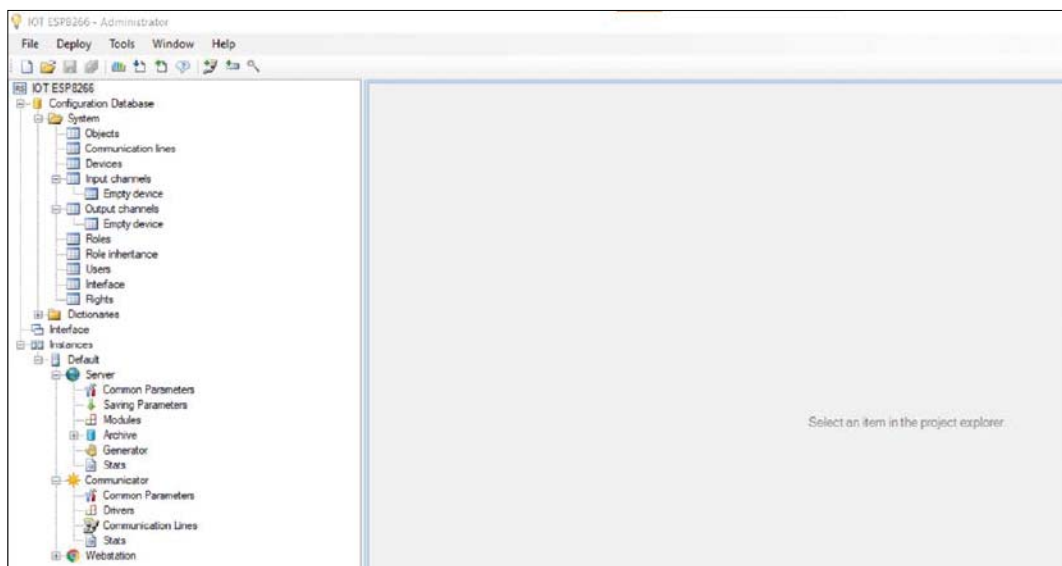


Fig. 3
Layout del software Rapid Scada.

comparirà la linea di comunicazione appena creata. Le impostazioni di tale linea di comunicazione vanno inserite nella pagina di configurazione raggiungibile dallo stesso percorso precedente, alla voce "Line Parameters" come esemplificato nella **Fig. 6**. Tra i dati da inserire c'è l'indirizzo IP del dispositivo IoT fisico, che renderemo statico mediante una istruzione dedicata del codice che vedremo più avanti, e ovviamente la porta standard di comunicazione del protocollo ModBus cioè la 502. Consigliamo di salvare periodicamente il progetto dalla classica icona a dischetto.

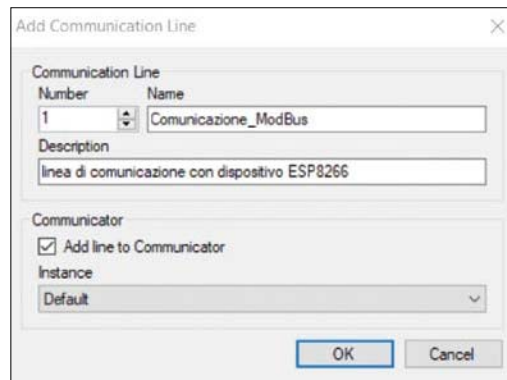


Fig. 5
Impostazioni di base di una nuova linea di comunicazione.

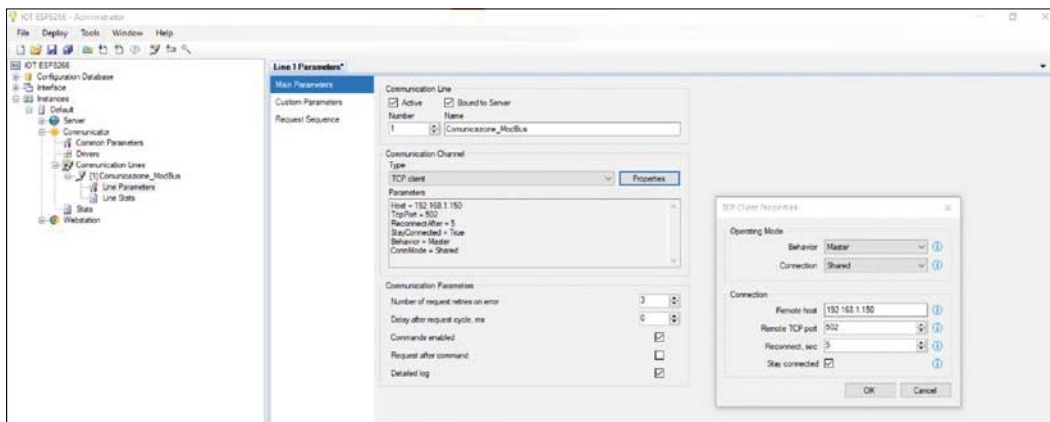


Fig. 6
Impostazioni di dettaglio della linea di comunicazione creata.

DISPOSITIVO

In una linea di comunicazione MODBUS si possono avere numerosi dispositivi associati, ovviamente tutti devono avere la possibilità di comunicare con il protocollo scelto per tale linea. Un progetto può avere numerose linee di comunicazione impostate con differenti protocolli di comunicazione. In Rapid Scada il dispositivo (device) è l'insieme dei segnali di ingresso e uscita del dispositivo fisico con il quale si vuole comunicare. Nel progetto inviamo due comandi (uscite), accensione e spegnimento, poi leggiamo (ingressi) il feedback dell'avvenuto invio del comando. L'insieme di tali segnali sono i canali. Cliccando su "tools → add device" verrà creato un nuovo dispositivo da impostare come in **Fig. 7**. Notare come viene inserita alla voce "Communicator line" la linea creata in precedenza. Reinserire l'indirizzo IP. Confermare con OK. Ora nell'albero del progetto "Instances → Default → Communicator → Communicator line" comparirà il nome del dispositivo creato.

CANALI

Cliccando sul dispositivo precedentemente creato si aprirà a destra una finestra dove è riepilogato lo

stato del dispositivo.

Cliccare su "Properties" e nella finestra che si aprirà modificare il protocollo in "ModBus TCP", da qui cliccare su "Create new Template", si aprirà una ulteriore finestra da dove si settano i canali, cioè le vie di ingresso ed uscita che ci permettono di inviare o ricevere dati dal dispositivo IoT.

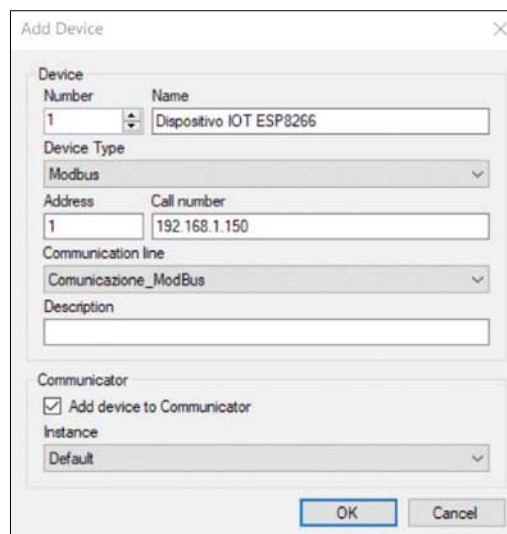


Fig. 7
Impostazioni di base di un nuovo dispositivo.

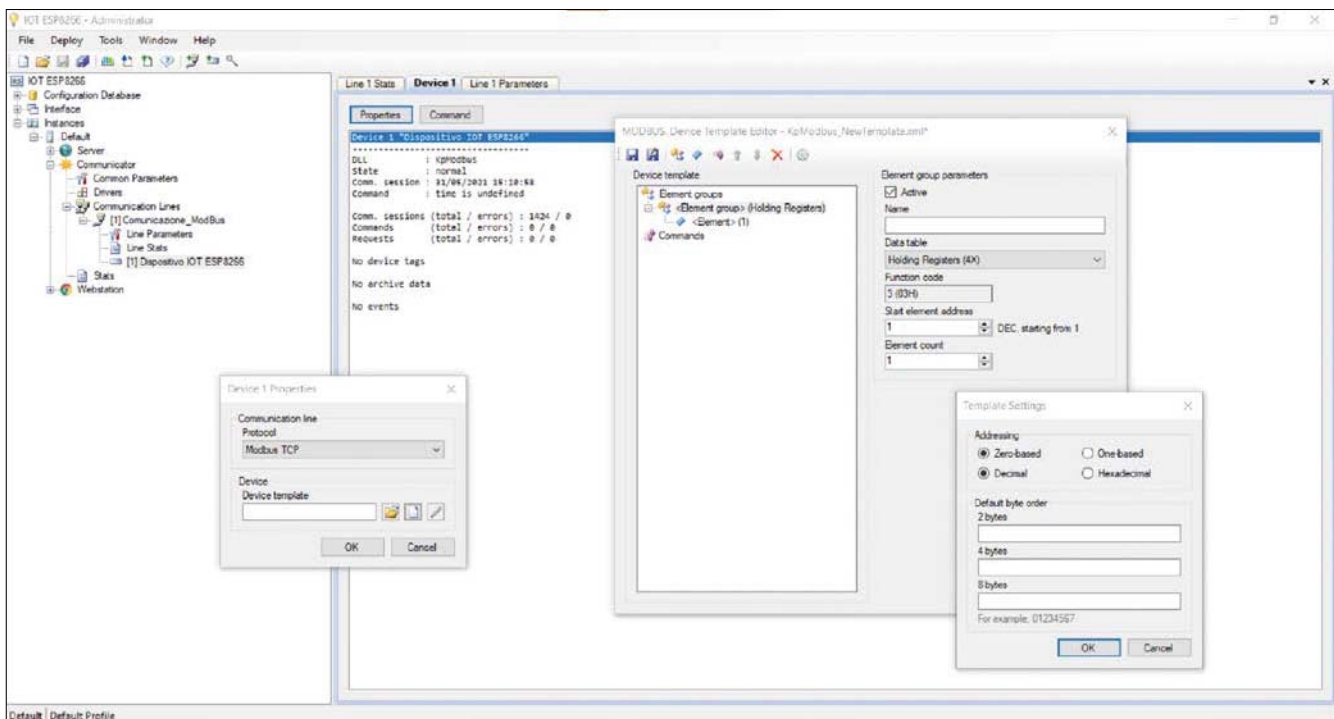


Fig. 8
Impostazioni
di base del
template dei
canali.

Fare attenzione che il "Template Setting" sia impostato su "Zero-based" ovvero che la numerazione dei canali inizi da zero. La configurazione fatta finora dovrebbe restituire un risultato simile a quello visibile nella **Fig. 8**.

Creare un nuovo gruppo di canali con il comando "Add element group" a cui dare un nome o lasciare quello predefinito. Aggiungere un nuovo canale "Add element" (icona azzurra) ed impostarlo come mostrato nella **Fig. 9**, ossia specificando che si sta creando un nuovo canale che andrà a leggere il registro 100 di tipo *Holding Register* del dispositivo IoT fisico. Questo è ovviamente un canale di ingresso, RS interroga il dispositivo, chiede i dati del registro

Fig. 9
Impostazione del
canale in ingresso.

100, e il dispositivo risponde.

Per creare un canale di uscita si utilizzano invece i "Commands". Cliccare sul comando "Add command" (icona fucsia) ed impostarlo come in **Fig. 10**. Salvare il Template dando un nome desiderato. Le fasi fondamentali sono ora terminate, occorre caricare il tutto sul server di RS, con il semplice comando "Deploy → Upload Configuration → Upload".

Se il dispositivo IoT fisico è alimentato e correttamente configurato per connettersi alla rete WI-FI usata, si può testare il tutto inviando un comando di ON – OFF e poi "Send" dalla finestra che si aprirà cliccando su "Command". Si avrà anche uno storico dei comandi inviati e lo stato del FeedBack come in **Fig. 11**.

PAGINA WEB DI INTERFACCIA

Il mimico sarà molto semplice, due tasti rispettivamente di ON e OFF ed una segnalazione circolare che cambia colore in base ai seguenti stati del feedback:

- ON → verde;
- OFF → nero;
- NON CONNESSO → grigio.

Il primo passo è abbinare ai canali creati nel dispositivo visto prima, i corrispettivi canali nelle

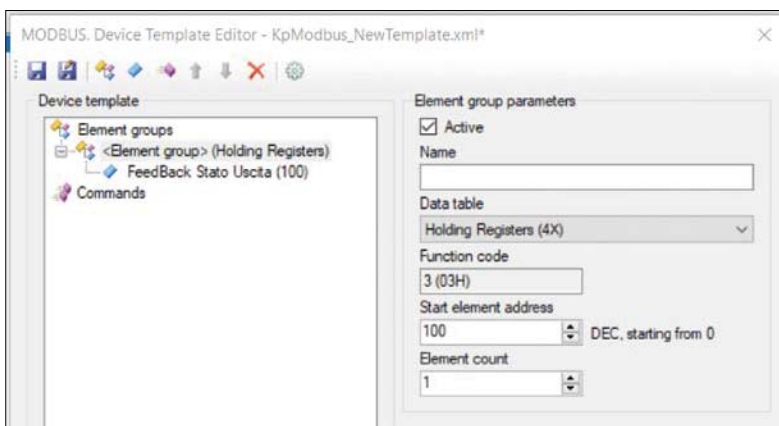


tabelle del database di RS suddivisi in input (canali di ingresso) e output (canali di uscita). Esiste una procedura guidata per far ciò (ed una manuale che vedremo più avanti), cliccando su "Tools → Create Channel" nel primo step viene chiesto di scegliere la linea di comunicazione e il dispositivo, vi sarà anche indicata la quantità dei canali di ingresso e uscita che verranno creati nel database. Questi dati dovrebbero essere già presenti nelle rispettive caselle, il che permette semplicemente di andare avanti (next).

Nel secondo step scegliere "Enterprise" dalla casella "Object", impostazione praticamente standard per la maggioranza dei progetti, quindi andiamo avanti.

Nel terzo e ultimo step viene indicata la numerazione che assumeranno i canali nel database, che si possono tranquillamente lasciare come preimpostati.

Si conclude questa fase con il comando "Create". Un messaggio dovrebbe confermare il successo dell'operazione. I canali creati saranno identificati con il numero 101, sia per l'ingresso che l'uscita. Questa numerazione è del tutto arbitraria, ma ci siamo allineati a quella creata dalla procedura guidata.

Dall'albero del progetto alla voce "Configuration Database → System → Input Channels" aprire la

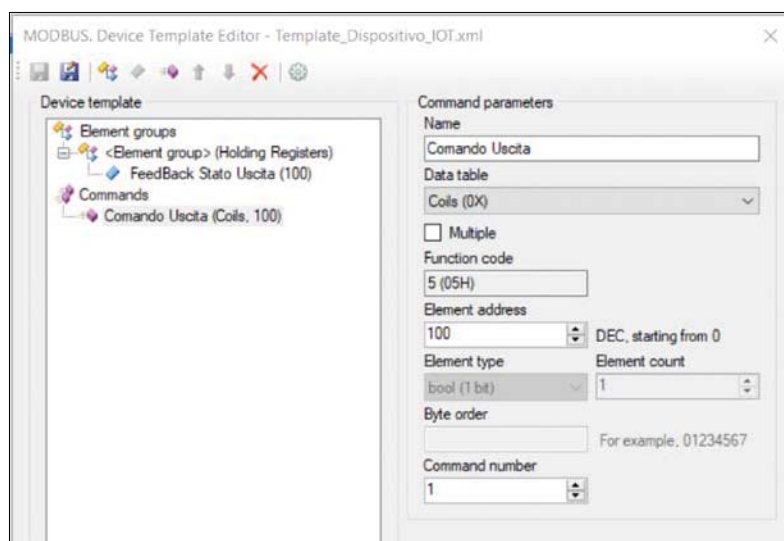


Fig. 10
Impostazione canale di uscita.

tabella dei canali di ingresso appena creata, con in questo caso un solo rigo, stessa cosa varrà per il canale di uscita "Configuration Database → System → Output Channels".

Queste due tabelle, permettono l'ingegnerizzazione dei canali, ovvero l'adattamento verso il mimico (quindi l'operatore umano) del dato grezzo fornito dal sistema di automazione (nel nostro caso dal dispositivo IoT).

Fig. 11
Stato del sistema e feedback.

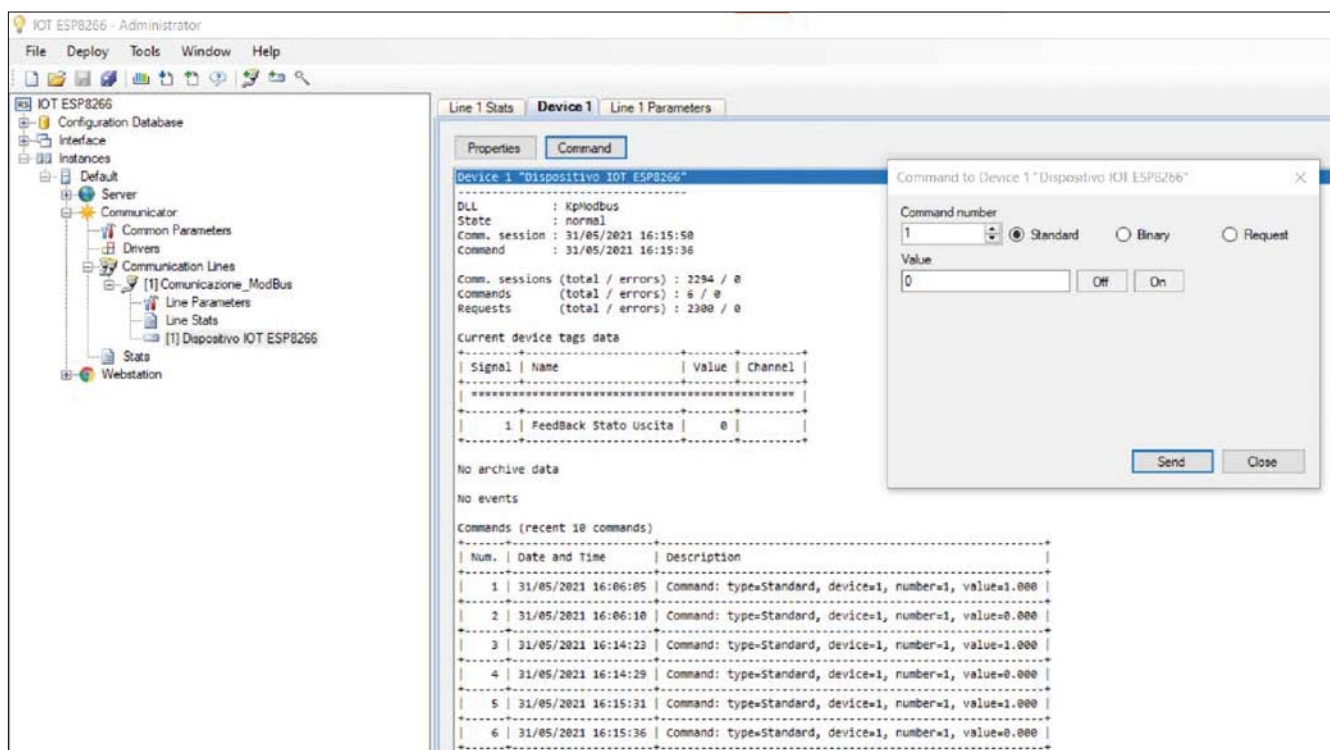


Fig. 12
Impostazione
database canali
di ingresso.

Per esempio, spesso i parametri fisici (temperatura, pressione, etc) vengono forniti da sistemi di automazione come un numero intero risultante dalla conversione analogica digitale, come avviene per gli ingressi analogici di Arduino dove il valore restituito da **analogRead()** è un numero intero compreso tra 0 e 1023. Spetta a noi nel codice, con una semplice formuletta, convertirlo ad esempio in gradi Celsius.

Nel nostro progetto ci siamo limitati ad impostare come D (numero intero) il Format del canale di ingresso. Da notare che il dato nella colonna "signal" corrisponde con il canale di comunicazione di ingresso (in questo caso, 1) che abbiamo creato per il dispositivo virtuale in precedenza. Esiste una alternativa all'inserimento dei dati direttamente nella tabella, ossia la maschera di immissione rapida, richiamabile agendo sul comando "Item Properties" come in **Fig. 12**.

Per l'uscita che andrà a comandare il triac, occorre inserire manualmente un altro canale del database, perché quello già creato invierà ON (1 → accensione del carico), il nuovo invierà OFF (0 → spegnimento del carico). Questi stati vanno inseriti nella colonna formula come semplici zero ed uno, atti-

Fig. 13
Impostazione
database canali di
uscita.

vando la spunta "Formula Used" come in **Fig. 13**. I numeri presenti nella colonna "Command Number" corrispondono con il canale di comunicazione di uscita (in questo caso 1) che abbiamo creato per il dispositivo in precedenza. Salvare il progetto. Le procedure viste finora sono propedeutiche ed indispensabili per disegnare ed impostare il mimico. Da menu ad albero, tasto destro su "Interface" → "New File". Dalla finestra che si aprirà si può scegliere il tipo di pagina di interfaccia che vogliamo creare (ci soffermeremo solo su "Scheme View") ed il file creato lo nomineremo "mimico.sch" (scelta ovviamente arbitraria).

Nel menu ad albero comparirà la voce "Interface → mimico.sch", cliccandoci due volte sopra si aprirà una pagina bianca del browser predefinito e una finestra di servizio con attivo il programma "Scheme Editor".

I pulsanti di ON e OFF si disegnano con la funzionalità "Button", e il cerchio di feedback con la funzionalità "Led" dalla scheda "Components" di "Scheme Editor" come da **Fig. 14**.

Dalla scheda "Properties" si possono modificare i font del testo, dimensioni, posizione e tante altre impostazioni degli oggetti grafici. Si possono aggiungere quindi tutte le proprietà degli oggetti che si sono o si andranno a disegnare. Consigliamo di prendere familiarità con il disegno e la modifica delle proprietà, in modo da potere personalizzare l'interfaccia nel miglior modo possibile.

Per una buona visualizzazione sui dispositivi mobile, cliccare sullo sfondo bianco ed impostare la sotto scheda "Size" su 200;400 (larghezza;altezza). Occorre ora abbinare ai tre oggetti grafici creati, i tre canali del database precedentemente configurati, ovvero:

- 101 canale di ingresso – Feedback del comando dato al dispositivo fisico;
- 101 canale di uscita – Comando di accensione del carico;
- 102 canale di uscita – Comando di spegnimento del carico.

Queste impostazioni vanno inserite nelle sotto schede "Data → Input Channel" per i canali di ingresso, "Data → Output Channel" per i canali di

Output channels - Device 1*											
Number	Active	Name	Command Type	Object	Device	Command Number	Command	Formula Used	Formula	Write Events	
101	☒	Dispositivo IOT ESP8266 - uscita - ON	Standard	Enterprise	Dispositivo IOT ESP8266	1		☒	1	☐	
102	☒	Dispositivo IOT ESP8266 - uscita - OFF	Standard	Enterprise	Dispositivo IOT ESP8266	1		☒	0	☐	

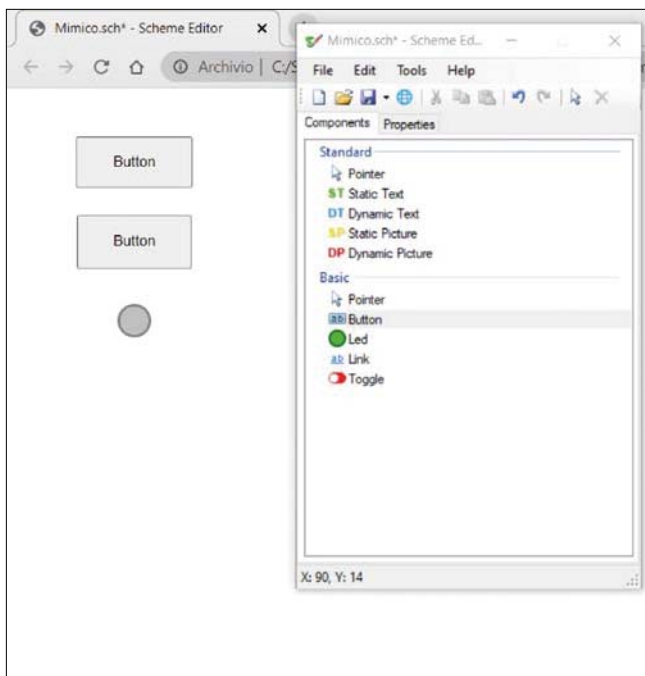


Fig. 14
Disegno dell'interfaccia utente.

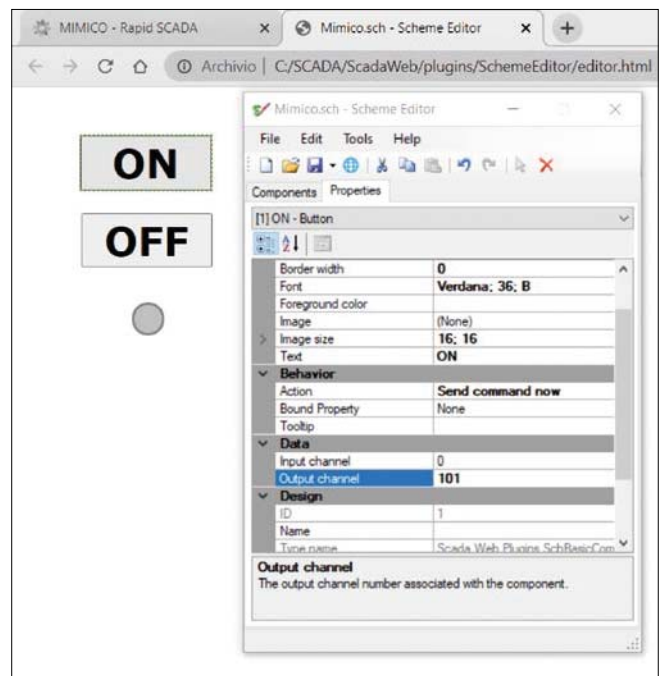


Fig. 15
Proprietà del pulsante ON.

uscita, come in **Fig. 15**.

Occorre inserire la logica delle variazioni di colori in base al feedback all'oggetto "led".

Aprire la sotto scheda "Behavior" → "Conditions", e cancellare con la x rossa gli "Items" preimpostati,

per poi configurarli nel seguente modo:

- Value = 0 → Color Black.
- Value = 1 → Color Lime
- Il grigio della mancata connessione è predefinito.

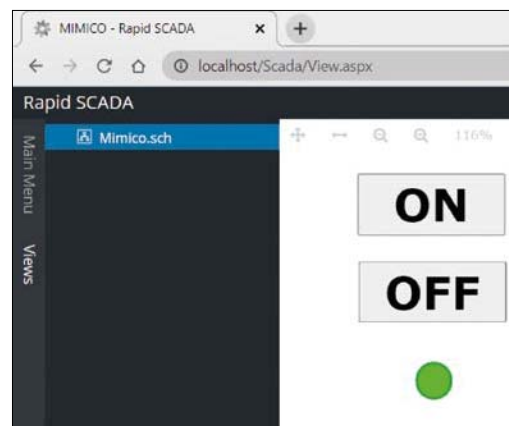
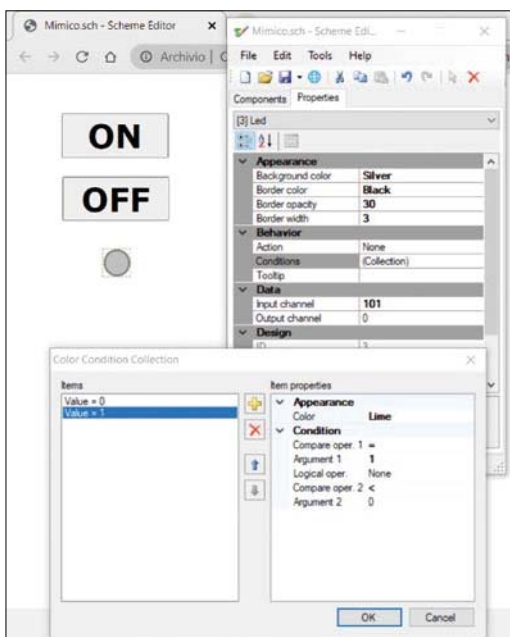
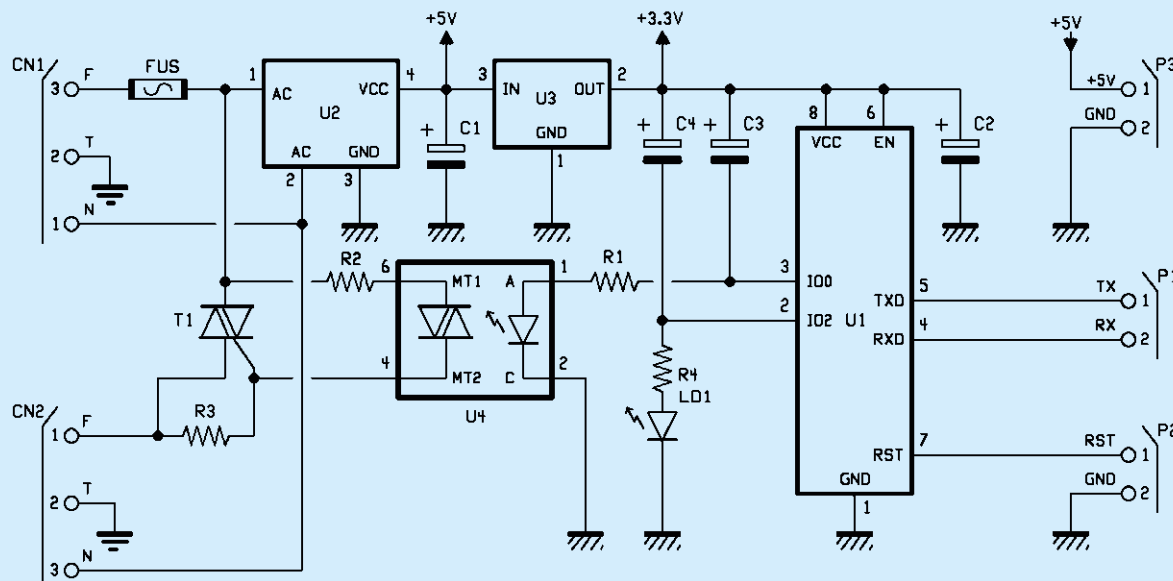


Fig. 16
Impostazioni logiche dei colori dell'oggetto "led".

Fig. 17
Interfaccia funzionante.



Salvare dal dischetto di "Scheme Editor" e chiudere la finestra, come anche la pagina di disegno del browser.

Essendo l'interfaccia una pagina web, occorre creare un menù per richiamare il mimico. Da "Configuration Database → System → Interface" configurare come in **Fig. 17**, dove la scritta "MIMICO" alla colonna "Title" sarà il nome del link del menù. Caricare sul server "Deploy → Upload Configuration → Upload".

Se il lavoro è andato a buon fine, e il dispositivo IOT è alimentato, aprire il browser all'indirizzo "<http://localhost/scada/>", comparirà l'interfaccia creata! Inviare i comandi per testarne la funzionalità. All'accesso verrà chiesto nome utente e password che di default sono : admin – 12345, modificabili alla pagina "Configuration Database → System → Users"; da dove si possono creare anche diversi ruoli e utenti.

Se si vuole utilizzare uno smartphone per il comando del dispositivo, aprire il browser all'indirizzo "192.168.1.1/scada/", anche in questo caso comparirà l'interfaccia creata.

Entrambi gli indirizzi IP del server e del dispositivo devono essere statici. Occorre a tal proposito verificare se il dispositivo di rete che si utilizzerà per la creazione della rete WiFi non abbia impostazioni specifiche per gli IP statici.

IL CIRCUITO DEL DISPOSITIVO IOT

Passiamo adesso a descrivere l'hardware del device IoT che abbiamo creato appositamente per il nostro progetto e che costituisce il sistema locale IoT con cui esercitare il controllo domotico.

Il cuore del dispositivo è il compatto e collaudato modulo WiFi ESP01, uno dei componenti più semplici ed economici della famiglia ESP8266; tale dispositivo è un completo transceiver WiFi che opera con una tensione di alimentazione pari a 3,3Vcc, che nel nostro circuito vengono ottenuti con un LD1117V33, siglato U3 e posto a valle di un modulo incapsulato in resina Hi-Link HLK-2M05, che nello schema elettrico è siglato U2 ed è preceduto dal canonico fusibile di protezione dal sovraccarico.

Dunque, il nostro apparato IoT si alimenta direttamente con la tensione di rete 230Vca e tramite un alimentatore switching stabilizzato con ingresso 100÷240Vca e uscita 5Vcc si ricava la bassa tensione con la quale viene alimentato poi il modulo WiFi ESP01 e ciò con cui esso interagisce.

Tutta la linea di alimentazione a 3,3V è filtrata in più punti mediante condensatori elettrolitici, utili a ridurre disturbi e ripple che la vicinanza della rete elettrica e la circuitazione interna al modulo alimentatore U2 possono determinare.

Si sarebbe potuto optare anche per un alimentatore esterno che fornisse direttamente 3,3Vcc, ma per la circuitazione adottata è stato meglio così, anche e soprattutto perché integrare l'alimentazione consente di contenere le dimensioni del device IoT e di renderlo di pratica installazione.

Nello schema vedete che del modulo ESP01 (siglato U1) abbiamo reso disponibili tre coppie di pin:

- P1 per collegare un altro dispositivo che comunichi sui pin seriali TX-RX;
- P2 per collegare un possibile pulsante di reset;
- P3 per alimentare un possibile dispositivo ag-

giuntivo a 5Vcc, con a disposizione circa 0,3 A (ad esempio un Arduino Uno assorbe 0,05A senza averci collegato alcunché).

Ciascuna coppia fa capo a una connessione dedicata a lato del circuito stampato, composta da piazzole a passo 2,54 mm cui si possono applicare dei pin-strip.

Le due linee di GPIO dell'ESP01 sono state configurate come uscite e risultano collegate così:

- IO0 ad un optoisolatore BRT21H (che nello schema elettrico vedete siglato U4) capace di pilotare il TRIAC di potenza con la funzionalità zero-crossing, ossia ad ogni impulso, prima di accendere il TRIAC collegato alla sua uscita (T1) attende che la sinusoide della tensione di rete abbia superato il passaggio per lo zero, il che riduce sia il sovraccarico che la produzione di disturbi elettromagnetici causati dalla chiusura ed apertura del circuito, soprattutto su carichi induttivi;
- IO2 è collegato ad un semplice LED che si illumina quando il dispositivo è connesso al WiFi.

Tutta l'elettronica del dispositivo IoT assorbe al massimo 90 mA di corrente nella condizione peggiore, cioè con il modulo WiFi connesso e il

fotoTRIAC attivo. Per alimentarlo con la stessa tensione di rete del carico, come già accennato, abbiamo scelto un alimentatore compatto 110-240Vac 0,2A → 5Vcc 0,4A della HI-Link, più che sufficiente per il nostro scopo.

Il fusibile posto a monte di tutto il circuito, cioè sull'ingresso di rete, va dimensionato in base al carico che intenderete collegare all'uscita a TRIAC del circuito; per esempio se vorrete pilotare un utilizzatore da 1.000 W vi converrà montare un fusibile da 5A. Ricordiamo che non bisogna superare i 10 ampere.

REALIZZAZIONE PRATICA

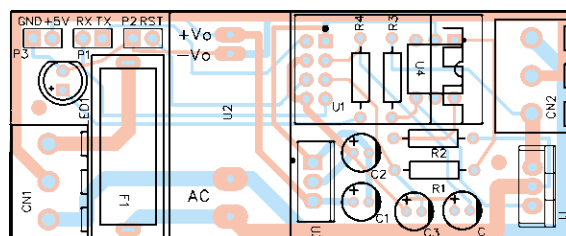
Bene, descritto il circuito del device IoT spendiamo qualche paragrafo per spiegare come realizzarlo in maniera sicura; per il dispositivo abbiamo disegnato un circuito stampato (lo vedete nelle foto del prototipo pubblicate in queste pagine) che potrete autocostituire seguendo (senza modificarle, dato che sono state progettate appositamente per il funzionamento a 220Vac...) le tracce lato rame scaricabili dal nostro sito www.elettronica.in.it insieme agli altri file del progetto, dalle quali ricaverete le pellicole utili a costruire la basetta ramata mediante fotoincisione. Inciso e forato il PCB, monterete i componenti in ordine di altezza, quindi prima

piano di MONTAGGIO

R1,R2,R4: 470 ohm
 R3: 390 ohm
 C1: 1 uF 25VL
 C2: 10uF 25VL
 C3, C4: 100uF 16VL
 F1: Fusibile 5A
 LED1: Led blu 3mm
 T1: Triac BTA16-600B
 P1, P2, P3: Microswitch (KRS6010)
 U1: ESP-01
 U2: Hi-Link HLK-2M05
 U3: Regolatore di tensione LD1117V33
 U4: Fotoaccoppiatore BRT 21H
 CN1, CN2: Morsetto 3 poli (YL044)

Varie

- porta fusibile da stampato
- strip femmina 4x2
- 3pz strip maschio 2 poli
- stampato 30x75mm S1567



Listato 1

```
#include <ESP8266WiFi.h>
#include <ModbusIP_ESP8266.h>
//Dichiarazione dei registri ModBus utilizzati
const int Registro_uscita = 100;    // è un Coil Register
const int Registro_fb = 100;        // è un Holding Register
//istanza dell'oggetto dalla libreria ModBus
ModbusIP mb;
// Variabile usata
int Stato_uscita;
// Pin usati
const int Pin_uscita_triac = 0; //
const int Pin_uscita_led = 2; //
void setup() {
    Serial.begin(115200);

    pinMode(Pin_uscita_triac, OUTPUT); digitalWrite (Pin_uscita_triac,LOW);
    pinMode(Pin_uscita_led, OUTPUT);   digitalWrite (Pin_uscita_led,LOW);
    delay(1000);
    IPAddress ip(192, 168, 1, 150);      // impostazione IP statico del dispositivo
    IPAddress gateway(192, 168, 1, 1);    // Ip del server dove vi è installato Rapid Scada
    IPAddress subnet(255, 255, 255, 0);
    WiFi.config(ip, gateway, subnet);
    WiFi.begin("SSID", "PASSWORD");    // DELLA VOSTRA RETE WIFI
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
        Serial.println("");
        Serial.println("WiFi connesso");
        Serial.println("Indirizzo IP : ");
        Serial.println(WiFi.localIP());
    }
    mb.server();                        //imposta il dispositivo come il server per il protocollo modbus
    mb.addCoil(Registro_uscita);         //imposto il registro di uscita 100 come Coil Register per comandare il triac
    mb.addHreg(Registro_fb);            //imposto il registro 100 come Holding Register per il feedback
}
void loop() {
    if (WiFi.status() == WL_CONNECTED) { digitalWrite(Pin_uscita_led, HIGH);} else { digitalWrite(Pin_uscita_led, LOW);} //
    comando led segnalazione wifi
    mb.task();                          // classe della libreria ModBus
    digitalWrite(Pin_uscita_triac, mb.Coil(Registro_uscita)); // comando uscita verso il triac
    Stato_uscita = mb.Coil(Registro_uscita); // leggo lo stato del registro che comanda l'uscita
    mb.Hreg(Registro_fb,Stato_uscita);   // memorizzo lo stato nel registro per leggerlo come feedback
    delay (10);
}
```

le resistenze ed il fotoTRIAC (quest'ultimo potete anche prevedere di dotarlo di uno zoccolo, per facilitarne eventualmente la sostituzione in caso di guasto) e i condensatori elettrolitici, proseguendo con le morsettiere, il portafusibile e il connettore 2x4 contatti femmina a passo 2x2,54 mm necessario a montare, una volta terminate le saldature, il modulo ESP01 (quest'ultimo andrà dotato, allo scopo, di pin-strip a passo 2,54 mm). Per completare il circuito si montano i due componenti in contenitore TO-220, ossia il regolatore di tensione a 3,3V e il TRIAC di potenza; entrambi vanno disposti con l'aletta metallica verso l'esterno del circuito stampato. Gli ultimi componenti da montare sono il modulo alimentatore AC/DC e il LED blu.

Per l'orientamento di tutti gli elementi polarizzati del circuito, bisogna riferirsi al piano di montaggio illustrato nella pagina precedente. Essendo il circuito alimentato con tensione di rete, occorre prestare la massima attenzione e prevedere tutte le precauzioni del caso per evitare il contatto diretto con le parti attive sotto tensione, perché potenzialmente pericolose, se non mortali. Per la dimensione delle piste, conviene non superare i 10A di carico, un valore più che sufficiente per la gestione di utilizzatori nell'ottica della domotica.

IL CODICE DEL DISPOSITIVO IOT

Diamo dunque uno sguardo al firmware che gestisce il nostro sistema, riferendoci al **Listato 1**. Il codice sfrutta due librerie di seguito descritte.



ESP8266WiFi.h: permette la connessione alla rete WiFi domestica e di assegnare un indirizzo IP statico al dispositivo IoT, in modo da avere un indirizzo univoco nella rete facilmente raggiungibile da RS. **ModbusIP_ESP8266.h:** permette di impostare il dispositivo IoT come server di una comunicazione ModBus, con l'obiettivo di rendere disponibili a RS i dati presenti nei suoi registri in seguito alle interrogazioni ricevute.

Il comando dell'uscita digitale dell'ESP01

"digitalWrite(Pin_uscita_triaco, mb.Coil(Registro_uscita))" è scritto direttamente utilizzando il valore del Coil Register 100 che diviene zero o uno si clicca su ON o OFF come abbiamo visto nella configurazione di Rapid Scada.

Il grande utilizzo del protocollo ModBus nell'automazione industriale è proprio nella sua natura Master-Slave. Il dispositivo Master (che in questo caso è Rapid Scada) interroga, e lo slave (vale a dire il nostro dispositivo IoT edge) risponde.

Se l'unità Master dovesse smettere di interrogare, la Slave continuerebbe comunque il proprio lavoro senza problemi, perché è il suo software a gestire il processo automatico.

Nel nostro progetto non abbiamo implementato logiche automatiche ma permane la funzionalità del nostro dispositivo di rimanere nel suo ultimo stato, essendo i suoi registri invariati anche se il Personal Computer dove è in funzione Rapid Scada dovesse spegnersi o perdere la comunicazione, o si dovesse perdere la rete WiFi. Ovviamente avremo perso la possibilità di comando e controllo, ma il dispositivo IoT mantiene l'ultimo stato d'uscita impostato.

Le logiche di automazione lato server accennate in precedenza si scrivono direttamente su Rapid Scada, in linguaggio C#, nella finestra "Configuration Database → Dictionaries → Formula (procedura che non verrà trattata in questo articolo).

CONCLUSIONI

Bene, possiamo ritenere terminata la trattazione del nostro sistema di gestione domotica.

Con questo progetto abbiamo cercato di creare un ponte tra il mondo dell'automazione industriale e quello della domotica; e ci siamo riusciti. Si tratta di un processo sicuramente più complesso rispetto alle procedure guidate di Blynk, Home Assistant, eccetera, ma apre alla creazione homemade di un completo ed autonomo sistema integrato di automazione.

Ci soffermiamo sul fatto come si tratta di un sistema autonomo, essendo il nostro progetto



La scheda del device IoT con il modulo ESP smontato.

indipendente da Internet e da vari sistemi cloud. Il che non significa che non possiamo interfacciarci ad esso via Internet da remoto; anzi, la cosa è fattibile, tanto che potrebbe essere oggetto di un futuro progetto in tema.

Continuate a seguirci e vedrete!



Cosa occorre?

Il modulo EPS01 (cod. ESP8266) è disponibile a Euro 5,00, il PCB può essere realizzato utilizzando il servizio di realizzazione PCB (cod. PCBPRODUCTION) disponibile sul sito [futurashop.it](http://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>

UN TEAM DI PROFESSIONISTI PER IL TUO PROGETTO



oltre **1500** progetti sviluppati
26 anni di attività
più di **15** progettisti



L'area R&D di Futura Group studia, progetta e realizza prototipi per differenti settori utilizzando tecnologie innovative. Nel team di Futura Group fa parte il personale interno e un numero elevato di collaboratori esterni, ciascuno specializzato nel proprio settore (Disegno CAD – Raspberry Pi – Arduino – Audio – Sviluppo firmware – Linux – ecc). La disponibilità di oltre 5000 prodotti elettronici e componenti consente di ridurre i tempi di test e sviluppo e di realizzare facilmente il primo prototipo. La decennale esperienza nel settore e la necessità di ottimizzare i tempi e i costi, ha portato Futura Group a selezionare una serie di terzisti per la produzione di piccola o grande serie.

I SERVIZI CHE TI OFFRIAMO



Progetto Completo

Partendo dalla tua idea progettiamo per te la soluzione.



Sviluppo PCB

Ottimizziamo il tuo prototipo disegnando un PCB compatto secondo le tue specifiche.



Scrittura Codice

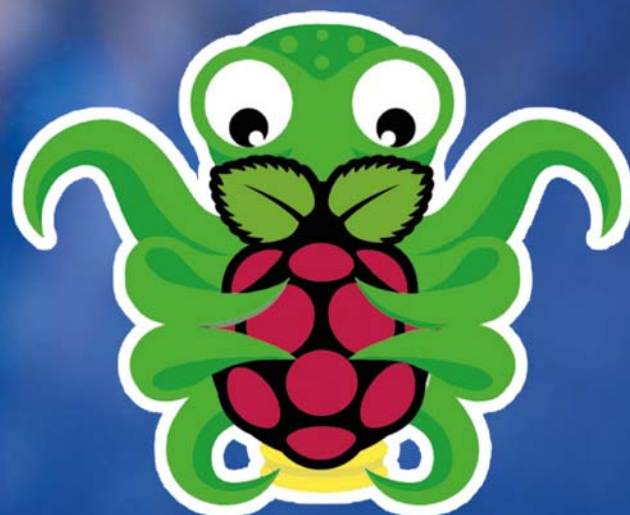
Scriviamo il codice del core per dare vita alla tua applicazione.



Produzione Progetto

Portiamo il prototipo alla fase di produzione seguendo per te tutte le fasi.

OCTOPRINT, LA STAMPA SMART



di GRETA GALLI

Permette di controllare le stampanti 3D anche da remoto rendendole a tutti gli effetti smart printer.

L'evoluzione tecnologica ha visto un incremento esponenziale nell'ultimo decennio e oramai siamo abituati ad avere quasi tutto smart, dalla macchina del caffè controllata da Alexa alla Tesla che si guida da sola, per arrivare infine ai dispositivi domotici nelle nostre case. Molti "oggetti" e dispositivi nascono smart, ma altri possono diventarlo nel tempo tramite apposite modifiche e l'implementazione di add-on; tra questi, anche le stampanti 3D come le nostre 3Drag, che tramite la soluzione qui descritta potranno diventare Smart. Per "rendere smart" intendiamo dotare di intelligenza qualcosa (il termine "smart" in inglese significa proprio 'intelligente') in modo da poter migliorare quell'oggetto. Nel caso delle stampanti 3D ci riferiamo all'aggiunta di OctoPrint,

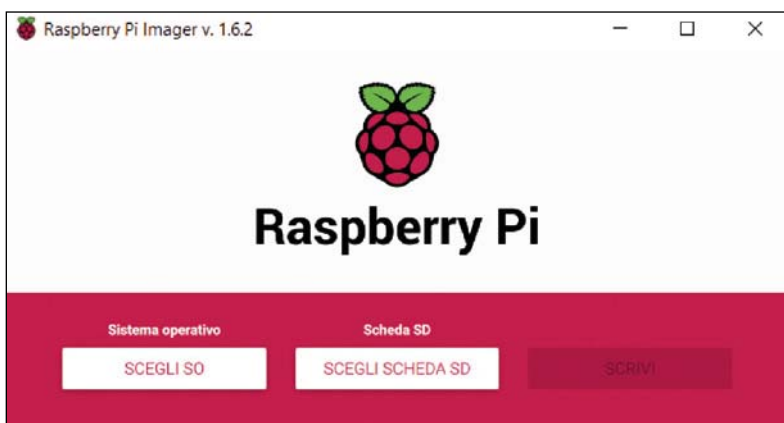


Fig. 1
Pagina di
avvio di
Raspberry Pi
Imager.

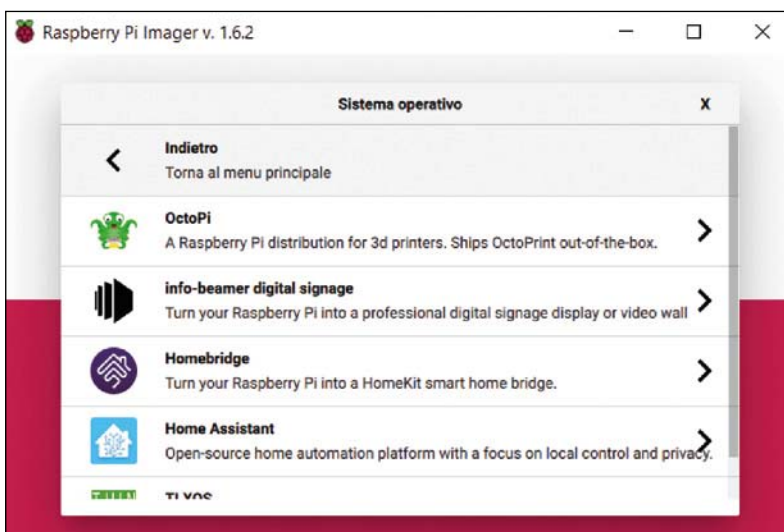
ovvero una modifica sia a livello hardware che a livello software che ci consente di rendere più autonome le nostre macchine. In queste pagine vi presenteremo OctoPrint, spiegheremo come installarlo e vi daremo alcuni consigli di utilizzo, soffermandoci sui plugin che vi si possono installare per rendere più versatile e potente lo strumento.

COS'È E A COSA SERVE OCTOPRINT?

OctoPrint viene definito dai suoi creatori come un'interfaccia web elegante per la tua stampante 3D, e si tratta quindi di un software gratuito che gira su una Raspberry Pi grazie alla distribuzione OctoPi.

Grazie all'installazione di questa modifica la stampante 3D sarà controllabile da remoto in tutte le sue attività: accensione e spegnimento, start e stop delle stampe, settaggio dei valori di temperatura, modifica dei file .stl e volendo, anche controllabile tramite l'aggiunta di una

Fig. 2
Scelta del
sistema
operativo.



fotocamera che permetterà non solo di vedere in real time la stampa, ma anche di fare il cosiddetto time lapse (tecnica di fotografia ad alta velocità, ovvero un video composto da immagini scattate dopo ogni tot secondi e poste una di seguito all'altra per dare l'effetto di movimento, come in un normale filmato, ma più velocemente) e fotografie nel corso di essa.

COME SI INSTALLA

Come prima cosa dobbiamo procurarci il materiale necessario alla realizzazione di questa modifica.

Avremo bisogno di una Raspberry Pi (le schede consigliate sono Pi 3A+, 3B, 3B+ e 4B, sebbene quest'ultima a fronte delle ottime prestazioni presenta spesso un eccessivo riscaldamento che può però essere ovviato con una ventola o un apposito dissipatore) una scheda micro SD (anche una da 8GB va bene ma dipende da quante foto e video vorrete poi farci e anche dalla quantità di file .gcode che vorrete salvare) una telecamera (opzionale), un cavo di alimentazione per il Raspberry Pi e un cavo USB per connetterlo alla stampante, inoltre, se si vuole, su <http://thingiverse.com/> si trovano centinaia di case per il Raspberry Pi e per la telecamera da stampare in 3D.

Per prima cosa bisogna installare Raspberry Pi Imager, un software per la gestione dei file immagine <https://www.raspberrypi.org/software/>. Una volta avviato bisogna selezionare come sistema operativo OctoPi; per fare questo basta cliccare sul pulsante "Scegli SO" che si trova nella pagina di avvio (**Fig. 1**), che apre una finestra di dialogo nella quale bisogna andare a cliccare sulla voce "Other specific Purpose OS" e successivamente selezionare OctoPi (**Fig. 2**) e scegliere quindi il primo risultato.

Una volta fatto ciò, premendo la combinazione di tasti ctrl+shift+x, ci si aprirà un menù dentro al quale andremo a spuntare la sezione del WiFi ed andremo ad inserire l'SSID e la password della nostra rete domestica. Per ultimo bisogna inserire la scheda SD nel computer e selezionarla sul Raspberry Pi Imager per scrivere il file immagine (quest'ultima verrà formattata prima dell'installazione). Da questo momento la nostra Raspberry Pi è configurata per gestire OctoPrint.

Fatto ciò è possibile fare tutti i collegamenti ed accendere il sistema, SD nella Raspberry Pi, cavo di alimentazione, cavo USB alla stampante e telecamera; ora bisogna trovare l'indirizzo IP

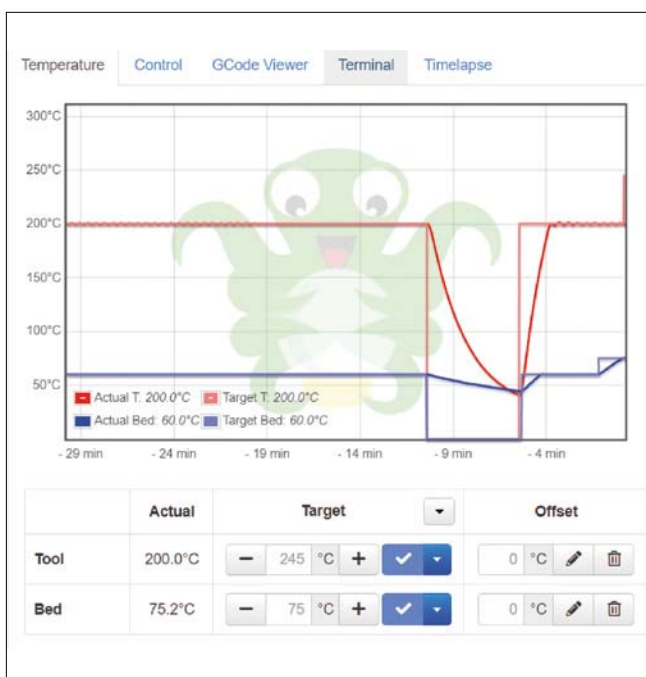


Fig. 3
Sezione Temperatura.

che è stato assegnato al dispositivo e ci sono diversi metodi per farlo: tramite SSH, tramite le impostazioni del router oppure tramite delle applicazioni per il telefono tipo Fing, l'importante è recarsi a quell'indirizzo dal nostro browser. Una volta trovato ci si aprirà l'interfaccia web di OctoPrint (al primo avvio dovremo registrarci al servizio).

Nella schermata di OctoPrint troviamo quindi diverse sezioni:

- Temperatura: (**Fig. 3**) si tratta della schermata dove è possibile regolare la temperatura della nostra stampante, sia del piatto di stampa che del nozzle, visibili anche dal grafico posto sopra;
- Control: (**Fig. 4**) è la sezione per vedere la telecamera e per muovere i vari assi tramite le frecce presenti, infatti è come la sezione control della stampante 3D;
- GCode Viewer: (**Fig. 5**) si utilizza durante una stampa e mostra il percorso che sta compiendo l'hotend in real time del layer che sta stampando, ma possono essere visualizzati anche i precedenti e i successivi;
- Terminal: è una sezione avanzata per dare

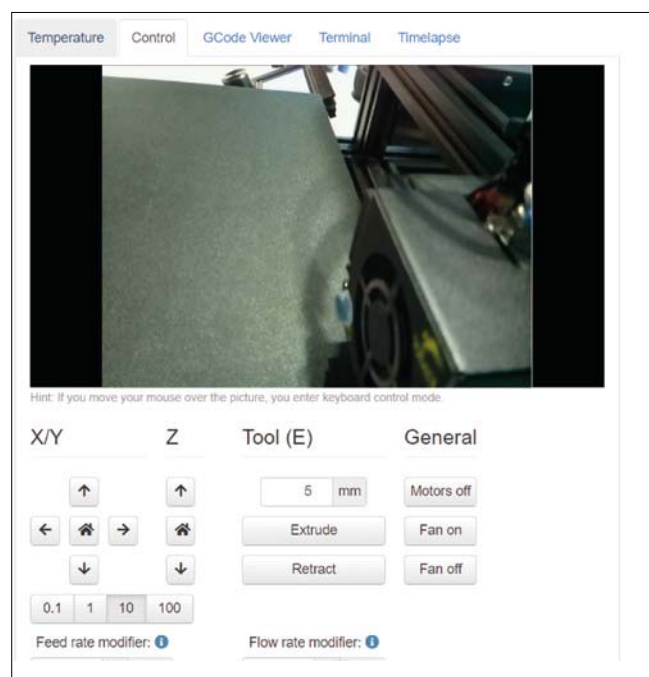


Fig. 4
Sezione Control.

- comandi tramite il terminale alla stampante;
- Timelapse: (**Fig. 6**) è dedicato alle impostazioni per eseguire un timelapse durante la stampa, si possono scegliere il numero di scatti, la frequenza con cui farli e anche un eventuale spostamento dell'hotend per evitare di farlo finire negli scatti.

Sul lato sinistro (**Fig. 7**) invece si trovano le informazioni riguardanti la stampa in corso o appena conclusa e i vari file caricati sul programma.

COME STAMPARE CON OCTOPRINT

Per stampare bisogna seguire all'incirca il procedimento tradizionale di stampa senza considerare la presenza di OctoPrint: si fa quindi il disegno tramite degli appositi editor per il disegno 3D

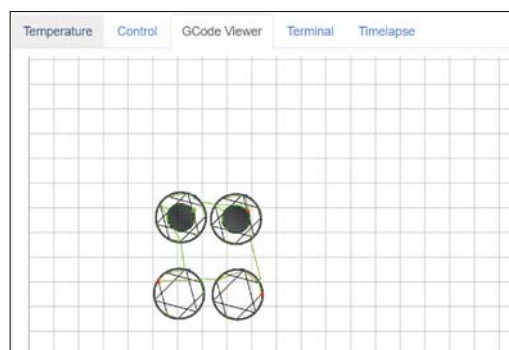
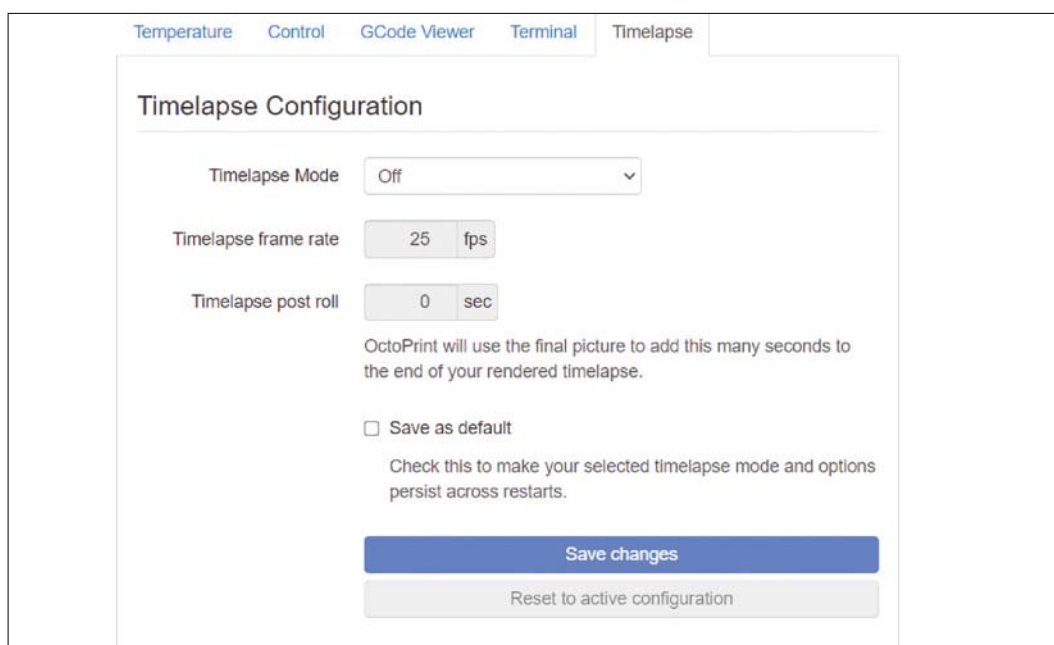


Fig. 5
Sezione Gcode Viewer.

Fig. 6
Sezione
Timelapse.



(come ad esempio SketchUp o Fusion 360), lo si esporta in .stl, lo si verifica (se necessario) con programmi per sigillarlo e infine lo si elabora con un software apposito come, ad esempio, Cura o PrusaSlicer per ottenere il file .gcode. Una volta preparato il file lo si può caricare su

OctoPrint cliccando sul pulsante upload dell'interfaccia utente o semplicemente trascinandolo nella schermata e, se si è pronti a stamparlo, basta cliccare il pulsante con la stampante posto di fianco al nome del file (**Fig. 8**); fatto, ciò il software farà tutto quanto da solo e voi potrete controllare gli aggiornamenti dello stato della stampa tramite le varie sezioni.

Fig. 7
Informazioni sulla
stampa.

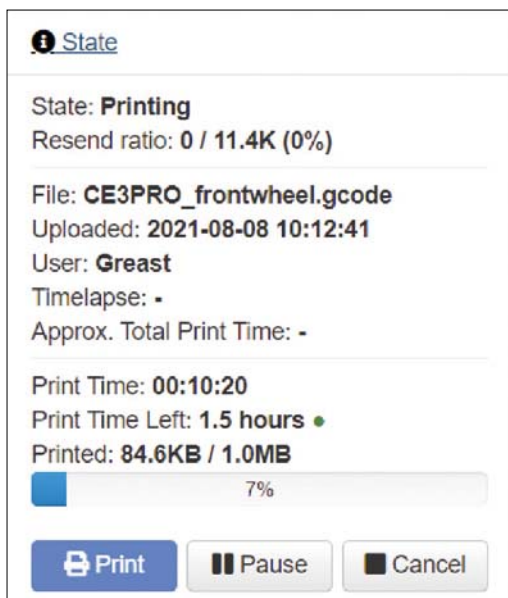
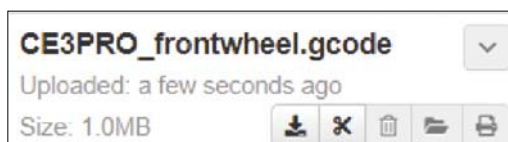


Fig. 8
Il comando di
stampa.



I MIGLIORI PLUGIN

Con OctoPrint abbiamo a disposizione diversi plugin, ovvero altre funzionalità da poter aggiungere a quelle illustrate precedentemente e in generale a quelle predefinite dall'installazione della piattaforma software per poter migliorare ancora di più la nostra esperienza con OctoPrint. Per installare un plugin (qualunque esso sia) bisogna cliccare sull'icona raffigurante una chiave inglese posta nella toolbar superiore della schermata, per poi selezionare la voce "plugin manager" posta sotto la sezione OctoPrint sulla sinistra. Come primi ci si presenteranno i plugin già installati, alcuni sono lì già per impostazione predefinita, mentre se clicchiamo sul pulsante 'get more', possiamo accedere ai vari plugin disponibili sia nella repository, sia da siti web esterni aggiungendone l'URL.

I plugin installati possono essere trovati sempre cliccando l'icona riportante la chiave inglese, ma questa volta cercando nella sezione dei plugin sempre nel menù di sinistra, facendo ciò è possibile modificarli e/o disinstallarli.



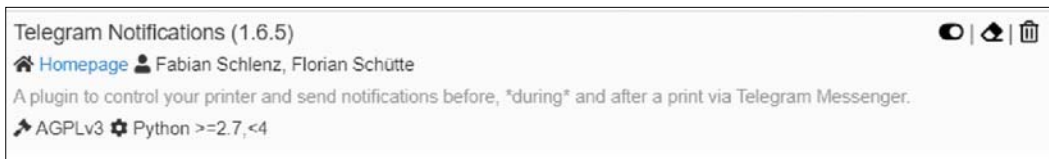


Fig. 9
Plugin Telegram notifications.

In questa guida ve ne presentiamo tre, tutti da provare assolutamente.

Il primo si chiama “Telegram notifications” (**Fig.9**) e ci permette di ricevere notifiche in tempo reale dalla stampante. Come prima cosa dovremo creare un bot di Telegram, ovvero un’applicazione di terze parti eseguita sempre su Telegram, tramite il canale BotFather. Su Telegram bisogna cliccare la lente d’ingrandimento e cercare BotFather, per poi selezionare quello col simbolo del verificato. Cliccare quindi sul pulsante start per inizializzare il bot, inviare poi il comando `/newbot`; ci verrà chiesto di dare un nome al bot (consigliamo di farlo finire con ‘bot’) e successivamente di dare uno username al bot (consigliamo di riprendere

il nome del bot aggiungendo “_bot” alla fine), se tutto è andato a buon fine ci verrà inviato un messaggio contenente un Token da utilizzare successivamente. Una volta creato il bot e installato il Plugin possiamo recarci nelle sue impostazioni selezionandolo dalla voce ‘plugin’ dopo aver cliccato la chiave inglese. Qui andremo ad inserire il token del bot creato e cliccando la nuvoletta potremo scegliere i vari comandi da fare o meno, quindi scegliere se ad esempio ricevere degli aggiornamenti continui, se poter mettere in pausa una stampa e molti altri.

Il secondo è un plugin dedicato a Cura, che ci permetterà di evitare di dover salvare ogni file .gcode sul nostro computer perché quest’ultimo

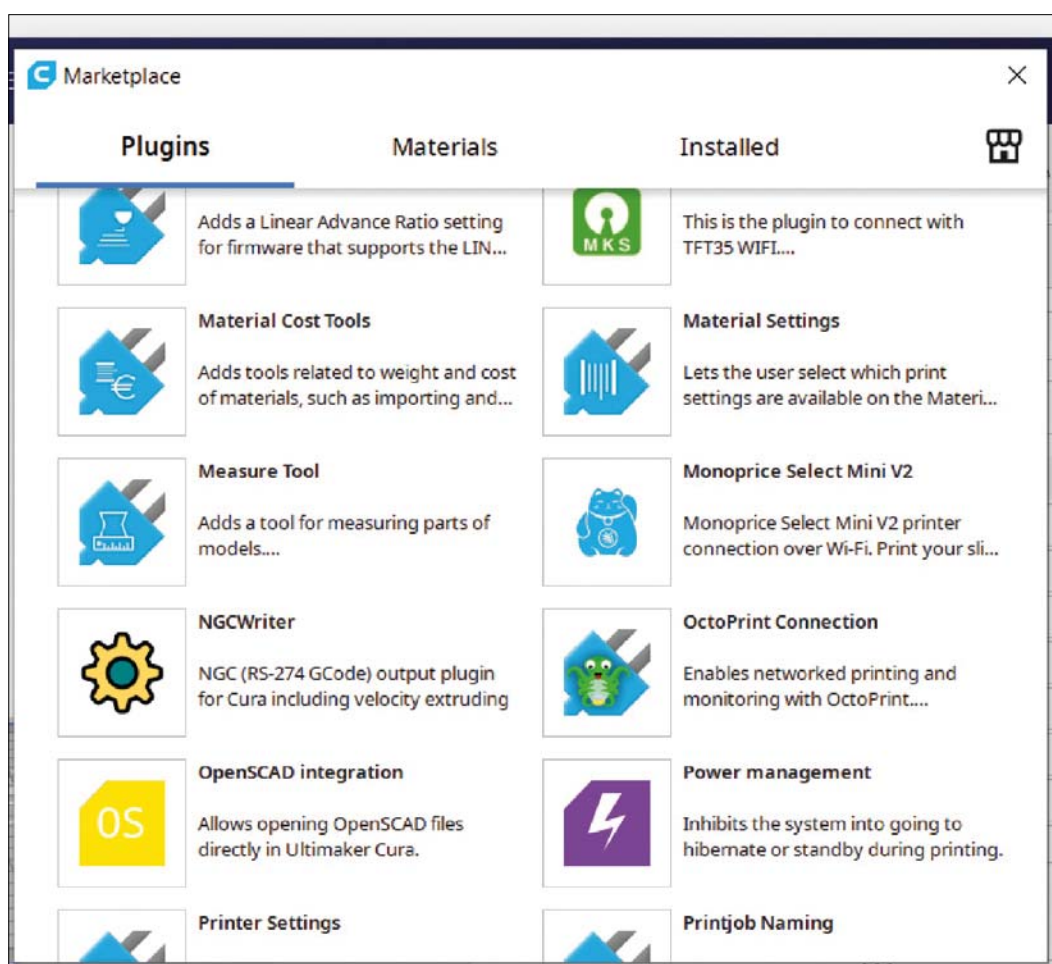
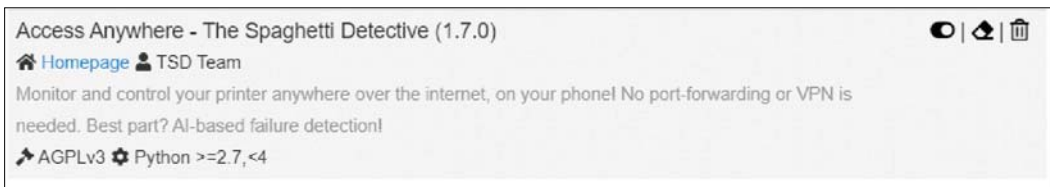


Fig. 10
Il plugin OctoPrint connection.

Fig. 11
Il plugin Spaghetti Detective.



lo manderà direttamente in stampa tramite OctoPrint. Per ottenere questo plugin dobbiamo andare sull'applicazione di Cura e successivamente sul suo marketplace, dobbiamo quindi cercare il plugin denominato "OctoPrint connection" per poi installarlo (**Fig. 10**).

Una volta riavviato il programma dobbiamo cliccare sul nome della stampante (in alto a sinistra), fare "Manage Printers" e successivamente "Connect to OctoPrint". Nella nuova finestra dovremo settare l'API Key dell'OctoPrint e la maniera più veloce per farlo è selezionare Request così da accedere direttamente tramite il browser con le nostre credenziali e successivamente accettare la richiesta di connessione.

In alternativa su OctoPrint cliccando la chiave inglese e nella sezione API possiamo trovare il codice che ci serve da incollare su Cura. Da questo momento quando prepareremo un file per la stampa potremo passarlo direttamente su OctoPrint piuttosto che sul computer o sulla SD della stampante, tra l'altro l'interfaccia di OctoPrint verrà aperta direttamente nel programma Cura. L'ultimo plugin invece è lo "Spaghetti Detective" (**Fig. 11**), un plugin in grado di rilevare se ci sono problemi durante la stampa e principalmente se ci sono degli spaghetti, ovvero se il nozzle (ossia l'ugello estrusore della testa di stampa...) ha iniziato a lasciare filamento ovunque tranne che sulla stampa, facendo quindi l'effetto "spaghetti". Per poter utilizzare questo plugin, la telecamera è obbligatoria.

Concludendo l'argomento "plugin" ci teniamo a segnalarvi un plugin necessario per le stampanti targate Creality; infatti su quest'ultime OctoPrint non riesce a rilevare le temperature correttamente, e per ovviare a questo problema basta installare un plugin dedicato a ciò, chiamato per l'appunto CrealityTemperature.

PERCHÉ SCEGLIERE OCTOPRINT?

Nella realtà dei fatti OctoPrint non è l'unica soluzione gratuita in grado di rendere smart una stampante 3D ma ne esistono di diverse, quindi perché scegliere quest'ultima?

MatterControl è un software open source che può essere installato solo su dispositivi Linux, quindi sul proprio computer senza dover aggiungere altro, oppure si può comprare l'apposito controller, ma non è predisposto per il controllo in remoto della stampante 3D.

Repetier Host è un pacchetto software che include il firmware, l'host e un'app per il controllo della stampante 3D. Esiste in due versioni, quella di base con poche funzionalità ma gratuita, oppure quella pro con più funzionalità ma a pagamento. Creality Box è un plugin con WiFi e telecamera inclusa, che offre anche la possibilità di controllare la stampa da remoto, quindi anche se ci troviamo fuori casa e pertanto da una rete diversa; l'unico limite di questa soluzione è che funziona solo sulle stampanti marchiate Creality.

Astroprint nasce sulla base di OctoPrint, quindi ha tutte le caratteristiche di quest'ultimo più altre funzionalità, tuttavia necessita del suo Astrobox per poter funzionare. Può essere installato anche su una classica scheda Raspberry Pi, ma le sue potenzialità diminuiscono. Inoltre il software è gratuito se utilizzato su una sola stampante, mentre diventa a pagamento per più stampanti oppure per avere più spazio. ■



Cosa occorre?

I componenti utilizzati in questo progetto sono disponibili presso Futura Elettronica. Le board Raspberry Pi sono disponibili a questo link bit.ly/rpiboards, la telecamera utilizzabile per la funzione TimeLaps (cod. YG016) è in vendita a Euro 9,50.

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>

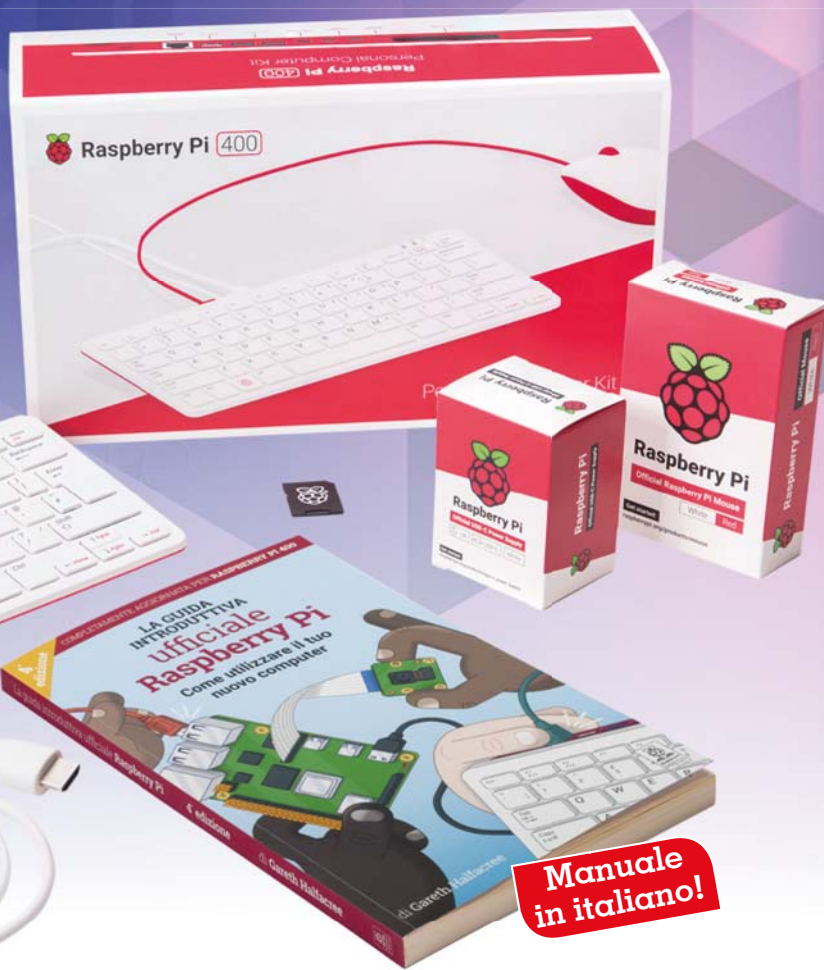
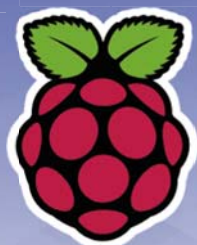
RASPBERRY PI 400!

La tastiera con il computer integrato

€ 146,00

Cod. PI400KIT-ITA

IVA inclusa



**Manuale
in italiano!**

Prezzo IVA inclusa.

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

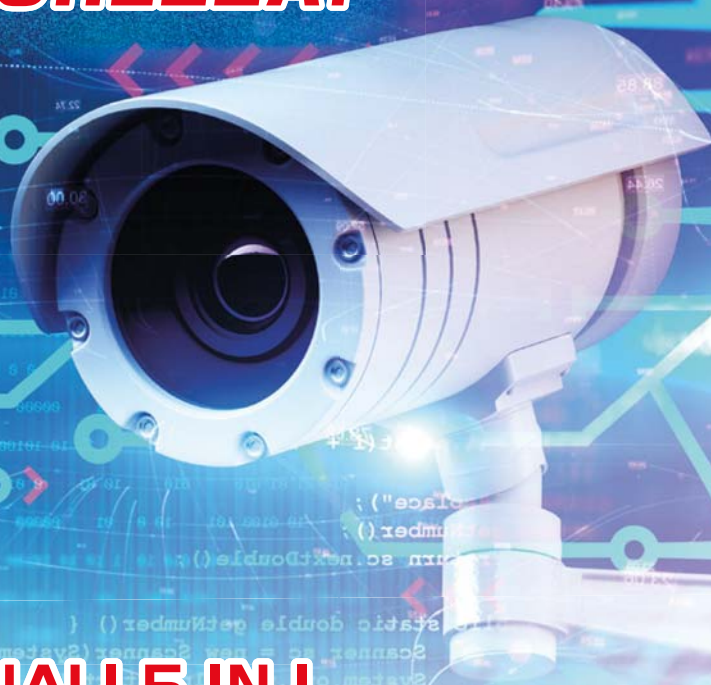


Raspberry Pi 400 è un personal computer completo che integra nella tastiera (italiana), una scheda Raspberry Pi 4 da 4Gb con processore quad-core a 64 bit, Wi-Fi, Bluetooth, doppia uscita video (micro HDMI) con una risoluzione fino a 4K, Gigabit Ethernet, USB 2.0 e USB 3.0.

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

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

PROBLEMI DI SICUREZZA? A SORVEGLIARE CI PENSA LUI!



DVR A 4 CANALI 5 IN 1



€69,00

Cod. FR810-4



Videoregistratore a 4 canali 5 in 1 (AHD / CVI / TVI / IP / Analogico), dalle caratteristiche professionali. Dispone di 4 ingressi video (BNC) che possono essere utilizzati sia in modalità HD (AHD / CVI / TVI) che in modalità analogica. È dotato di interfaccia Ethernet (10-100 Base-T) che consente di gestire fino a 16 telecamere IP. Visione remota tramite PC connesso a Internet e Smartphone (iOS, Android) con possibilità di effettuare registrazioni e backup. Funzione OSD (On Screen Display) gestibile tramite mouse. La confezione comprende: DVR, mouse, alimentatore da rete (con uscita 12 VDC - 2A) e guida rapida di installazione in italiano. Hard Disk acquistabile separatamente.



VERSIONE A 8 CANALI



VERSIONE A 16 CANALI



DISPONIBILI ANCHE NELLE
VERSIONI A 8 CANALI

COD. FR810-8 € 85,00

E A 16 CANALI

COD. FR810-16 € 148,00



CONOSCERE E USARE

di FRANCESCO FICILI

Node-RED

2

Iniziamo a scoprire Node-RED, un tool di flow-based programming, molto orientato all'IoT ed alla connettività, originariamente sviluppato dall'IBM Emerging Technology Services team e adesso parte della JS Foundation.

In questa puntata ci addentriamo maggiormente nell'uso del tool, esaminando i nodi di utilizzo più comune, vediamo come aggiungerne di nuovi e scopriamo cosa sono i context in Node-RED. Seconda puntata.

Abbiamo iniziato questo corso introducendo, nella prima puntata il tool di flow-based programming Node-RED, descrivendone i concetti di base, quindi passando alla pratica installando l'ambiente ed analizzando le caratteristiche dell'editor grafico. Come di consueto, ci siamo quindi spostati sul lato pratico, realizzando anche un semplice esempio di utilizzo (il classico esempio di Hello World usato nei sistemi embedded, ossia il lampeggio di un LED) su Raspberry Pi. Ritenendo di avervi dato le basi per proseguire proficuamente la trattazione, in questa seconda puntata approfondiremo la nostra conoscenza dell'ambiente concentrandoci sulle principali categorie di nodi a disposizione spieghiamo come aggiungere nuovi nodi installandoli sulla palette ed introduciamo il concetto dei context, che ci permettono di immagazzinare informazioni con diversi gradi di visibilità all'interno dei nostri flow.

FLUSSI, NODI E MESSAGGI

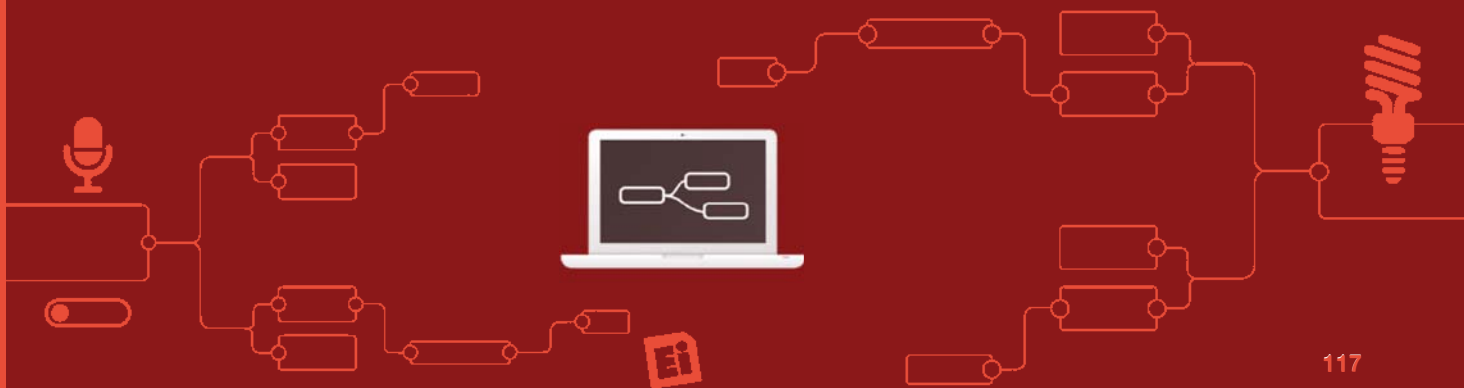
Come abbiamo brevemente accennato nella puntata precedente, gli elementi fondamentali che costituiscono un programma Node-RED sono:

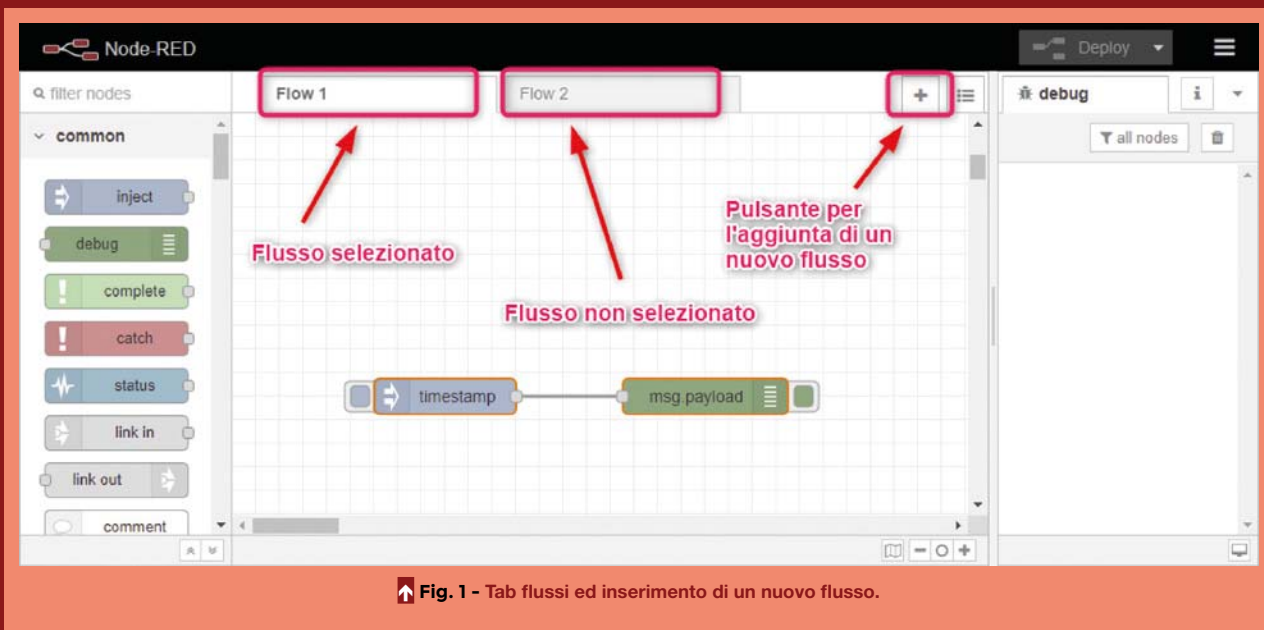
- Flussi;
- Nodi;
- Messaggi.

Espandiamo questi concetti di base, in modo da comprendere meglio queste fondamentali strutture ed il loro funzionamento.

Flussi

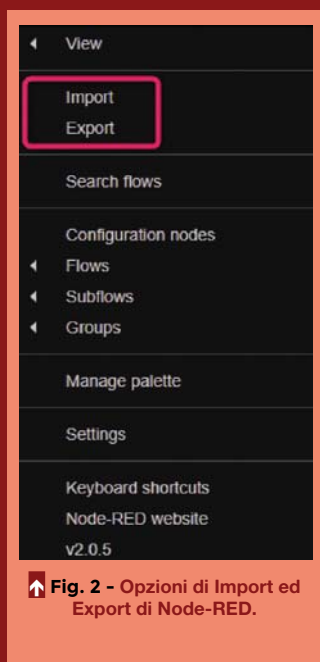
Un flusso (o flow) è il principale meccanismo per contenere una struttura connessa di nodi. Al livello dell'editor di Node-RED è rappresentato dal workspace, all'interno del quale i nodi possono essere trascinati (prelevandoli dalla palette) e connessi tra di loro per costituire il codice grafico. Una singola istanza di Node-RED può contenere diversi flows,





rappresentati come una serie di tab nella parte alta del workspace stesso. Node-RED mostra sul workspace, di volta in volta, i nodi e le connessioni del flow attivo. Per aggiungere un nuovo flow al workspace bisogna premere il tasto '+' situato nell'angolo in alto a destra del workspace (alternativamente può essere utilizzata l'opzione Flows → Add del menu principale), come illustrato in Fig. 1. Il termine flusso può anche essere utilizzato per indicare informalmente un insieme atomico di nodi connessi, di conseguenza un flusso (tab) può essere costituito da un insieme di flussi (intesi come molteplici set i nodi interconnessi).

In Node-RED un flusso può essere salvato come un file JSON (Javascript Object Notation). Questo permette di memorizzare e scambiare molto facilmente i flows come file JSON. I flussi possono essere esportati ed importati tramite le due opzioni "Import" ed "Export" presenti sul menu principale (Fig. 2). Eseguendo un doppio click sul tab di un flusso si accede alla sua finestra delle proprietà, dalla quale è possibile eseguire una serie di operazioni aggiuntive, come rinominare il flusso, abilitarlo o disabilitarlo o anche cancellarlo (Fig. 3). Sempre dalla finestra delle proprietà è possibile inserire una descrizione del flusso, in modo da



creare una utile documentazione utente del programma che stiamo sviluppando. La descrizione usa la sintassi di Markdown e compare nella sezione info della sidebar.

Subflows

I flow possono anche essere collassati in un singolo nodo ed in questo caso si parla di subflows. I subflows sono usati per ridurre la complessità di un flow o per impacchettare un flow (inteso come gruppo di nodi) all'interno di un singolo nodo da riutilizzare più volte in altri flows. Una volta creati, i subflows appaiono nella palette come un nodo disponibile e possono essere aggiunti al workspace come i nodi standard.

Nodi

Un flow Node-RED è popolato da Nodi (l'elemento principale di Node-RED) e relative connessioni. I nodi costituiscono le parti eseguibili di un flow e si scambiano messaggi tramite wires (fili) che vengono collegati alle porte di ingresso e di uscita di un nodo. I nodi sono triggerati sia ricevendo un messaggio su una porta d'ingresso o tramite un evento interno (come ad esempio lo scadere di un timer, il cambio di stato di una line digitale, o una richiesta proveniente da un canale di comunicazione). Quando viene triggerato, il nodo elabora il messaggio ricevuto o l'evento interno, e successivamente, può inviare un messaggio ai nodi successivi del flow. In qualche caso i nodi hanno degli elementi integrati nella loro icona, con i quali l'utente può interagire direttamente dal flow. Questi elementi possono essere pulsanti, elementi di output, eccetera.

Ci sono tre modi per aggiungere un nodo ad un flow:

- Prelevandolo dal Node Palette
- Tramite la quick-add dialog box

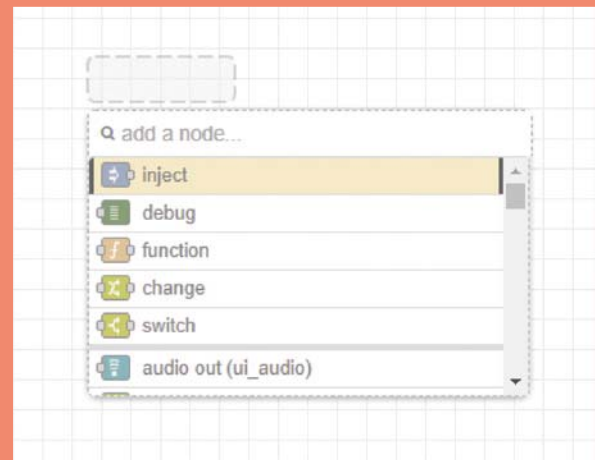


Fig. 4 - Quick-add dialog box.

- Tramite la funzione Import

La prima opzione è quella più comune: si seleziona il nodo dalla Node Palette premendo e mantenendo premuto il pulsante sinistro del mouse e lo si trascina sul workspace, rilasciando il pulsante per posizionare il nodo. Per utilizzare la quick-add dialog box invece è sufficiente tenere premuto il tasto Ctrl e cliccare con il tasto sinistro all'interno del workspace. Questo farà comparire una finestra di selezione come quella riportata in Fig. 4 (la quick-add dialog box, appunto...) dalla quale è possibile selezionare il nodo da aggiungere. L'opzione di import infine permette di importare un nodo (o più comunemente un intero flow)

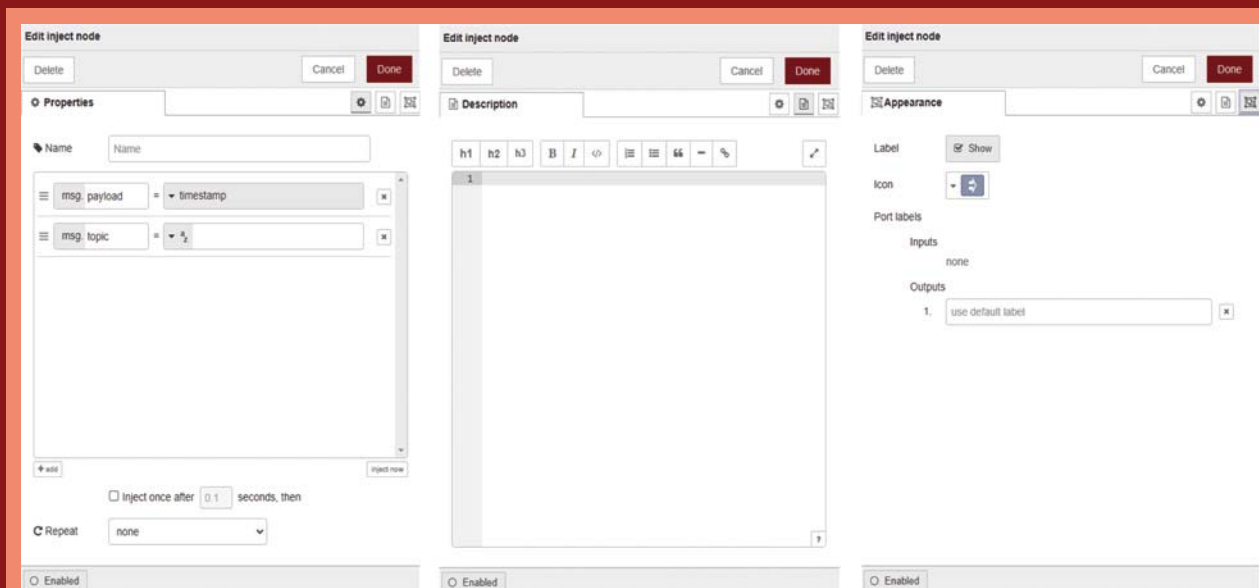


Fig. 5 - Finestra delle proprietà (sezioni Properties, Description ed Appearance) per un nodo Inject.

precedentemente salvato sottoforma di file JSON (tramite la funzione **duale export**). Esattamente come i flows, anche i nodi dispongono di una finestra delle proprietà, alla quale si può accedere eseguendo un doppio click sul nodo specifico (o alternativamente premendo il tasto **invio** quando il nodo ha il focus). La finestra delle proprietà di un nodo è divisa in tre sezioni principali, che descriviamo qui di seguito.

Properties: da questa sezione è possibile editare le proprietà specifiche del nodo in questione. Questa sezione varia da nodo a nodo, poiché nodi differenti hanno differenti opzioni di configurazione e proprietà.

Description: da questa sezione è possibile editare la documentazione del nodo, formattata con sintassi **Markdown**. La descrizione inserita in questa sezione comparirà nella sezione **info** della sidebar, come documentazione del nodo. Esattamente come per i flows, questa funzionalità permette di creare una documentazione utente molto ben fatta dei nostri programmi.

Appearance: da questa sezione è possibile customizzare l'aspetto grafico del nodo. Ad esempio è possibile definire se visualizzare o meno la label del nodo, è possibile selezionarne l'icona (nel caso non si voglia usare quella di default) ed inoltre è possibile definire delle label specifiche per ogni porta.

La **Fig. 5** rappresenta le tre sezioni **Properties**, **Description** ed **Appearance** della finestra delle proprietà per un nodo **Inject**.

Configuration Nodes

Non sempre i nodi sono visibili sul workspace, infatti esiste un tipo particolare di nodo, chiamato **configuration node**, che non ha una parte grafica (icona e porte di connessione), ma costituisce un insieme di configurazioni che possono essere riutilizzate da più nodi in un flow. I nodi che necessitano un **configuration node** per funzionare ne permettono la creazione direttamente dalla loro finestra delle proprietà. Ad esempio i nodi **serial in** e **serial out**, che necessitano di un **configuration node serial port** (che contiene le opzioni di configurazione della porta seriale, che possono essere utilizzate da diverse istanze dei nodi di comunicazione seriale), ne permettono la creazione (o, se il nodo è già esistente, l'associazione) direttamente dalla loro finestra delle proprietà, come visibile in **Fig. 6** e **Fig. 7**. Non avendo parte grafica, la finestra delle proprietà dei **configuration nodes** è costituita dalla sola parte **Properties**. Una lista completa dei **configuration nodes** presenti può essere ricavata dalla sezione **configuration nodes** della sidebar.

Messaggi

Durante il funzionamento di un flow, i vari nodi che lo compongono si scambiano messaggi, e questa è una ca-

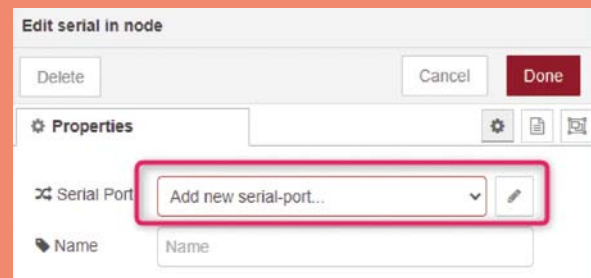


Fig. 6 - Creazione del **configuration node serial port** dalla finestra di proprietà del nodo **serial in**.

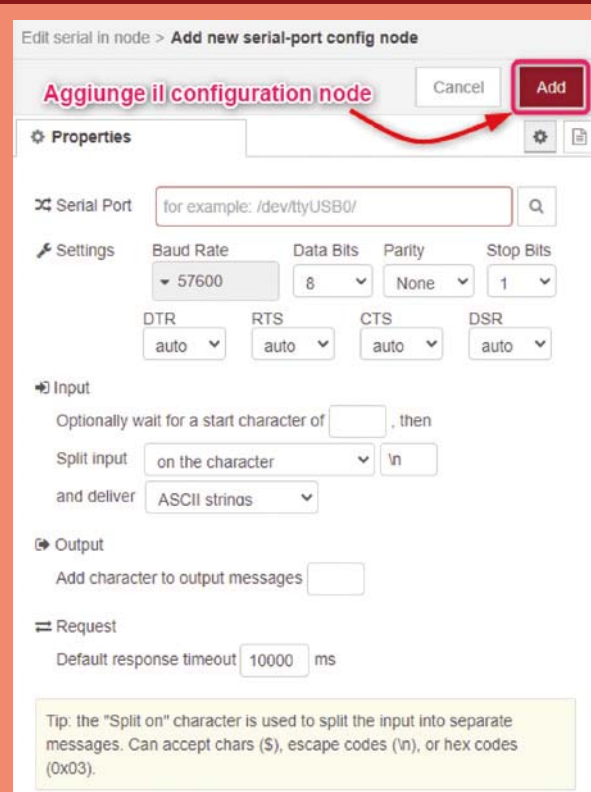


Fig. 7 - Finestra delle proprietà del **configuration node serial port**.

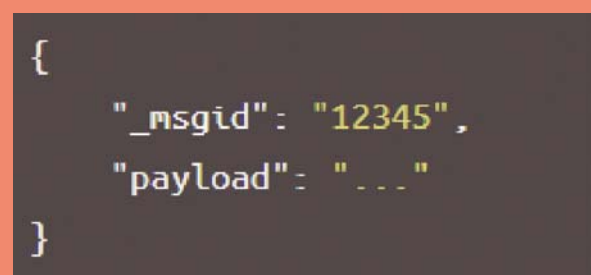


Fig. 8 - Esempio di messaggio Node-RED.

ratteristica peculiare di Node-RED e della FBP. I messaggi che i nodi si cambiano sono semplici oggetti JavaScript che possono avere qualunque set di proprietà. Tipicamente una proprietà denominata "payload" è sempre presente, ed è la proprietà con cui più frequentemente i nodi lavorano, ma possono essercene molte altre. Node-RED inoltre aggiunge ad ogni messaggio una proprietà "_msgid", che sostanzialmente identifica in maniera univoca il messaggio all'interno di un flow (Fig. 8). Il valore di una proprietà di un messaggio può essere un qualsiasi tipo JavaScript, ad esempio:

- Booleano – true/false
- Numerico – 1, 123.4
- Stringa – "Hello World"
- Array – [1,2,3,4,5,6,7,8,9]
- Object – {"a" : 1, "b" : 2}
- NULL

CORE NODES

Come abbiamo visto in precedenza, l'elemento principale di un flow Node-RED sono i nodi. Node-RED contiene una ricca dotazione di nodi predefiniti e moltissimi altri nodi possono essere aggiunti alla palette. Per ragioni di spazio è impossibile descrivere nel dettaglio tutti i nodi a disposizione nell'ecosistema di Node-RED, nonché anche poco utile in pratica, dato che si tratta di un ambiente in continua evoluzione. Esiste tuttavia un set di nodi standard, a cui ci si riferisce spesso con il termine "Core Nodes" che è importante conoscere (Fig. 9). Questi nodi sono:

- Inject;
- Debug;
- Function;
- Change;
- Switch;
- Template.

Qui di seguito li descriviamo uno per uno.

Inject

Il nodo Inject permette di inviare messaggi all'interno di un flow, sia manualmente (interagendo con il pulsante integrato nell'icona del nodo), che automaticamente. Il messaggio inviato può avere qualsiasi proprietà si desideri e le proprietà possono essere settate in vari modi. Ad esempio una proprietà di un messaggio del nodo inject può essere:

- uno dei tipi base supportati da JavaScript (ad esempio un boolean, un valore numerico, una stringa, un array...);
- un timestamp;
- una stringa JSON;
- una espressione JSONata.

In questo modo possono essere iniettati nel nostro flow messaggi contenenti qualsiasi dato.

Di conseguenza il nodo inject è molto utilizzato per il debug

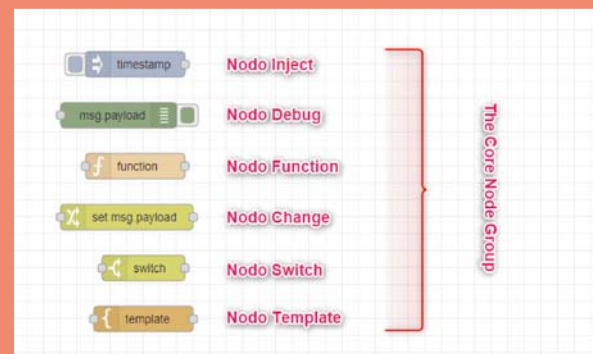


Fig. 9 - I Core Nodes.

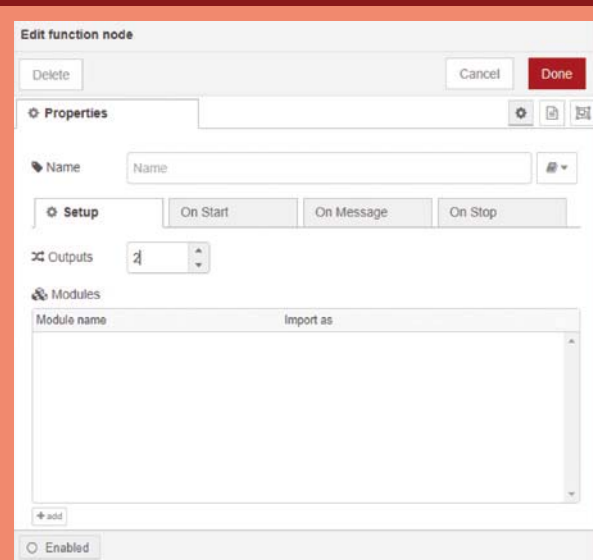


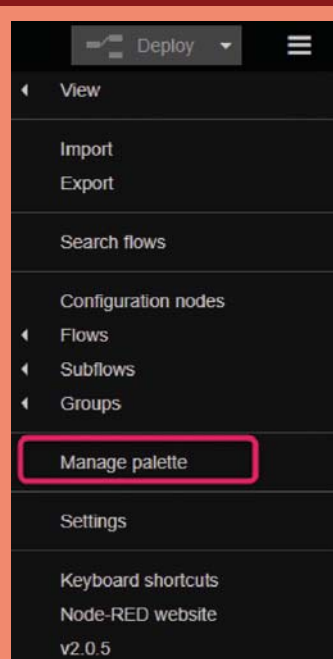
Fig. 10 - Finestra delle proprietà del nodo Function.

e per testare il funzionamento di altri nodi, pur trovando anche altri interessanti utilizzi (ad esempio per l'inject periodico di valori all'interno del flow, come abbiamo visto nell'esempio pratico della scorsa puntata).

Debug

Il nodo Debug permette di visualizzare i messaggi che riceve dalla sua porta in ingresso nella visuale di debug della sidebar.

La finestra di debug della sidebar permette di visualizzare in maniera strutturata le informazioni contenute nel messaggio, rendendo quindi il nodo di debug molto utile per il debugging dei flow (come era anche facilmente intuibile dal suo nome).



↑ Fig. 11 - Opzione Manage Palette tra le opzioni di Node-RED.

Function

Il nodo Function (Fig. 10) permette di scrivere del codice JavaScript che agisce direttamente sul messaggio passato al nodo stesso.

Il messaggio è passato in un oggetto chiamato `msg`, e possono essere creati quanti output si vuole.

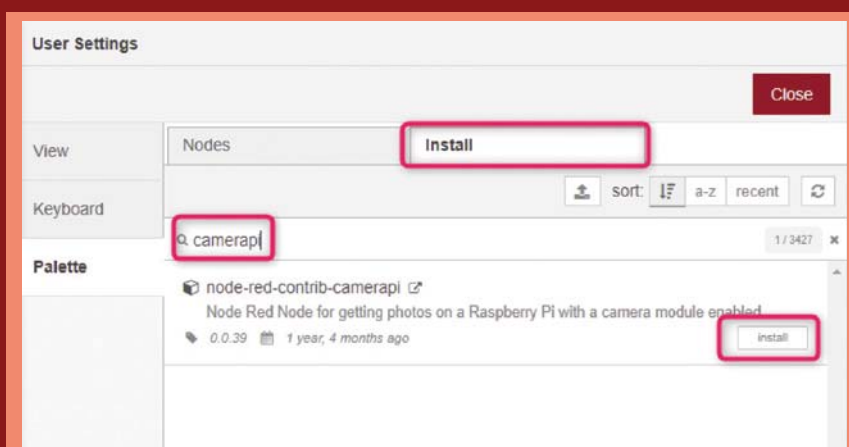
Dalle versioni più recenti di Node-RED è possibile importare moduli JavaScript da utilizzare all'interno del blocco ed è inoltre possibile invocare codice JavaScript all'avvio ed allo stop del nodo (ad esempio allo scopo di inizializzare delle variabili di stato e via di seguito).

Change

Il nodo Change è utilizzato per modificare una proprietà di un messaggio senza dover ricorrere al nodo function. Ogni nodo può essere configurato per eseguire una serie di operazioni in ordine.



↑ Fig. 13 - Nodo camerapi installato.



↑ Fig. 12 - Installazione del nodo "node-red-contrib-camerapi".

Le operazioni disponibili sono:

- **Set:** settare una proprietà ad un certo valore. Il valore può essere una varietà di tipi differenti, o può essere preso ad un'altra proprietà.
- **Change:** cercare e rimpiazzare parte di una proprietà di un messaggio.
- **Move:** muovere o rinominare una proprietà.
- **Delete:** cancellare una proprietà

Switch

Il nodo Switch permette di indirizzare messaggi verso differenti porte in uscita, in base alla valutazione di regole interne al nodo. A tale riguardo va precisato che sono previsti i quattro tipi di regola descritti qui di seguito.

- **Valore:** Il valore di una specifica proprietà può essere controllato con espressioni comparative o booleane (ad esempio maggiore di, minore di, uguale, vero falso, etc..).
- **Sequenza:** la sequenza di arrivo dei messaggi può essere controllata, estraendo solo alcuni elementi della sequenza stessa, dalla testa, dalla coda o da un numero arbitrario di elementi intermedi. Vengono triggerate le porte di uscita per cui si verifica un match rispetto al controllo di sequenza.
- **JSONata:** il messaggio viene controllato per mezzo di una espressione JSONata (viene fornito un apposito editor per impostare e testare la query) e viene triggerato lo specifico output solo se l'espressione ritorna true.
- **Otherwise:** questo è il classico caso di default delle espressioni switch dei linguaggi classici, viene triggerato se nessuna delle precedenti regole viene soddisfatta.

Template

Il nodo Template, infine, è un nodo che è in grado di genera-

Javascript Object Notation (JSON)

Il formato JSON (JavaScript Object Notation) è un formato di interscambio molto utilizzato in applicazioni client/server. È basato sul linguaggio JavaScript Standard ECMA-262 3ª edizione (dicembre 1999), ma ne è indipendente, trattandosi di un formato completamente indipendente dallo specifico linguaggio di programmazione utilizzato, sebbene le convenzioni che utilizza siano familiari alla famiglia dei linguaggi basati sul C. La semplicità di JSON ne ha decretato un rapido utilizzo specialmente nella programmazione in AJAX.

Il suo uso tramite JavaScript è particolarmente semplice, infatti l'interprete JavaScript è in grado di eseguirne il parsing tramite la funzione `JSON.parse()`. Questo lo ha reso velocemente molto popolare a causa della diffusione della programmazione in JavaScript nel mondo del Web.

La maggior parte dei linguaggi di programmazione possiede un typesystem molto simile a quello definito da JSON per cui sono nati molti progetti che permettono l'utilizzo di JSON con altri linguaggi quali, per esempio: ActionScript, C, C#, Adobe ColdFusion, Common LISP, Delphi, E, Erlang, Java, JavaScript, Lua, ML, Objective Caml, Perl, PHP, Python, REBOL e Ruby.



re un output testuale utilizzando le proprietà dei messaggi per riempire un template usando la sintassi del Mustache template language (maggiori informazioni sul Mustache template language possono essere reperite a questo indirizzo web: <https://mustache.github.io/mustache.5.html>). Questo nodo è molto utile per generare testi formattati in modi differenti partendo da tags contenuti su delle message properties. Ad esempio, un nodo template con all'interno questa dicitura:

```
Il messaggio di testo contenuto nel payload è: {{payload}}
```

A fronte della ricezione di un messaggio contenente nella

property payload la stringa "Ciao Mondo!!!", genererebbe il seguente output testuale:

```
Il messaggio di testo contenuto nel payload è: Ciao Mondo!!!
```

che andrà pertanto a sostituire al tag `{{payload}}` il contenuto di `msg.payload`.

AGGIUNGERE NODI ALLA PALETTE

Oltre a questi nodi di base, Node-RED contiene diversi altri nodi pre-installati di default, aventi interessanti funzioni aggiuntive, ma che non possono essere descritti uno per uno per ovvie ragioni di spazio.

Ma non è tutto, infatti la natura fortemente open dell'ambiente e la sua vasta community hanno portato alla produzione di una quantità di nodi aggiuntivi, che possono essere installati. Sarebbe assolutamente impossibile descriverli tutti, trattandosi di diverse migliaia di nodi, mentre può essere utile descrivere brevemente la procedura per aggiungere dei nodi alla palette che contiene quelli installati per impostazione predefinita.

L'aggiunta di un nuovo nodo può essere fatta selezionando l'opzione "Manage Palette" presente nelle opzioni di Node-RED (Fig. 11).

Una volta cliccato sull'opzione si aprirà una finestra di dialogo, ed in questa finestra bisogna selezionare il tab "Install" e poi cercare il nodo desiderato usando l'apposito motore di ricerca integrato.

Supponiamo, ad esempio, di voler installare un nodo per la gestione della camera pi (<https://www.raspberrypi.org/products/camera-module-v2>).

Per farlo selezioniamo l'opzione "Manage Palette", e nella finestra di dialogo che ci viene presentata selezioniamo il tab install e scriviamo "camerapi" nell'apposita textbox per la ricerca, come illustrato in Fig. 12.

Selezionando il pulsante "Install" di uno dei risultati della ricerca, Node-RED procederà all'installazione del nodo. Nel caso dell'esempio appena visto, ossia l'installazione del nodo camerapi, al termine dell'operazione troveremo il nodo di Fig. 13 tra i nodi della palette.

COSA SONO I CONTEXT

Passiamo adesso all'ultimo argomento di questa puntata, ossia i Context: Node-RED prevede un sistema per lo stoccaggio di informazioni in memoria che possono essere utilizzate da diversi nodi oppure da differenti istanze di esecuzione di un nodo stesso: si tratta dei context. Ebbene, esistono tre diversi tipi di context in Node-RED, cui sono associate tre differenti visibilità (o scope):

- Node: visibile solo da uno specifico nodo;
- Flow: visibile all'interno di uno specifico flow;
- Global: visibile globalmente da parte di tutti i nodi di Node-RED.

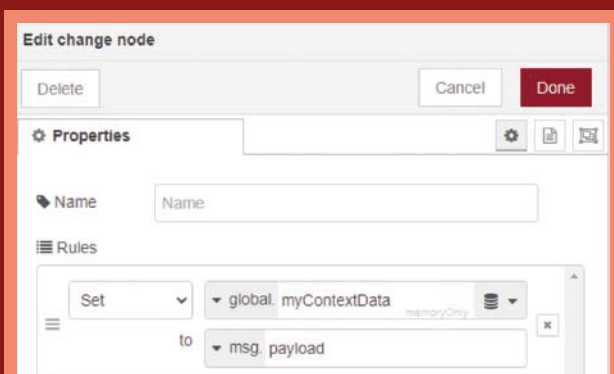


Fig. 14 - Esempio di utilizzo di un context con Change node.

Il tipo specifico di scope da utilizzare dipende da quello che si desidera fare con il context: se, a titolo di esempio, il valore di un determinato dato deve essere visibile solo all'interno di un function node, allora è sufficiente il Node Context. Se invece il dato deve essere visibile a più nodi, allora sarà necessario utilizzare un Flow Context oppure un Global Context. I context vengono salvati in RAM di default, quindi si comportano in maniera simile a delle variabili globali, ma tuttavia non sono persistenti ad un riavvio di Node-RED. Esiste comunque la possibilità di cambiare lo store type dei context per far sì che il loro valore venga salvato sul file system anziché in RAM. I context possono essere manipolati in diversi modi. Il modo più semplice è quello di utilizzare un change node, ad esempio la regola in Fig. 14 salva il contenuto presente in msg.payload sul context globale myContextData.

Esistono inoltre delle API che permettono di manipolare (leggere e scrivere) i context da un function node. Abbiamo 3 tipologie di accesso per ognuno dei 3 tipi di context, ed ognuna di queste può accedere a metodi di set e get:

- Context: node context;
- Flow: flow context;
- Global: global context.

Ad esempio, per leggere il valore myContextData con node

scope da un function node possiamo scrivere la seguente riga di codice all'interno di un nodo function:

```
var myData = context.get("myContextData");
```

Invece per scrivere il valore della nostra variabile locale, una volta che è stato aggiornato, sul context, possiamo utilizzare la seguente istruzione:

```
context.set("myContextData", myData);
```

I due esempi appena proposti utilizzano il node scope (context), ma sono validi anche per il flow e global context, basterebbe cambiare la keyword context appunto con flow o global.

ESEMPIO PRATICO

Passiamo adesso all'esempio pratico di questa puntata. Ciò che vogliamo fare è nuovamente far lampeggiare un LED, ma stavolta applicando alcuni dei nuovi concetti esposti in questa puntata.

Utilizzeremo ancora una volta un nodo inject per creare la temporizzazione ed il medesimo nodo rpi-gpio-out per controllare la linea di uscita, ma stavolta, al posto del nodo trigger, faremo uso di un function node, all'interno del quale scriveremo un po' di codice javascript per generare la logica necessaria al lampeggio ed utilizzeremo un context per immagazzinare il valore booleano corrispondente allo stato del LED tra un'attivazione del nodo function e la successiva. Iniziamo inserendo sul workspace i nodi necessari al nostro progetto, ossia:

- un nodo inject;
- un nodo function;
- un nodo rpi-gpio-out.

e collegandoli poi tra di loro logicamente come proposto nella Fig. 15.

I nodi inject e rpi-gpio-out devono essere configurati esattamente come nell'esempio pratico che vi abbiamo proposto nella puntata precedente del corso, così come rimane

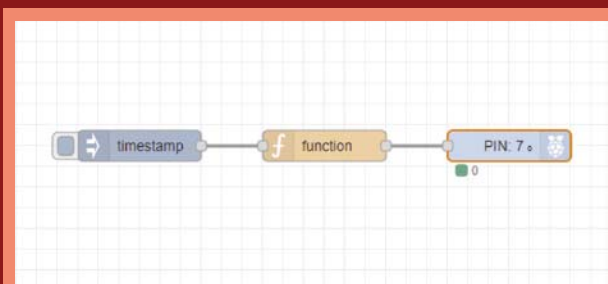


Fig. 15 - Flow per Led Blink con nodo function.

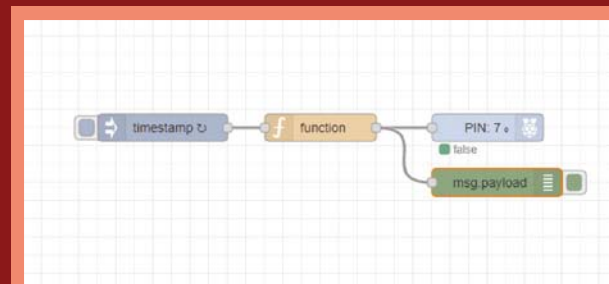
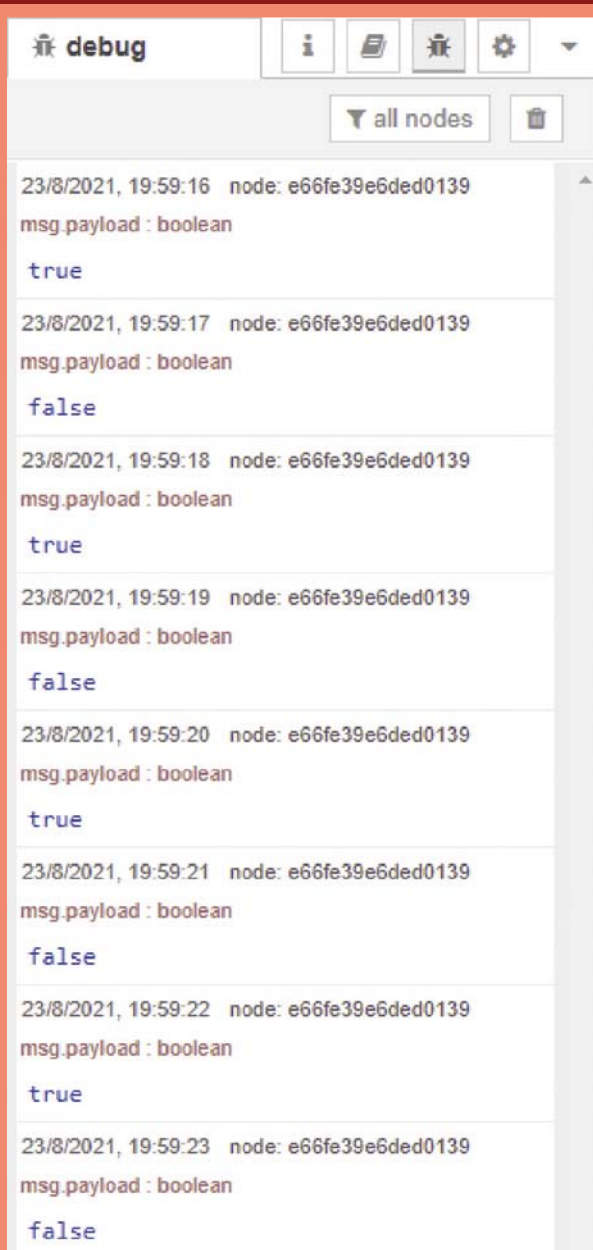


Fig. 16 - Inserimento del nodo Debug.

Listato 1 - Codice javascript per il LED blink

```
// Create the variable Led and load it with the context
var Led = context.get("LED");
// Invert the variable value
Led = !Led;
// Store back the inverted value into the context
context.set("LED", Led);
// Associate the payload with the variable value (Led status)
msg.payload = Led;

return msg;
```

**Fig. 17 - Debug Log.**

la stessa la configurazione hardware da utilizzare (il pin al quale è collegato il LED è il GPIO4).

Il nodo inject ha un'attivazione periodica e genera un messaggio ogni secondo; il contenuto di tale messaggio è irrilevante, in quanto non viene utilizzato dal nodo function, ma serve soltanto ad attivarlo.

Ad ogni attivazione, il nodo function deve generare in uscita un messaggio contenente sulla property payload lo stato del pin (e quindi del LED); questo viene realizzato tramite il codice javascript riportato nel **Listato 1**.

Come è possibile vedere da tale listato, viene creata una variabile Led, che viene caricata con il contenuto del context LED mediante l'istruzione `context.get(LED)`. Il context ha uno scope di tipo node, dal momento che non deve essere utilizzato da nessun altro nodo del flow al di fuori del nodo function stesso.

Di fatto questo context funziona come una variabile globale che immagazzina lo stato del LED tra una invocazione del nodo function e la successiva.

Nella riga successiva lo stato della variabile Led (appena caricato) viene invertito, allo scopo di generare il lampeggio desiderato.

A questo punto il nuovo stato viene salvato sul context mediante l'istruzione `context.set("LED", Led)`; fatto questo, non resta altro da fare che copiare il valore della variabile Led su `msg.payload` e "ritornare" il messaggio in uscita. Se lanciamo il flow in esecuzione mediante il pulsante deploy, utilizzando lo stesso setup dell'esempio pratico che vi abbiamo proposto nella prima scorsa puntata, vedremo il medesimo risultato, ossia il LED collegato al GPIO4 lampeggiare alla frequenza di 1Hz.

Se colleghiamo un nodo debug all'uscita del nodo function (**Fig. 16**) vedremo anche apparire un log che illustra il risultato pratico del codice che abbiamo inserito nel nodo, ossia una successione di messaggi con alternanza dello stato booleano tra true e false (che va a pilotare la linea digitale 4 della scheda Raspberry Pi), come visibile in **Fig. 17**.

CONCLUSIONI

In questa seconda puntata del corso abbiamo approfondito la nostra conoscenza di Node-RED, soffermandoci maggiormente sui suoi concetti fondamentali, come i flow, i nodi e i messaggi. Abbiamo inoltre analizzato alcuni nodi fondamentali installati per impostazione predefinita con l'ambiente (si tratta dei cosiddetti Core Nodes), visto come aggiungere nodi alla palette ed introdotto i context. Infine abbiamo concluso proponendo il solito esempio pratico, allo scopo di farvi "prendere la mano" con i concetti esposti e farveli fissare più facilmente nella mente.

Nella prossima puntata inizieremo a concentrare la nostra attenzione sull'interazione con il mondo esterno, quindi vedremo come fare per controllare alcune periferiche della scheda Raspberry Pi utilizzando le funzionalità offerte da Node-RED.



Oltre il 15% di efficienza per il FV in perovskite

Il successo nel raggiungimento della neutralità del carbonio richiederà un'espansione significativa nei luoghi in cui è possibile installare moduli fotovoltaici; i moduli più utilizzati sono realizzati con silicio cristallino e sono pesanti, e questo, oltre alla loro rigidità, limita dove possono essere installati. Il modulo fotovoltaico in perovskite a base di film polimerico è un'interessante alternativa perché è sottile, leggero e flessibile e può essere installato in luoghi in cui è difficile utilizzare moduli fotovoltaici in silicio, come tetti a bassa portanza e uffici finestre. I miglioramenti nell'efficienza dei moduli fotovoltaici in perovskite li hanno portati a un livello paragonabile a quello

dei moduli fotovoltaici in silicio.

L'ultimo viene di Toshiba e consiste in un nuovo metodo di rivestimento dello strato di perovskite. L'azienda aveva precedentemente sviluppato un processo di rivestimento in due fasi che prima applicava uno strato di inchiostro a piombo ioduro su un substrato, seguito da uno strato di inchiostro a ioduro di metammmonio ($\text{CH}_3\text{NH}_3\text{I}$), innescando una reazione che formava uno strato MAPbI₃. Ma questo approccio aveva un basso tasso di rivestimento e spesso lasciava sezioni inerti nello strato di perovskite. L'alternativa è un processo in un unico passaggio che applica direttamente l'inchiostro MAPbI₃, ma non è facile con-



trollare la cristallizzazione del MAPbI₃ e ottenere uno strato di perovskite uniforme su un'ampia area. Toshiba ha risolto questo problema con il rivestimento in un'unica

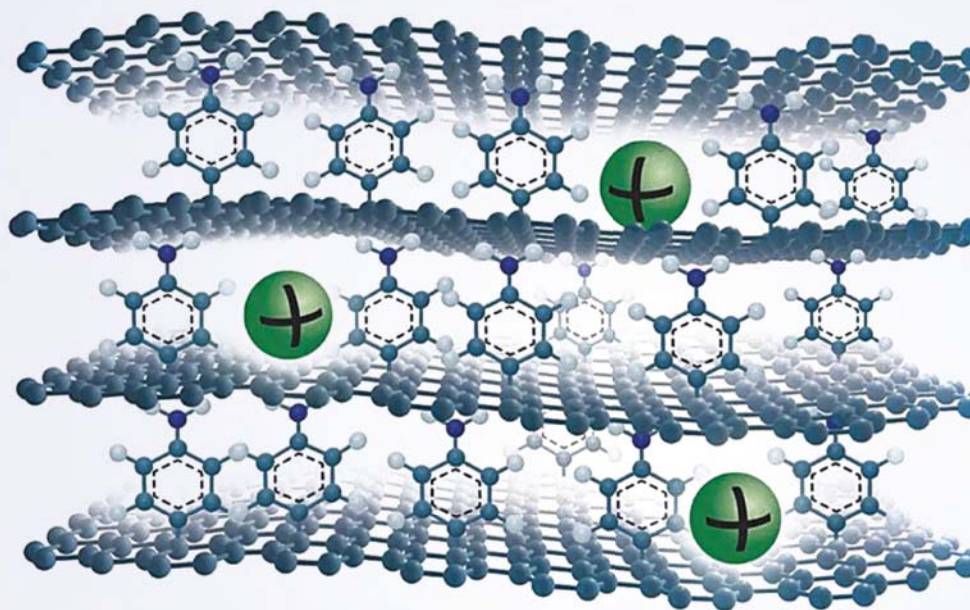
Alta capacità per le batterie a ioni di sodio

Dopo anni di attesa, le batterie agli ioni di sodio stanno iniziando a mantenere la loro promessa di diventare protagoniste dell'accumulo di energia elettrica. Finora, la loro commercializzazione è stata limitata agli impieghi su larga scala come l'immagazzinamento di energia della rete elettrica durante i periodi di surplus per la reimmissione durante i picchi di richiesta. La ragione di tale destinazione è nella densità di potenza, non competitiva rispetto a quella di batterie più diffuse nelle applicazioni come veicoli elettrici, laptop e in generale dispositivi mobile, come ad esempio quelle a ioni e a polimeri di litio. Infatti, con circa 285 Wh/kg, le batterie agli ioni di litio hanno il doppio della densità di energia del sodio, il che le rende più adatte per quelle applicazioni portatili.

Abbondante ed economico, e con proprietà chimiche simili a quelle del litio, il sodio è un sostituto promettente del litio nelle batterie di nuova generazione. La stabilità e la sicurezza delle batterie al sodio le rendono particolarmente promettenti per l'elettronica e le auto, dove le batterie agli ioni di litio si sono talvolta rivelate pericolose, ad esempio se surriscaldate.

Attualmente il problema principale delle batterie agli ioni di sodio è trovare un materiale adatto per l'anodo.

Affinché la batteria si carichi rapidamente e immagazzini molta energia, gli ioni devono scivolare facilmente dentro e fuori dal



materiale dell'anodo. Le batterie agli ioni di sodio utilizzano catodi fatti di ossidi metallici di sodio, mentre i loro anodi sono tipicamente a base di carbonio proprio come per quelle al litio (anche se è in fase di sviluppo una tecnica che impiega anodi e catodi fatti con il pigmento blu di Prussia). Alcuni sviluppatori di batterie al sodio utilizzano carbone attivo per l'anodo, che trattiene ioni sodio nei suoi pori, ma dev'essere di alta qualità, pertanto molto costoso e poco

reperibile. Ora i ricercatori propongono un nuovo tipo di elettrodo al grafene che potrebbe aumentare la capacità di stoccaggio delle batterie al sodio per competere con quelle al litio; il materiale può contenere quasi tanti ioni sodio in volume come un elettrodo di grafite convenzionale fa al litio ed apre la strada alla produzione di batterie al sodio compatte e a basso costo. La grafite, che è il materiale dell'anodo nelle batterie agli ioni di litio, è un'opzione a

fase che utilizza inchiostri migliorati, processi di essiccazione della pellicola e apparecchiature di produzione per formare uno strato uniforme di perovskite. Queste innovazioni dimezzano i passaggi per la deposizione dello strato e accelerano il processo a 6 metri al minuto su un modulo $5 \times 5 \text{ cm}^2$. Ciò permette moduli con efficienza record del 15,1%. Toshiba continua ad evolvere il processo e la sua ricerca ha l'obiettivo di aumentare il PCE (rapporto di conversione dell'energia solare in elettricità) al 20% e far scendere i costi di produzione a 0,14%/W.

www.toshiba.com

basso costo. Tuttavia, gli ioni sodio non si muovono in modo efficiente tra la pila di fogli di grafene che compongono la grafite. I ricercatori pensavano che ciò fosse dovuto al fatto che gli ioni di sodio sono più grandi degli ioni di litio, ma non è così, perché sembra che siano la chimica superficiale degli strati di grafene e la struttura elettronica che non possono ospitare ioni di sodio. I ricercatori hanno però creato un nuovo materiale simile alla grafite che supera questi problemi: applicano un singolo foglio di grafene su un foglio di rame e attaccano un singolo strato di molecole di benzene alla sua superficie superiore. Impilando molti di questi fogli formano una torta a strati di grafene tenuta separata dalle molecole di benzene. Lo strato di benzene aumenta la distanza tra gli strati per consentire agli ioni di sodio di entrare ed uscire facilmente. Creano anche difetti sulla superficie del grafene che fungono da siti di reazione attivi per assorbire gli ioni. Inoltre, il benzene ha gruppi chimici che si legano fortemente agli ioni sodio.

I calcoli dei ricercatori mostrano che la struttura garantisce densità di energia comparabili con quelle delle celle agli ioni di litio; la capacità della grafite per gli ioni sodio è in genere di circa 35 mAh per grammo, ma il nuovo materiale può contenere oltre 330 mAh/g.

news.cision.com



Volare con i jetpack come si pilota un drone

I jetpack possono sembrare divertenti, ma imparare a controllare un paio di motori a reazione legati alla schiena non è un'impresa facile. Ora una startup britannica vuole semplificare le cose sviluppando un jetpack con pilota automatico che lo renda controllabile come un quadricottero.

I Jetpack hanno fatto il salto dalla fantascienza al mondo reale fin dagli anni '60, ma da allora non hanno trovato molto impiego al di fuori delle apparizioni sulla celluloid. Negli ultimi anni, però, vivono un rinnovato interesse e i loro produttori vogliono dimostrare che la tecnologia non è più solo per stuntman e potrebbe estendersi a un vasto pubblico.

La società americana Jetpack Aviation insegnerà a chiunque a pilotare il suo jetpack JB-10, una tuta volante simile a quella del mitico super-eroe Iron Man, dotata di propulsione a reazione, sviluppata dalla start-up britannica Gravity Industries, già impiegata dai marine per salire a bordo delle navi e per portare rapidamente i medici in cima alle montagne.

Siccome i jetpack possono richiedere molto addestramento, l'esperto di animatronica di Hollywood Matt Denton e il comandante della Royal Navy Antony Quinn hanno fondato la Maverick Aviation, azienda che punta a svilupparne uno gestito dall'elettronica e manovrabile facilmente.

Il Maverick Jetpack è dotato di quattro turbine a getto in miniatura fissate a un telaio in alluminio, titanio e fibra di carbonio e viaggerà fino a 30 miglia

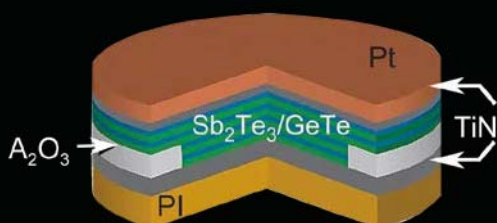
all'ora. Un software controlla automaticamente i motori per mantenere un volo stazionario stabile e convertire le istruzioni del pilota in movimenti precisi; ciò si rende necessario perché, diversamente dai motori elettrici dei droni, i turbogetti non rispondono istantaneamente alle variazioni di potenza e il loro controllo manuale può diventare davvero difficile. Per ovviare a questo problema, l'azienda ha aggiunto servomotori a ciascuna turbina che consente loro di muoversi indipendentemente per variare rapidamente la direzione della spinta (processo noto come vettore di spinta): spostando l'allineamento dei motori, il software di controllo del volo può mantenere il jetpack perfettamente posizionato utilizzando il feedback delle unità di misura inerziale, GPS, altimetri e sensori di distanza dal suolo. Semplici istruzioni direzionali del pilota possono anche essere tradotte automaticamente nelle necessarie modifiche di basso livello alle turbine. Il software è simile a un controller di volo per droni, ma l'ampiezza e la direzione della spinta devono essere gestite da circuiti di controllo separati a causa dei diversi tempi di reazione, ma devono comunque sincronizzarsi; il processo di controllo è anche critico perché diversamente da un drone, il jetpack trasporta una persona.

Maverick punta ora alla navigazione autonoma impostata su una mappa.

jetpackaviation.com

Qubit su substrato di silicio

Ricercatori giapponesi hanno sviluppato un qubit superconduttore interamente in nitrato, utilizzando la crescita epitassiale su un substrato di silicio che non usa l'alluminio come materiale conduttivo. Questo qubit utilizza nitrato di niobio (NbN) con una temperatura di transizione superconduttiva di 16 K



(-257°C) come materiale dell'elettrodo e nitrato di alluminio (AlN) per lo strato isolante della giunzione Josephson. Realizzando questo nuovo qubit di materiale su un substrato di silicio, sono stati ottenuti lunghi tempi di coerenza: un tempo di rilassamento energetico (T1) di 16 microsecondi e un tempo di rilassamento di fase (T2) di 22 microsecondi come valori medi. Questo è circa 32

volte il T1 e circa 44 volte il T2 dei qubit superconduttori di nitrato cresciuti su un substrato di ossido di magnesio convenzionale.

Le aspettative verso il calcolo quantistico crescono perché i circuiti a semiconduttore sono prossimi ai loro limiti, tuttavia, lo stato di sovrapposizione quantistica, indispensabile per il funzionamento di un computer quantistico, viene facilmente distrutto da vari disturbi (rumore). La sfida è come estendere il tempo di coerenza, che è la durata degli stati di sovrapposizione quantistica. Per superare questo problema si utilizza alluminio e film di ossido di alluminio come materiali qubit superconduttori, tuttavia sono fonte di rumore. Invece il nitrato di niobio cresciuto epitassialmente (NbN) riduce l'entità del problema.

Il qubit superconduttore siffatto è ancora nelle prime fasi di sviluppo e i ricercatori puntano a migliorare ulteriormente il tempo di coerenza ottimizzando il processo di progettazione e fabbricazione del qubit.

www.nict.go.jp/en

Farmaci anti-COVID creati dal Deep Learning

La minaccia del Covid-19 è insita principalmente nella sua capacità di mutare, il che rende difficile sviluppare terapie efficaci; il Deep Learning può aiutare, ma sinora è stato usato con malattie come il cancro, mentre con il COVID-19 è difficilmente attuabile a causa dei pochi dati disponibili per addestrare un algoritmo. Ora però gli scienziati del Computer Science and Artificial Intelligence Laboratory (CSAIL) del MIT e della Jameel Clinic for Machine Learning in Health hanno tentato un approccio basato su una rete neurale: siccome la sinergia tra farmaci spesso si verifica attraverso l'inibizione di bersagli biologici (come proteine o acidi nucleici), il modello apprende congiuntamente l'interazione farmaco-bersaglio e la sinergia farmaco-farmaco per estrarre nuove combinazioni. Il predittore del target del farmaco modella l'interazione tra un farmaco e un insieme di bersagli biologici noti correlati alla malat-

tia scelta. Il predittore dell'associazione target-malattia impara a comprendere l'attività antivirale di un farmaco, il che significa determinare la resa del virus nelle colture di tessuti infetti. Insieme, possono prevedere la sinergia di due farmaci. Sono state trovate due nuove combinazioni di farmaci utilizzando questo approccio: remdesivir (attualmente approvato dalla FDA per il trattamento di Covid-19) e reserpina, nonché remdesivir e IQ-15, che, nei test biologici, si sono dimostrati efficaci contro il virus. Il modello non è limitato a un solo ceppo di SARS-CoV-2, ma potrebbe anche essere utilizzato per la variante Delta o per altre che potrebbero sorgere. Per perfezionare la modellizzazione biologica, il team prevede di incorporare ulteriori informazioni come l'interazione proteina-proteina e le reti di regolazione genica.

www.mit.edu

Rendere sensibili gli oggetti stampati in 3D

I ricercatori del MIT hanno sviluppato un metodo per integrare le capacità di rilevamento in strutture stampabili in 3D composte da celle ripetitive, che consente di prototipare rapidamente dispositivi di input interattivi, nonché nell'elettronica indossabile.

Alcune varietà di oggetti stampati in 3D possono così "sentire", utilizzando una nuova tecnica che costruisce sensori stampando con metamateriali, sostanze costituite da griglie di cellule ripetute. Quando la forza viene applicata a un metamateriale flessibile, alcune delle cellule possono allungarsi o comprimersi e gli elettrodi incorporati in queste strutture possono rilevare l'entità e la direzione di questi cambiamenti di forma, nonché la rotazione e l'accelerazione. I ricercatori hanno prodotto oggetti fatti di plastica flessibile e filamenti elettricamente conduttivi con celle larghe 5 millimetri. Ogni cella ha due pareti opposte in filamento conduttivo e plastica non conduttiva, con le pareti conduttive che fungono da elettrodi. Le forze applicate agli oggetti modificano la distanza e l'area di sovrapposizione tra gli elettrodi, generando segnali elettrici che rivelano dettagli sulle forze applicate. I ricercatori suggeriscono che questi metamateriali potrebbero aiutare i progettisti a creare e modificare rapidamente dispositivi di input flessibili per un computer. Incorporare sensori in un oggetto potrebbe consentire un'analisi di come gli utenti interagiscono con esso: ad esempio, una sedia smart potrebbe rilevare una persona e accendere la luce o la TV.

www.mit.edu

HDMI
↓
VGA

€ 11,50

Cod. BOXHDMITOVGA

CONVERTITORE DA HDMI A VGA 1080P PLUG AND PLAY

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

CONVERTITORE DA AV A HDMI PLUG AND PLAY

AV
↓
HDMI

€ 14,00

Cod. BOXAVTOHDMI

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

Convertitori di segnale

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

CONVERTITORE DA HDMI AD AV PLUG AND PLAY

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

VGA
↓
HDMI

€ 11,50

Cod. BOXVGATOHDMI

CONVERTITORE DA VGA A HDMI PLUG AND PLAY

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

HDMI
↓
AV

€ 11,00

Cod. BOXHDMITOAV

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