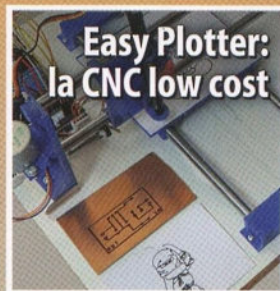


Elettronica In

www.elettronica.in.it

€ 6,00 - Anno XXIV - n. 226 - GIUGNO 2018

SAFETY SEA: salva l'uomo in mare



- Cebit, paladino della digital transformation
- Antennino: scheda programmabile wireless
- Fishino, rileva i fulmini e invia notifiche push
- Tactigon controlla VLC
- L'importanza del Bootloader
- Stazione meteo con ePaper



KIT RETROPIE e MINIRETROPIE con Raspberry Pi 3 !



KIT RETROPIE PER VIDEOGIOCHI ARCADE



Costruisci una postazione per videogiochi Arcade come quelle degli anni '80 e '90! Grazie a questo kit potrai emulare vari tipi di console: Nintendo, Game Boy, MAME, Sega Master System, Amiga, Commodore...

€ 115,00

Cod. **RETROPIEKIT_N**



Attenzione: il cabinet non è compreso, ma i disegni per realizzarlo sono scaricabili dalla scheda online del prodotto.

KIT MINI RETROPIE PER VIDEOGIOCHI ARCADE



Il kit permette di realizzare un mini cabinet per videogiochi Arcade degli anni '80 e '90. Consente di emulare vari tipi di console: Nintendo, Game Boy, MAME, Sega Master System, Amiga, Commodore...
Dimensioni contenute: 250x250x250 mm.

€ 184,00

Cod. **MINIRETROPIE**



Attenzione: il case mini cabinet realizzato con stampante 3D è acquistabile separatamente (9 colori disponibili). Cod. CASEMINICABINET • € 189,00.

Prezzi IVA inclusa

**FUTURA
ELETTRONICA®**

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

Caratteristiche tecniche
e vendita on-line su:

www.futurashop.it

Direttore responsabile:

Arsenio Spadoni
(Arsenio.Spadoni@elettronica.in.it)

Redazione:

Stefano Garavaglia, Paolo Gaspari,
Boris Landoni, Davide Scullino,
Alessandro Sottocornola
(redazione@elettronica.in.it)

Grafica:

Alessia Sfulcini, Filippo Gori
(Alessia.Sfulcini@elettronica.in.it)

Ufficio Pubblicità:

Monica Premoli (0331-752668)
(Monica.Premoli@elettronica.in.it)

Ufficio Abbonamenti:

Elisa Guarniero (0331-752668)
(Elisa.Guarniero@elettronica.in.it)

DIREZIONE, REDAZIONE, PUBBLICITÀ:

FUTURA GROUP srl - Divisione Editoriale
via Adige 11 - 21013 Gallarate (VA)
Telefono 0331-752668 Fax 0331-792287

Abbonamenti:

Annuo 10 numeri Euro 45,00
Esteri 10 numeri Euro 45,00 (digitale)
Le richieste di abbonamento vanno inviate a:

FUTURA GROUP srl
via Adige 11, 21013 Gallarate (VA)
tel. 0331-752668.

Distribuzione per l'Italia:

SO.D.I.P. Angelo Patuzzi S.p.A.
via Bettola 18 - 20092 Cinisello Balsamo (MI)
Telefono 02-660301 Fax 02-66030320

Stampa:

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

Elettronica In:

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 i collaboratori di Elettronica In, l'invio di articoli, materiale redazionale, programmi, traduzioni, ecc. implica da parte del Collaboratore l'accettazione dei compensi e delle condizioni stabilite dall'Editore (www.elettronica.in.it/ase.pdf). Manoscritti, disegni e foto non richiesti non verranno in alcun caso restituiti. L'utilizzo dei progetti e dei programmi pubblicati non comporta alcuna responsabilità da parte della Società Editrice.

Futura
Group
Edizioni

© 2018 FUTURA GROUP srl

MIND, da Milano un esempio per l'Italia

Si chiamerà MIND, Milano Innovation District, il nuovo Parco della scienza, del sapere e dell'innovazione progettato al posto dell'Expo di Milano. Una decisione, quella della destinazione dell'area Expo, ed un nome, MIND, presi in tempi ragionevolmente brevi, almeno per il nostro Paese.

"Mind è un nome evocativo che richiama Milano

e il suo sistema territoriale, ma anche il sapere, la conoscenza e l'innovazione

- spiega Giuseppe Bonomi, amministratore delegato di Areaexpo

- Il nome serve a creare un luogo e anche una nuova identità. L'eredità di Expo ha creato un patrimonio di credibilità ma l'area è in fase di trasformazione per diventare qualcos'altro".

L'area ospiterà Human Technopole, il Campus scientifico dell'Università Statale, un grande centro ospedaliero come il Galeazzi e gli uffici e i laboratori di una lunga serie di imprese private attive nei settori della ricerca scientifica, medica e farmaceutica, delle life sciences di cui proprio Milano è avanguardia europea.

Un nome ed una iniziativa in sintonia con quella che è la nuova Milano, una città che è radicalmente cambiata negli ultimi 10-15 anni, e non solo nello skyline. Una città che è diventata capitale mondiale del Design, della Moda e dell'Innovazione,

una città con una precisa visione del futuro che oltre la tecnologia ora mette in campo anche l'etica, con una capacità unica di legare competenze scientifiche e interessi umanistici. E questo al di là di chi ha governato la città in questo periodo. D'altronde Milano ha sempre avuto un ruolo importante nei momenti di cambiamento,

come testimoniano le opere del Bramante e di

Leonardo; anche nel dopoguerra, Milano era alla guida di un Paese alla riconquista del prestigio perso, guidata da uomini come Leopoldo Pirelli e Adriano Olivetti, che

a Milano era di casa. Erano

i tempi del grattacielo Pirelli (per anni il più alto edificio d'Europa) e dell'Olivetti Programma 101, il primo vero personal computer al mondo.

Una iniziativa importante, dunque, quella di MIND, non solo per Milano ma per tutto il Paese. Milano, sempre più capitale di un rinnovamento e di un cambiamento, può infatti diventare la locomotiva di una crescita economica che conduca finalmente il Paese fuori da quella crisi, non solo economica, che lo ha attanagliato negli ultimi decenni.



Arsenio Spadoni
(Arsenio.Spadoni@elettronica.in.it)

Sommario

Giugno 2018
n. 226



IN COPERTINA:

La tecnologia al servizio della sicurezza in mare, con un progetto che sfrutta la grande portata e la semplicità di utilizzo dei sistemi wireless LoRa (Long Range) di Semtech.

ELENCO INSERZIONISTI

- Arrow
- Futura Elettronica
- Microchip
- RM Elettronica

ARTICOLI

21 IN VETRINA CEBIT, SI FA PALADINO DELLA DIGITAL TRANSFORMATION. A PARTIRE DAL LOOK

Un CEBIT tutto nuovo ci aspetta a giugno: la più grande manifestazione dell'Information Technology cambia pelle e si trasforma in un Business Festival dell'innovazione.



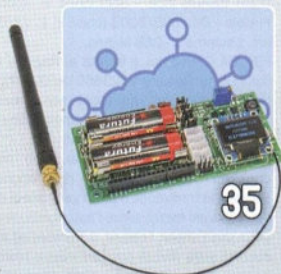
29 FISHINO FISHINO, RILEVA I FULMINI E INVIA NOTIFICHE PUSH

Una nuova libreria per le board Fishino consente di inviare con semplicità notifiche push multi-device utilizzando Pushetta. Proviamola in abbinamento al rilevatore di fulmini.



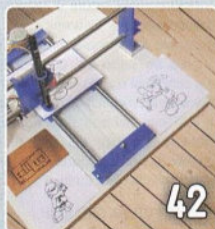
35 WIRELESS ANTENNINO: L'ARDUINO CON L'ANTENNA

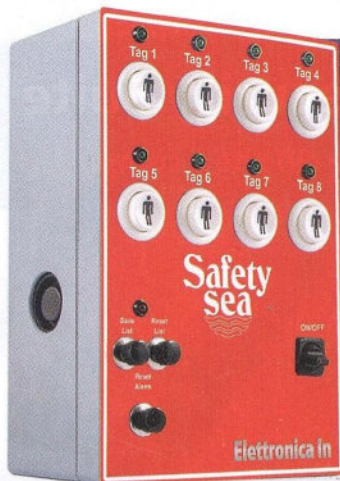
Scopriamo come caricare il bootloader nella nostra scheda per IoT, preparandola all'utilizzo. Seconda e ultima puntata.



42 GADGET EASY PLOTTER

Disegna e scrive a tratto mediante un pennarello a punta fine, su fogli di carta ed altro materiale, guidato da una meccanica a due assi autocostituibile con una spesa ridotta.





SICUREZZA

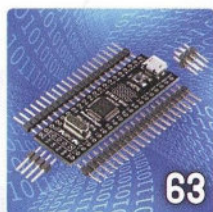
SAFETY SEA: SALVA L'UOMO IN MARE

Basato su unità trasmettente e ricevente LoRa consente di dare l'allarme se una persona cade in mare.

51

63 DIDATTICA L'IMPORTANZA DEL BOOTLOADER

Conoscere i microcontrollori, le sequenze di boot del firmware e la relativa programmazione.



63

77 APPLICAZIONI VLC & THE TACTIGON

Utilizziamo il dispositivo di prototipazione di applicazioni indossabili e rilevamento di gesturi tridimensionali, per controllare la riproduzione di una playlist VLC con dei movimenti delle mani.



77

La tiratura di questo numero è stata di 18.500 copie.



Mensile associato all'USPI,
Unione Stampa Periodica Italiana

82 HOME-PROJECT STAZIONE METEO CON ePAPER

Acquisiamo i parametri delle previsioni del tempo a tre giorni via WiFi da un sito Internet specializzato e li visualizziamo su un display e-Ink, realizzando un sistema a bassissimo consumo ideale per l'alimentazione a pile.



82

RUBRICHE

01 EDITORIALE

05 ELETTRONICA
INSIEME

12 APPUNTAMENTI
& EVENTI

17 COMPONENTI
& SISTEMI

109 SCIENZA
& TECNOLOGIA

123 EMBEDDED IOT

126 RINNOVABILI

93 RASPBERRY PI FRUITNANNY, OVVERO IL BIMBO A PORTATA DI RETE

Il custode elettronico che veglia sul riposo del nostro bimbo e sulle condizioni ambientali della sua cameretta.



93

113 DIDATTICA MIT APP INVENTOR

Proseguiamo lo studio e la conoscenza di App Inventor, il tool di sviluppo per applicazioni Android creato dal MIT e basato su un linguaggio di programmazione grafico. In questa terza puntata analizzeremo due interessanti famiglie di componenti, ossia Storage e Sensors.



113

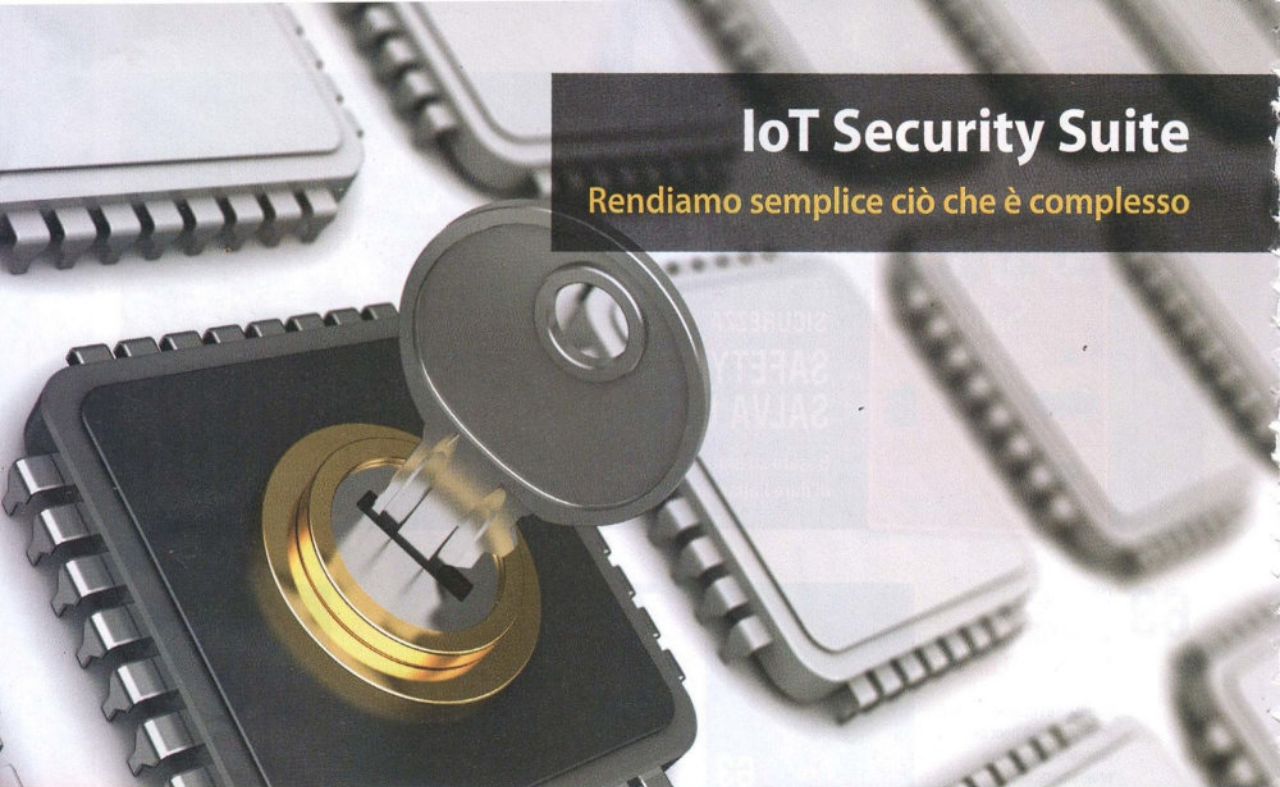


ELETTRONICA IN ~ Giugno 2018

3

IoT Security Suite

Rendiamo semplice ciò che è complesso



La IoT Security Suite per MPU SAMA5D2 consente un rapido e facile utilizzo delle sue avanzate caratteristiche di sicurezza, quali la tecnologia ARM® TrustZone® e crittografia hardware, senza lunghi tempi di apprendimento. La suite copre i requisiti di sicurezza per i produttori di dispositivi IoT in un unico package facile da usare. Supporta archiviazione, crittografia, decrittografia e lo scambio di chiavi tra dispositivi e applicazioni e, i suoi API, molto facili da usare, ti fanno risparmiare tempo.

Caratteristiche

- ▶ **Trusted Boot** – Avvio verificato da Root of Trust (RoT)
- ▶ **Firmware Protection** – Crittografia ed esecuzione di firmware autenticato
- ▶ **Trusted Device ID** – Certificato univoco del dispositivo legato al RoT
- ▶ **Secure Storage** – Conservazione sicura di chiavi, certificati e dati
- ▶ **Secure Communications** – Pairing dei dispositivi autenticati e comunicazioni cloud IoT
- ▶ **Secure Firmware Update** – Upgrade sicuro del firmware da remoto



SAMA5D2 Xplained
Ultra Evaluation Board
(ATSAMA5D2-XULT)

Per iniziare subito, scarica L'IoT Security Suite Evaluation Kit (è gratuito).

microchip
DIRECT

 **MICROCHIP**

www.microchip.com/SAMA5D2

Queste pagine sono dedicate alle richieste, ai suggerimenti ed alle segnalazioni dei lettori. Raccomandiamo, per quanto possibile, di proporre argomenti di interesse generale. Contattateci numerosi all'indirizzo: redazione@elettronica.in.it

IEEE 802.3bw 100BASE-T1

L'Ethernet per le auto

Ho letto del nuovo transceiver Ethernet di Texas Instruments DP83TC811S-Q1 che dispone dell'intera gamma di interfacce multimediali indipendenti, MII, RMII, RGMII e SGMII ecc. Posso utilizzare questo dispositivo anche sulla rete Ethernet domestica?

Luigi Colombo - Milano

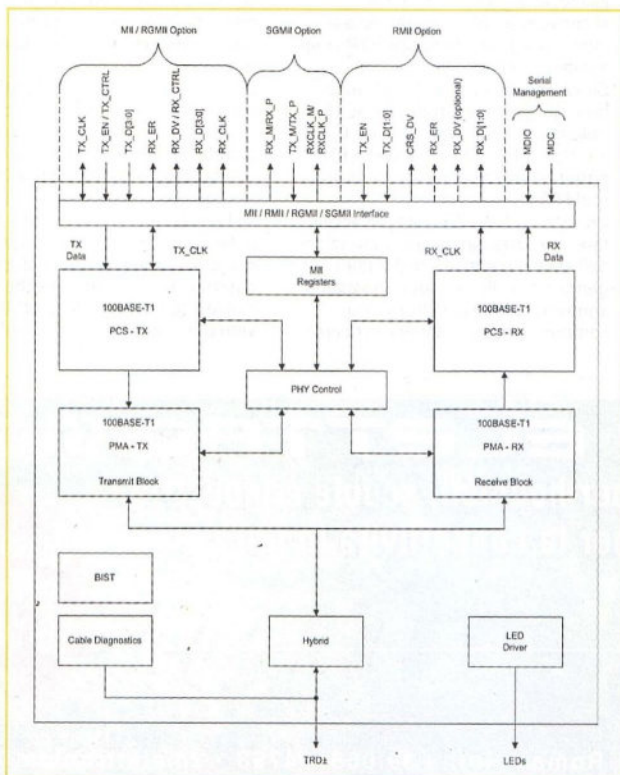
Absolutamente no, questo chip nasce infatti per il nuovo standard Ethernet per impiego automobi-

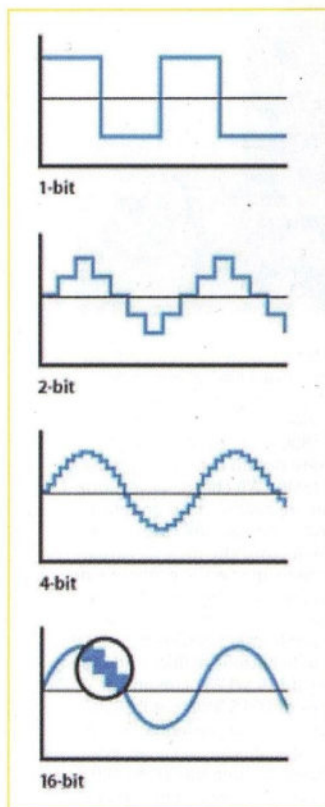
listico IEEE 802.3bw 100BASE-T1, studiato per venire incontro alle esigenze del mercato automobilistico e che utilizza un solo doppino intrecciato non schermato. Lo sviluppo di Ethernet 100BASE-T1 è stato pensato per soddisfare le nuove esigenze del settore automobilistico: i sistemi avanzati di assistenza alla guida richiedono una velocità di trasferimento dei dati più elevata, per sfruttare tutte le informazioni acquisite dalle numerose telecamere, oltre a una bassa latenza, per gestire efficacemente il controllo adattativo durante la marcia. Le reti Ethernet consentono di supportare comunicazioni dati veloci e a costi contenuti collegando a un bus comune i diversi

sottosistemi elettronici presenti nel veicolo. L'adozione della tecnologia Ethernet permette di semplificare il cablaggio, ridurre i consumi di carburante, i costi di produzione e il time-to-market. Lo standard 100Base-T1 utilizza le comunicazioni in modalità full duplex, in modo che sia possibile trasferire dati fino a 1 Gbit/s.

Questo standard, precedentemente noto come BroadR-Reach, è mirato ad aumentare il throughput dei dati e ad una riduzione del peso e del costo dei cablaggi all'interno dell'automobile rispetto agli standard Ethernet 10BASE-T e 100BASE-TX. Altra caratteristica importante del 100BASE-T1 è il basso livello di interferenze elettromagnetiche (EMI) ottenuto applicando il principio della sovrapposizione e specifici sistemi di encoding e scrambling. Naturalmente questo sistema consente anche la gestione dei dispositivi di infotainment di bordo. Se infatti osserviamo lo schema a blocchi dell'integrato DP83TC811S-Q1 possiamo notare che sono disponibili le interfacce MII, RMII, RGMII e SGMII che presentano le seguenti caratteristiche:

- MII (Media Independent Interface):
 - 4-bit-wide data interface
 - Receive and transmit controls
 - Receive and transmit clocks
 - RMII (Reduced Media Independent Interface):
 - 2-bit-wide data interface
 - Receive and transmit controls
 - Single clock reference
 - RGMII (Reduced Gigabit Media Independent Interface):
 - 4-bit-wide data interface
 - Receive and transmit controls
 - Receive and transmit clocks
 - SGMII (Serial Gigabit Media Independent Interface):
 - 2-pin low-voltage differential signaling (LVDS) receive path
 - 2-pin LVDS transmit path
- Nato sull'esperienza (spesso falli-





Bit ADC	Livelli	Tensione gradino (5V)
8-bit	256	19.53 mV
10-bit	1024	4.88 mV
12-bit	4096	1.22 mV
16-bit	65536	76.29 µV
18-bit	262144	19.07 µV
20-bit	1048576	4.76 µV
24-bit	16777216	0.298 µV

di tensione che l'ADC è in grado di apprezzare, è sufficiente applicare la seguente formula:

$$\text{Step Size} = V_{pp} / 2^n$$

Dove Step Size è il valore minimo che l'ADC è in grado di rilevare, V_{pp} la tensione massima del segnale da campionare e n il numero di bit del convertitore.

Prendiamo ad esempio un'onda sinusoidale con un intervallo di tensione di 5 volt. Se utilizziamo un convertitore ADC con risoluzione di 12 bit, il minimo valore di tensione che riusciremo ad apprezzare sarà di:

$$\text{Step Size} = 5V / 4096 \text{ ovvero } 1,22 \text{ mV.}$$

La Tabella consente di conoscere immediatamente la risoluzione di un ADC che campiona un segnale a 5V in funzione dei bit dell'ADC stesso.

C'è onda e onda

Ho visto che tra gli inverter a cosiddetta "onda quadra modificata" d'uscita e quelli ad "onda sinusoidale pura" esiste una differenza di prezzo consistente, anche di 2/3 volte. Come si giustifica questa differenza? Il mio inverter "cinese" da pochi euro continua da anni ad alimentare delle lampade ad incandescenza senza problemi...

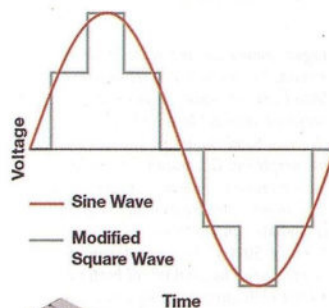
Giovanni Sini - Napoli

L'inverter è un dispositivo elettronico che trasforma una tensione continua (solitamente 12 o 24 volt) in una tensione alternata a 220 volt, simile a quella che l'ENEL porta in tutte le case e che utilizziamo normalmente per alimentare gli elettrodomestici, le luci di casa e i vari dispositivi elettronici (TV, PC, impianto stereo, ecc.). L'inverter consente di avere a disposizione una tensione a 220 volt in tutte quelle situazioni dove non è possibile allacciarsi alla rete elettrica, ad esempio, in barca, in auto, in località isolate, in campeggio, ecc. Gli inverter possono servire anche a casa, nell'ipotesi in cui venga a mancare la tensione di rete, e purché sia disponibile una batteria di potenza sufficiente. Recentemente sono stati introdotti (vedi Powerwall di Tesla) dei sistemi a batteria con inverter incorporato che, collegati ad un array di pannelli fotovoltaici, consentono di rendere autonoma la propria abitazione dalla rete elettrica pubblica, garantendo una sorgente a 220 volt per alimentare tutte le utenze. Anche in questo caso, l'inverter provvede a trasformare la tensione continua presente ai capi della batteria di accumulo in una tensione alternata di potenza, sufficiente ad alimentare elettrodomestici e lampade.

Due sono le caratteristiche più importanti di un inverter, da cui

dipende ovviamente il prezzo: la potenza nominale e la tecnologia utilizzata, ovvero la forma d'onda che il dispositivo è in grado di erogare all'uscita. Per quanto riguarda la potenza, è evidente che l'inverter deve avere una potenza nominale pari o superiore a quella complessiva del carico da alimentare. Per quanto riguarda questo aspetto, bisogna considerare che molte apparecchiature assorbono all'accensione - per un breve istante - una potenza che può superare anche di 2/3 volte quella nominale. È importante perciò, nella scelta dell'inverter, che anche quest'ultimo disponga di una potenza di picco adeguata alla potenza istantanea richiesta.

E veniamo alla forma d'onda. Negli inverter più economici, lo stadio elevatore dell'inverter è seguito da un modulatore ad onda quadra; ne consegue che in uscita la forma d'onda è più simile ad un'onda quadra piuttosto che ad una sinusoidale (vedi immagine). Se l'onda presenta dei fronti di salita o discesa molto ripidi significa che esistono delle armoniche di frequenza tanto maggiore quanto più ripido è il fronte di salita o discesa e di ampiezza proporzionale allo scalino in tensione dell'onda quadra di



uscita. In altre parole, significa che questo genere di inverter non sono adatti ad alimentare elettrodomestici "sensibili" come i televisori o gli impianti stereo. In questi casi, infatti, sentiremo del ronzio di fondo o noteremo dello sfarfallio nello schermo. Anche i motori non sono fatti per funzionare con una tensione simile, alimentarli con un'onda quadra li danneggia, per cui se dobbiamo alimentare un qualcosa che ha un motore (trapano, frigorifero, ecc.) è consigliabile utilizzare un inverter "pure sine wave".

Forse solo le lampade ad incandescenza (non quelle a LED che utilizzano un sensibile circuito di controllo) e le resistenze elettriche di forni e riscaldatori possono essere alimentate senza problemi con gli inverter del primo tipo. Per quanto riguarda la differenza di prezzo, questa è (in parte) giustificata con la maggior complessità di un circuito di potenza che deve generare l'onda sinusoidale.

Se il truck di Elon Musk non convince

Leggo sempre con interesse la vostra rivista. Sul numero di Febbraio ho letto l'articolo sulla Tesla e sulla presentazione del loro camion elettrico Semi che mi ha lasciato un po' perplesso. Gli annunci di Tesla mi sembrano perlomeno fantasiosi, in quanto per spostare una massa di 36 tonnellate (un moderno TIR) le motrici hanno 4-500CV. Se, come dicono, il camion Semi ha 200kWh di batterie e 800Km di autonomia significa che ha un motore elettrico su per giù da 20kW cosa assolutamente impossibile per una motrice. Oppure non ha 200kWh di batterie ma ne ha 2000, quindi ci starebbe come autonomia e potenza con 4 motori per un totale di 200kW, ma non ci starebbe con il costo, visto che 2000kWh di batterie, anche se fossero solo 200€/kW, costano 400000€ e porterebbero questo camion completamente fuori mercato come costo. Per quanto riguarda la supersportiva non vedo problemi nell'accelerazione, ma nella velocità massima, visto che per fare 400Km/h dovrà avere dei motori di almeno 4-500kW, ma per fare 1000Km di autonomia dovrà avere un bel fardello di batteria, che renderà la macchina pesantissima, e quindi in difficoltà nel



fare la velocità massima. Sono corretti i miei ragionamenti?

Fabio Fioravanzo - Vicenza

Effettivamente i dati snocciolati da Elon Musk durante la presentazione del nuovo camion Semi Truck e della Roadster 2 vanno presi per quelli che sono: l'annuncio di una nuova avventura e di un prodotto che quando arriverà sul mercato (dopo qualche anno) non risponderà esattamente a quanto annunciato (per prezzo, prestazioni o altro). È già successo con la prima Roadster, con la Model X e con la Model 3, anche se c'è da osservare che alcune volte le prestazioni sono risultate superiori a quanto dichiarato in sede di presentazione. In tutti i casi, tra l'annuncio e la reale disponibilità del prodotto passano sempre 2/3 anni, che nel campo delle tecnologie di cui stiamo parlando possono garantire sviluppi sorprendenti, sia dal punto di vista tecnico che per quanto riguarda i costi. E, ancora una volta (come già successo per le Tesla elettriche, Space X, Hyperloop, ecc.), va riconosciuto a Elon Musk di essere riuscito a guardare (non a caso dicono di lui che sia un visionario) da un punto di vista completamente diverso un sistema di trasporto ritenuto dai più ormai consolidato. In fondo non era difficile immaginare un camion elettrico: in questo caso, infatti, al contrario delle vetture, non ci sono problemi di spazio o di peso per le batterie e il motore elettrico è enormemente più semplice del motore a gasolio, sia per quanto riguarda la manutenzione che il rendimento. Tornando "a bomba" alle caratteristiche tecniche, il nuovo TIR utilizzerà quattro motori elettrici a magneti permanenti (gli stessi della Model 3) da 271 CV cadauno per complessivi 1.084 CV. La potenza, dunque risulta più che adeguata ai gravosi compiti del veicolo. Com-

pletivamente questi motori consumano circa 2 kWh al miglio per cui, effettivamente nella versione long-range con autonomia di 500 miglia (800 km circa) il veicolo dovrà essere dotato di un pacco batterie da 1.000 kWh che, nella versione con minore autonomia scenderà a 600 kWh. Tanti, in ogni caso, sia in termini di costo che di peso. Attualmente però, grazie alle economie di scala consentite dalla GigaFactory e anche grazie all'ottimizzazione del nuovo formato delle celle, il costo delle batterie al litio si aggira attorno ai 125 \$/kWh, con un target price di 100 \$/kWh per il 2020 quando verranno consegnati i primi camion. Il costo del pacco batterie si attesterà dunque tra i 60 e i 100 mila dollari, in linea con quello che è il prezzo finale del veicolo. Per quanto riguarda il peso, attualmente siamo arrivati a valori di 150-180 Wh/kg con un target per il 2020 di 200 Wh/kg. Ciò significa che il pacco batterie del nuovo Semi peserà tra 3 e 5 tonnellate, valore anche in questo caso compatibile con un veicolo con una capacità di carico di 36 tonnellate. Infine, dobbiamo tener conto che il truck Semi di Tesla Motor sarà il capostipite (un po' come lo è stata la Roadster) di una categoria di veicoli che sicuramente in pochi anni sarà soggetta ad una rapidissima evoluzione tecnologica. Naturalmente per la Roadster 2 valgono le stesse considerazioni appena fatte per il nuovo TIR elettrico. A nostro avviso la vera incognita per Elon Musk è la sua Tesla Motor è un'altra: quella di possedere la capacità di trasformare un sogno visionario in una realtà industriale, per molti versi sicuramente più complessa. Anche perché c'è chi, pur non avendo le stesse intuizioni di Elon Musk, ha una consolidata tradizione industriale in questo settore e tanti miliardi da investire (leggi: tedeschi e cinesi).

La Carta di Smith

Un mio amico radioamatore utilizza spesso quello che lui chiama la Carta di Smith per controllare e mettere a punto la sua stazione HF. Di cosa si tratta esattamente?

Roberto Merlo - Torino

La carta di Smith è uno strumento molto usato per risolvere tutti i problemi di adattamento delle linee a radiofrequenza. Si tratta di un diagramma circolare, costituito da due serie di circonferenze che si intersecano e che consentono di determinare l'impedenza, o più spesso, l'ammettenza, nelle sue due componenti, reale ed immaginaria, della linea in ogni suo punto. L'uso della carta di Smith è cresciuto in modo costante negli anni ed essa è ancora oggi largamente utilizzata, non solo come aiuto nella risoluzione di alcuni problemi, ma anche per mostrare graficamente come alcuni parametri RF cambiano al variare della frequenza. La Carta di Smith relativa a una serie di dati numerici risulta, infatti, in molti campi, di più immediata comprensione

rispetto a una tabella contenente le stesse informazioni. Dal punto di vista pratico, la carta di Smith consente di scegliere il punto adatto lungo la linea, per inserire lo stub in grado di compensare la parte reattiva, indesiderata, del carico e permettere al generatore di vederne solo la parte resistiva e non generare, di conseguenza, onde riflesse indesiderate. Nella sua forma più comune è un diagramma bi-dimensionale che permette il calcolo grafico approssimato di una funzione. Si basa su una rappresentazione dei dati in un opportuno sistema di coordinate non

cartesiane. Come il regolo calcolatore, è uno strumento grafico di calcolo analogico.

La carta di Smith viene tracciata sul piano complesso del coefficiente di riflessione, e i valori rappresentati sono tipicamente impedenze o ammettenze normalizzate, o in alcuni casi entrambi, che sono rappresentate con colori diversi per permettere di distinguerle. Con semplici costruzioni grafiche è possibile convertire ammettenze o impedenze normalizzate nel corrispondente coefficiente di riflessione.

La parte della Carta di Smith utilizzata più frequentemente è quella posta sulla circonferenza di raggio o quella collocata nella zona centrale. L'impiego della Carta di Smith per il calcolo e la verifica delle impedenze di una linea di trasmissione non è semplicissima comporta spesso disagi di tipo grafico, soprattutto in situazioni complesse come la presenza di più impedenze di carico e/o componenti discreti all'interno di una linea.

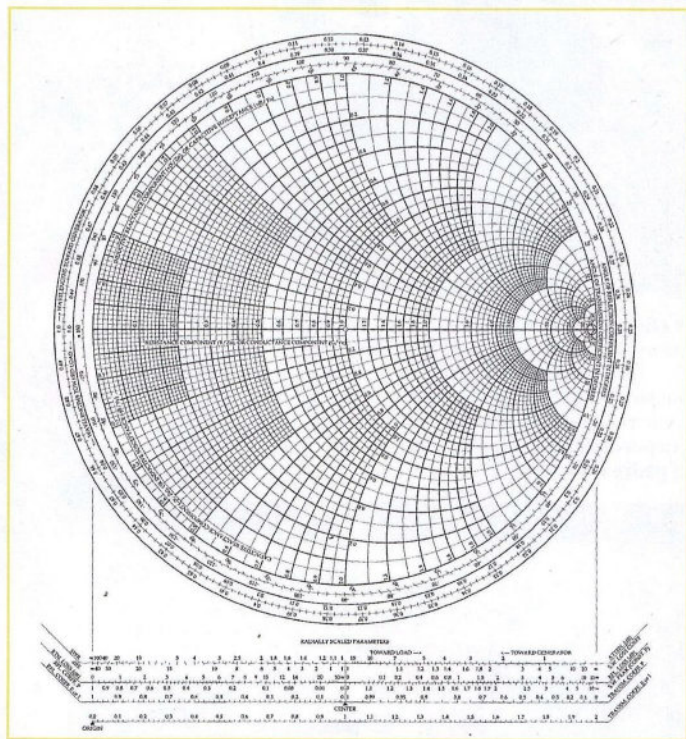
Per questo motivo esistono delle soluzioni software e web-based che ne consentono un utilizzo molto più semplice.

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.elettronica.in.it/webforum



SU Elettronica In TROVI SEMPRE L'ISPIRAZIONE PER CREAZIONI UNICHE!

Mese dopo mese
realizza i progetti
descritti sulla rivista,
rimani aggiornato
con i nostri Corsi,
approfondisci
la conoscenza delle
tecniche e dei
componenti più avanzati
con i nostri tutorial
e le nostre news.

COME ABBONARSI

① on-line

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

② mail

scrivendo a abbonati@elettronica.in.it
con i dati riportati nel coupon a lato.

③ posta

compilando il modulo di abbonamento
riportato a lato e inviandolo al seguente
indirizzo:

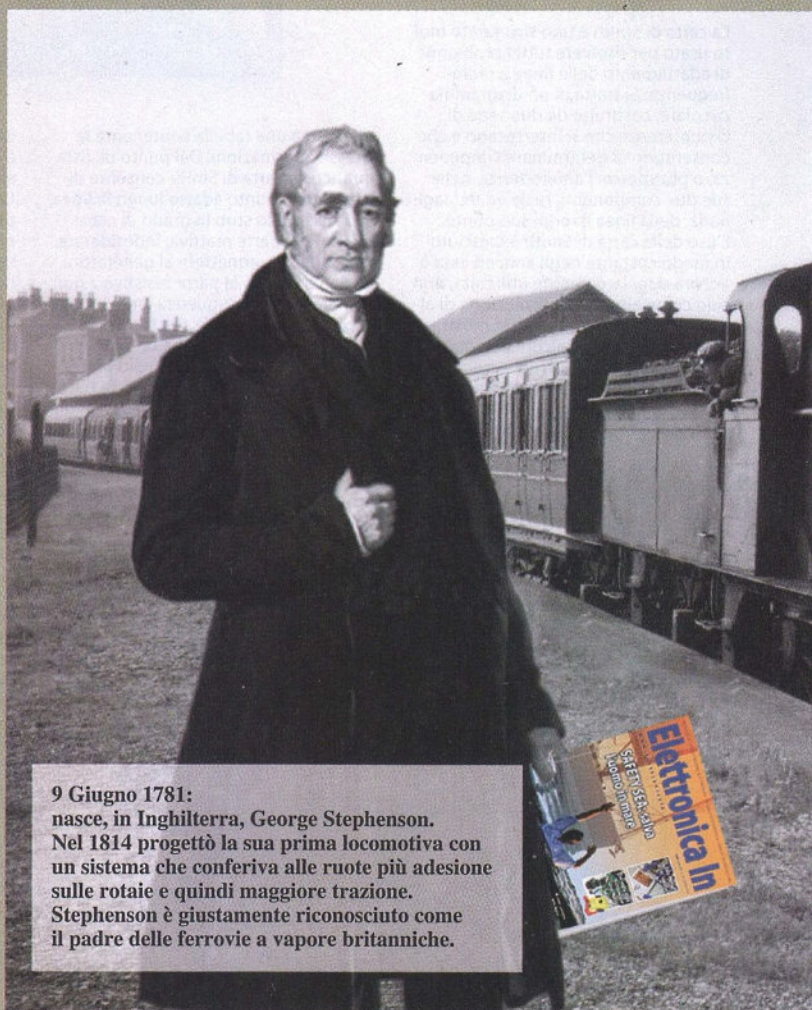
FUTURA GROUP srl
Via Adige 11
21013 Gallarate (VA)

④ telefono

telefonando a
+39-0331-752668

⑤ fax

compilando il modulo di
abbonamento riportato
a lato e inviandolo al
seguente numero:
+39-0331-792287



9 Giugno 1781:
nasce, in Inghilterra, George Stephenson.
Nel 1814 progettò la sua prima locomotiva con
un sistema che conferiva alle ruote più aderenza
sulle rotaie e quindi maggiore trazione.
Stephenson è giustamente riconosciuto come
il padre delle ferrovie a vapore britanniche.

PER I RINNOVI ATTENDERE NOSTRO AVVISO



UNO STRUMENTO INDISPENSABILE PER I TUOI STUDI, IL TUO LAVORO O LA TUA PASSIONE CHE PUOI RICEVERE COMODAMENTE A CASA TUA OGNI MESE SOTTOSCRIVENDO UN...

ABBONAMENTO ANNUALE



10 NUMERI
a soli euro
45,00
anziché euro 60,00



IN OMAGGIO PER TE:

10% di sconto SUI TUOI ACQUISTI
FUTURA ELETTRONICA®



1 VOLUME A SCELTA

effettuati con la **DISCOUNT CARD** presso il punto vendita di Gallarate (VA) o nel nostro store on-line www.futurashop.it

della collana "L'Elettronica per tutti".

Sì

desidero abbonarmi per un anno alla rivista *Elettronica In*. Riceverò i seguenti omaggi:



DISCOUNT CARD "Futura Elettronica"



UNO, A SCELTA, TRA QUESTI VOLUMI della collana "L'Elettronica per tutti".



Entra nel mondo di Raspberry Pi 3



Fishino - Arduino e l'Internet delle Cose in un'unica innovativa scheda



Randa - Raspberry Pi + Arduino



ARDUINO UNO - Programmazione avanzata e Librerie di sistema



Tecnologia e apparati GPS



Impara l'elettronica sperimentando



Primi passi in elettronica



Arduino e le tecniche



di programmazione dei microcontrollori ATME



Raspberry Pi - il mio primo Linux Embedded



Alla scoperta di Arduino YUN



Programmiamo con i PIC



L'ABC di Arduino



Stampiamo in 3D

Resto in attesa del primo numero e degli omaggi.

Cognome:

Nome:

Via:

N°:

CAP:

Città:

Provincia:

Telefono:

e-mail:

Data:

Firma:

Resto in attesa di vostre disposizioni per il pagamento. Ai sensi dell'art. 13 del D.Lgs. 196/2003, in tema di protezione dei dati personali, FUTURA GROUP srl informa che i dati liberamente forniti attraverso la compilazione del presente modulo saranno trattati e conservati in conformità a tutte le normative vigenti. I dati suddetti saranno trattati su supporto cartaceo e informatico, mediante sistemi di protezione atti alla tutela della riservatezza, per finalità connesse all'esecuzione dei rapporti contrattuali instaurati. Inviare in busta chiusa a:



FUTURA GROUP srl

Via Adige 11 • 21013 Gallarate (VA)

Tel. +39-0331-752668 • Fax: +39-0331-792287

AUTOMOTIVE 2018

La terza edizione del Convegno AUTOMOTIVE 2018 si terrà dal 9 all'11 luglio 2018 presso il Politecnico di Milano (Bovisa Campus).

AUTOMOTIVE 2018 riunirà per tre giorni in un evento internazionale la comunità automobilistica e dell'Information and Communication Technology per presentare e discutere i risultati più recenti della ricerca scientifica e tecnologica in questo settore, con particolare attenzione alle applicazioni e alle nuove tendenze. La Conferenza prenderà in considerazione tutti gli aspetti relativi alla tecnologia dei veicoli elettrici, delle autovetture autonome connesse e delle infrastrutture relative.

Il Comitato direttivo e il Comitato tecnico della Conferenza comprendono esperti del mondo Accademico e delle Associazioni nonché rappresentanti dell'industria automobilistica e autorità pubbliche.

Il programma di AUTOMOTIVE 2018 sarà strutturato in sessioni scientifiche, speech e tavole rotonde, riguardanti gli aspetti tecnici, le tendenze di mercato e il quadro normativo. Verranno anche affrontati aspetti legali, etici e sociali relativi alle nuove opportunità offerte dalla tecnologia in ambito automobilistico.

AUTOMOTIVE 2018 si rivolge ad un pubblico accademico e industriale, professionisti attivi nel settore automobilistico, tra cui designer, progettisti, e produttori, nonché funzionari della pubblica amministrazione, analisti e investitori interessati a questo settore in grande sviluppo e di elevato impatto sociale.

<https://convegni.aeit.it/automotive/>



A Modena il 27 e 28 giugno per Fortronic 2018

Promosso da Assodel (*Associazione Distretti Elettronica Italia*), si terrà quest'anno il 27 e 28 giugno a Modena Fiere, Fortronic, da oltre 20 anni incontro privilegiato sulle nuove tecnologie articolato su tre eventi:

1. lo **Strategic Innovation Summit** (27 giugno), un momento di ispirazione per i manager italiani che desiderano fare dell'innovazione tecnologica una strategia di successo;
2. l'annuale **Gala Assodel** (27 giugno), durante il quale saranno assegnati gli Assodel Award all'industria;
3. la quindicesima edizione del **Power Forum** (28 giugno), l'unico evento italiano per i tecnici e i professionisti dell'elettronica di potenza.

27 Giugno: un summit sull'innovazione tecnologica

Nella prima giornata, lo Strategic Innovation Summit analizzerà i trend dell'innovazione e le proposte del territorio, tra casi di successo e progetti "di ispirazione". L'evento si rivolge a tutti i manager italiani che investono in tecnologia e si focalizzerà sulle tematiche legate alla mobilità del futuro. Da sempre, infatti, il settore automotive è precursore nell'utilizzo delle nuove tecnologie che nel tempo vengono poi adottate dall'industria. Il convegno avrà inizio alle 14.30 con una serie di presentazioni di start-up (come Italian Volt, eccellenza nostrana nella produzione di moto elettriche, e VisLab, spin-off dell'Università di Parma pioniere nei sistemi per auto a guida autonoma) ed enti del territorio (come ASTER, la società consortile dell'Emilia-Romagna per l'innovazione e il trasferimento tecnologico, MUNER – Motorvehicle University e CRIT – Centro Ricerche e Innovazioni Tecnologiche).

27 Giugno: networking dinner e premio all'industria

Il Summit terminerà con un aperitivo di networking e l'annuale **Gala Assodel** a cui partecipano oltre 300 manager del settore e che si terrà per l'occasione nella splendida cornice di Villa Cesi a Modena.

Durante la serata, saranno assegnati gli **Assodel Award** ai manufacturer dell'industria elettronica più innovativi e propositivi nei confronti della supply chain. Tra la novità della sedicesima edizione, anche uno speciale **Industry Award** alle aziende clienti che si sono maggiormente distinte in tema di innovazione tecnologica.

28 Giugno: L'evento tecnologico sull'elettronica di potenza

La seconda giornata sarà completamente dedicata al **Fortronic Power**, l'unico evento italiano sull'elettronica di potenza. Con oltre 70 aziende presenti su un'area espositiva di 3000 mq saranno presentate le soluzioni e i prodotti più innovativi del settore. In parallelo si terranno due sessioni plenarie sulle ultime novità tecnologiche, a cui parteciperanno aziende del calibro di Infineon, Fuji Electric, USCi, ST Microelectronics, Rohm Semiconductor e Tektronix.

Contemporaneamente, sarà possibile assistere ai diversi workshop tenuti dai top player del settore.

<https://fortronic.it/>



**POWER
FORTRONIC**

Ritorna il 20 giugno GET CONNECTED by Arrow



Il mercato dell'IoT è in continua evoluzione e Arrow Electronics è attenta a questi cambiamenti, pronta a fornire nuove soluzioni chiavi in mano.

Per fare conoscere i nuovi scenari nonché prodotti e soluzioni in grado di rispondere alle nuove esigenze, anche quest'anno Arrow organizza GET CONNECTED, l'evento di una giornata dedicata al mondo dell'IoT che offre la possibilità di toccare con mano l'intero ecosistema,

dal sensore al cloud, "sensor to sunset".

La giornata prevede al mattino una sessione Plenaria dove si succederanno speech di grande attualità sui temi del momento come Industry 4.0, Sicurezza in ambito IoT e Vertical Market, con la partecipazione di Microsoft, il Politecnico di Milano, IoT Block Chain, Arduino, Vodafone e Digital Transformation.

Nella seconda parte della giornata ci sarà la possibilità di partecipare a show case con applicazioni e proof of concept e speech tematici focalizzati su sensoristica, wireless, cloud, sicurezza e soluzioni embedded.

Alcuni interventi saranno dedicati ai Management aziendali, addetti degli Uffici Acquisti e del Marketing durante i quali verranno presentati i nuovi servizi Arrow come Value Recovery, Indiegogo, My Arrow e altri ancora.

Saranno presenti circa 40 fornitori e 20 tra incubatori e system integrator. Oltre 900mq di show case consentiranno di incontrare i maggiori player del mercato suddivisi per tecnologia (Connectivity, Sensors, Cloud&Security, Embedded-SOM&LCD) e di interagire con applicazioni IoT connesse al cloud. Sarà anche possibile fare un'esperienza "from sensor to sunset", dal sensore al cloud.

Quest'anno GET CONNECTED si terrà all'Hotel NH Milano Congress Centre di Assago (Milanofiori) il 20 giugno 2018.

www.arrow.com

EVENTI & APPUNTAMENTI

Keysight HOTSPOTS, a Roma il 14 giugno 2018

Da più di 75 anni Keysight Technologies aiuta scienziati, ricercatori e progettisti ad affrontare ogni sfida nel campo della misura per ottenere risultati precisi e certi. La storia di Keysight affonda le sue radici nelle attività nel campo della misura svolte da Hewlett-Packard sin dalla sua fondazione nel 1939, che furono poi scorporate nel gruppo EMG di Agilent, la quale, successivamente, diventò un'azienda completamente indipendente. Attraverso tutti questi cambiamenti, Keysight ha aiutato le aziende a sviluppare prodotti elettronici tecnologicamente avanzati e a introdurli sul mercato in tempi più brevi e ad un costo più basso.

Con l'iniziativa HOTSPOTS, Keysight promuove una serie di seminari tecnici che offrono a quanti operano professionalmente nel settore delle comunicazioni wireless, dell'aerospazio e difesa, nel settore automobilistico, dell'energia, dei semiconduttori e in svariate diverse realtà, una piattaforma per incontrarsi e discutere delle sfide che il futuro propone.

Il programma dei seminari HOTSPOTS prevede i seguenti argomenti:

- Fondamenti di Misure RF & Micro-onde
- Fondamenti di Misure ad Onde Millimetriche - NOVITA'
- Fondamenti di Misure Digitali
- Fondamenti di Misura sull'Integrità dei Segnali - NOVITA'
- Misure Elettroniche nella Conversione di Potenza
- Fondamenti di Misure su Materiali & Dispositivi
- Fondamenti di Misure su Dispositivi IoT
- La Taratura: come scegliere la soluzione più idonea

Il prossimo appuntamento è a Roma il 14 giugno 2018 con un seminario su "Fondamenti di Misure su Materiali & Dispositivi" con il seguente programma:

- Benvenuto e Introduzione: Keysight nella Misura & Collaudo
- Sfide e Soluzioni in applicazioni di test per l'Ingegneria e la Scienza dei Materiali
- Soluzioni di Test Avanzate per la Misura di Impedenza
- Caratterizzazione di Dispositivi & Materiali dalla continua alle onde millimetriche & THz
- Caratterizzazione Elettrica di dispositivi in GaN & SiC con lo strumento Keysight B1505A

Comprendere le proprietà dei materiali (sia naturali che artificiali) è importante per diversi motivi. Questo seminario offre una panoramica delle principali tecnologie emergenti e di come Keysight offra delle soluzioni capaci di superare anche le sfide della misura più difficili.

www.keysight.com

KEYSIGHT HOTSPOTS

Where Technologies and Experts Meet



More experts. More knowledge. More in-depth.

19-22 GIUGNO 2018

Automatica 2018

Monaco di Baviera • Germania



"Optimize your Production" è questo il motto dell'edizione 2018 di Automatica, il Salone Internazionale di Automazione Intelligente e Robotica organizzato da Messe München dal 19 al 22 giugno 2018 a Monaco di Baviera.

La missione della più grande rassegna mondiale del settore è chiara: offrire i mezzi e le procedure per realizzare prodotti di qualità migliore con più efficienza, garantire accesso a innovazione, conoscenze e sviluppi di grande rilevanza commerciale, orientare alla trasformazione digitale. Automatica offre una visione completa sui seguenti settori:

- robotica industriale e di servizio
- impianti di assemblaggio
- sistemi di visione industriali
- componenti

La precedente edizione della manifestazione (che ha cadenza biennale) ha visto la presenza di 43.052 visitatori provenienti da 100 Paesi con il 35% di visitatori esteri e 833 espositori provenienti da 47 Paesi che hanno occupato una superficie espositiva di 66.000 mq.

"In 12 anni il salone è diventato uno degli appuntamenti più importanti del settore a livello internazionale" ha dichiarato Armin Wittmann, Exhibition Group Director automatica, nella conferenza stampa di presentazione.

Rispetto allo stesso periodo dell'edizione precedente, la superficie espositiva già prenotata è superiore dell'11% (36% in più per quanto riguarda gli espositori esteri), mentre gli espositori confermati vedono un incremento del +13%. Le aziende alla prima partecipazione occupano il 17% della superficie complessiva, con nomi quali: Alfing Montagetechnik, Asic Robotics (CH), Canon, Doosan Robotics (KR), Dunkermotoren, Magazino, Misumi, Rollon, SICK, SIEMENS, Stein Au-

tomatica, TÜV SÜD, Shenyang SIASUN Robot (CN), Sumitomo, Volkswagen, WAGO Kontakttechnik, wenglor sensoric, ZOLLERN.

Tra i protagonisti di Automatica 2018, ci sono le aziende più importanti a livello mondiale di questo settore: ABB, Siemens, Kuka, Fanuc, Festo, Beckhoff e tante altre ancora.

Proponendo i temi della trasformazione digitale in produzione, della collaborazione fra uomini e robot e della robotica di servizio, Automatica offre un contributo importante alla realizzazione del Lavoro 4.0, dove la figura umana assume un ruolo e una responsabilità ancora più centrali.

I settori espositivi di Automatica 2018 coinvolgono le seguenti aree:

- Assemblaggio e manipolazione
- Visione industriale
- Robotica
- Robotica di servizio professionale
- IT2Industry

Numerose le aziende italiane che hanno confermato la propria presenza; tra queste: Comau, Flli Bettinelli, CEIA, BETTINELLI, Robotic Business Line, Cosberg, GIMA, IT+Robotics, SIR, Sinteco e tante altre.

Programmi collaterali

Per la prima volta ad Automatica AUTOMOBIL PRODUKTION, Convegno specializzato sul tema della digitalizzazione e intelligenza artificiale nella produzione di autoveicoli, in programma il 18-19 giugno 2018. Tra le keynote:

- Nico Feirer, Leiter Globale Seriensteuerung, Beschaffungsprozesse und -systeme Volkswagen Konzernbeschaffung: "Il nuovo rapporto fra VW e i suoi fornitori";

<http://2018.automatica-munich.com/>

5-7 GIUGNO 2018

PCIM Europe 2018

Norimberga • Germania



Con quasi 500 espositori (il 51% proveniente dall'estero), l'edizione 2018 di PCIM si prepara a superare ancora una volta i numeri record del 2017, confermandosi l'evento europeo più importante per l'elettronica di potenza. Un evento che vede la presenza di tutti i keyplayer globali ma anche di tantissime aziende di piccole dimensioni che operano nell'ambito dell'elettronica industriale. L'attenzione si concentra sui semiconduttori di potenza e sui componenti passivi, proposti rispettivamente dal 36% degli espositori; gli alimentatori e i sistemi di controllo del calore occupano il secondo posto col 25% mentre bobine e materiali magnetici vengono proposti dal 20% degli espositori. Questi temi rendono l'evento particolarmente interessante per i visitatori dei settori dell'elettronica industriale, dell'elettromobilità, dei veicoli, dell'automazione e dei sistemi di alimentazione. Nel padiglione 6, durante tutti i tre giorni della fiera, è in programma una nutrita serie di conferenze dedicate agli attuali sviluppi del settore ed alle tendenze future. In agenda sono previste, tra le altre, due conferenze dal titolo "From Niche to Mainstream - SiC Power Devices as Enabler for Revolutionary Designs" (Infineon Technologies) e "Automotive Power Modules - Design Changes and Technology Innovations to Come?" (Yole Développement) nonché due tavole rotonde organizzate dalla rivista professionale "Bodo's Power Systems" su "SiC - Devices for the Future Design" e "GaN - Devices for the Future Design". Parallelamente alla fiera si terrà la Conferenza internazionale dedicata all'elettronica di potenza che, con oltre 300 contributi inediti, tenterà di colmare il divario tra ricerca e industria; la Conferenza rappresenta il punto d'incontro per esperti e utilizzatori dell'elettronica di potenza di tutto il mondo.

www.pcim.de



■ 26-28 GIUGNO 2018
SENSOR + TEST
Norimberga • Germania

SENSOR + TEST è una delle più importanti manifestazioni mondiali dedicata ai sensori e alle tecnologie di misura e test; nel 2017 sono stati ben 569 gli espositori provenienti da 29 paesi diversi che hanno presentato i loro prodotti nei padiglioni della Fiera di Norimberga, città tedesca che ospita la manifestazione. Le Conferenze internazionali che integrano l'evento - *ettc2018 European Test and Telemetry Conference* e *ITG / GMA-Fachtagung Sensoren und Messsysteme 2018* - offrono ai partecipanti lo stato dell'arte della ricerca in questo campo nonché una visione del futuro di questo settore. Oltre alle Conferenze sono previste 55 presentazioni da parte delle aziende espositrici che proporranno i loro ultimi sviluppi, prodotti e servizi nei padiglioni 1 e 5 durante tutti i tre giorni della fiera. Tra queste, una sessione speciale sul tema "*Sensor and Measuring Technology in the Industrial Internet*". Particolarmente apprezzate dai visitatori sono le dimostrazioni pratiche offerte da numerosi espositori, specie quelle relative ai test sui veicoli. Tutte le informazioni aggiornate su espositori, conferenze e altre iniziative sono disponibili online sul sito web di SENSOR + TEST.

www.sensor-test.com



■ 19-21 GIUGNO 2018
IFSEC International
Londra • UK



IFSEC International è l'appuntamento europeo di riferimento per i produttori di sistemi per la sicurezza fisica e informatica; l'anno scorso la manifestazione ha visto la presenza di oltre 27 mila visitatori provenienti da 116 differenti paesi. Gli espositori sono stati circa 600, con una discreta rappresentanza di aziende italiane.

Quest'anno più che mai IFSEC cercherà di dare risposte più precise alle nuove esigenze di un mercato in continua evoluzione; nel 2017 il panorama della sicurezza è cambiato in maniera significativa con attacchi sempre più complessi che hanno sconvolto il mondo. In un panorama sempre più connesso sono aumentati i cyber-attacchi mentre altre minacce sono cresciute nell'imprevedibilità e nell'invasività. Persone, perimetri, prodotti e database appaiono più vulnerabili che mai. La manifestazione, suddivisa in sei aree tematiche, oltre a mettere in mostra il meglio di prodotti fisici e sistemi integrati in grado di aumentare il livello di sicurezza, offre anche un ventaglio di risorse (seminari, convegni, presentazioni) che hanno lo scopo di accrescere le capacità di quanti si

occupano professionalmente di sicurezza: strumenti indispensabili per installatori, integratori, distributori e consulenti, che costituiscono il target della manifestazione londinese. Agli installatori, IFSEC offrirà un ricco programma di dimostrazioni pratiche e *Show Me How* con applicazioni complete mentre per gli integratori ci sarà la possibilità di confrontare tecnologie differenti e seminari di più alto livello. Queste le principali aree tematiche di IFSEC 2018:

- Physical & Perimeter Security
- IT & Cyber Security
- Video Surveillance
- Access Control
- Integrated Security
- Intruder Alarms, Detectors and Sensors

Anche quest'anno ICE e ANIE Sicurezza organizzeranno una partecipazione collettiva italiana a IFSEC International, che quest'anno si svolgerà dal 19 al 21 giugno 2018 presso il Centro Espositivo ExCeL di Londra.

www.ifsec.events/international/

MOSTRE MERCATO

CENTOBUCHI (AP)
FIERA MERCATO DELL'ELETTRONICA
Centro Logistico Orlando Marconi
Organizzazione: Federvol
Tel: 3477433924
Fax: 0734841316
federvol@gmail.com
www.federvol.it
14 e 15 Luglio 2018

LOCRI (RC)
MOSTRA MERCATO DEL RADIOAMATORE, DELL'ELETTRONICA E DEL COMPUTER
Istituto di I° Grado F. Maresca
Organizzazione: G.C.S.
Tel: 3382704739
gallcosimo@tiscali.it - www.cisarlocri.net
21 e 22 Luglio 2018

L'elenco aggiornato di tutte le Mostre Mercato del 2018 e dei più importanti eventi nazionali e internazionali di elettronica, sicurezza e fonti rinnovabili è disponibile sul sito www.elettronica.in

ARROW

Five Years Out

presenta la 6° edizione di

GET CONNECTED



Sicuri di sapere tutto sulla
Blockchain IoT?

Non perdetevi l'evento dell'anno!
Avrete la possibilità di toccare con
mano l'intero ecosistema IoT e fare
un'esperienza from "sensor to
sunset", dal sensore al cloud.
Oltre 900 mq di show case, tanti
produttori di componentistica e
tantissimi partner tra incubatori e
system integrators

20 GIUGNO 2018
NH MILANO
CONGRESS
CENTRE



Conferma la tua
presenza tramite il QR
code o registrati su
www.arrow.com

Produttori



Partners



In collaborazione con



AMS introduce un progetto di riferimento per sensore di pressione sanguigna senza bracciale gonfiabile

AMS ha annunciato il rilascio del primo progetto di riferimento di sensore per pressione sanguigna basato sul suo IC AS7024: questa soluzione può eseguire misurazioni della pressione arteriosa 24 ore su 24, precise, veloci e convenienti. Il nuovo design di riferimento basato su AS7024 integra tutti i componenti hardware, incluso l'AS7024, nonché il software necessario per implementare la misurazione della pressione sanguigna, la misurazione della frequenza cardiaca (HRM), la variabilità della frequenza cardiaca (HRV) e l'elettrocardiogramma (ECG). Il sensore AS7024 comprende tre



LED; fotodiodi, un front end ottico con sequencer per HRM e un front end analogico per ECG in un package a basso profilo da 6,1 x 2,7 mm. Questa soluzione apre la strada al monitoraggio dei segni vitali in una gamma

più ampia di dispositivi e mercati finali quali il monitoraggio automobilistico o il controllo personale della salute.

Il software proprietario nel progetto di riferimento AS7024, sviluppato in collaborazione con alcuni partner, analizza le misurazioni HRM e ECG sincronizzate per calcolare la pressione sanguigna. L'accuratezza delle misurazioni della pressione arteriosa dell'AS7024 è stata convalidata in uno studio clinico condotto presso l'Università di Medicina di Graz, in Austria, secondo lo standard IEEE per dispositivi indossabili senza bracciale gonfiabile.

www.ams.com

ATS344, sensore differenziale ad effetto Hall con magnete di polarizzazione



scita PWM in current-mode per ridurre al minimo il numero di pin. È presente anche una EEPROM in grado di supportare fino a 100 cicli di lettura/scrittura sulla quale vengono memorizzati i parametri di calibrazione. L'ATS344 è ideale per applicazioni automobilistiche che richiedono il rilevamento ad elevata risoluzione del moto lineare (long stroke > 5 mm).

Il magnete di back-bias integrato semplifica notevolmente l'installazione e la messa in funzione, richiedendo solo un target ferromagnetico da rilevare. L'ATS344 è disponibile in un package SP a 3 pin.

www.allegromicro.com

Allegro MicroSystems Europe ha introdotto un nuovo IC con sensore ad effetto Hall programmabile ad alta precisione, completo di magneti di polarizzazione. L'ATS344 di Allegro include due sensori ad effetto Hall, funzionamento differenziale e un package comprendente anche un magnete per la polarizzazione. Il sensore dispone di un'u-

KOE introduce un nuovo display TFT WXGA Rugged+

KOE ha annunciato l'introduzione di un nuovo display TFT Rugged+ da 7,0 pollici. Il modello TX18D212VM-0BAA con risoluzione WXGA (1280 x 768 pixel) presenta un formato di 5:3, un'ampia gamma di temperature di esercizio e la tecnologia IPS (In-Plane Switching), che garantisce prestazioni ottiche eccezionali da qualsiasi angolo di visuale.

Il display Rugged+ da 7,0 pollici è progettato per installazioni su veicoli e per ambienti industriali, medici, aerospaziali e marini ad alta affidabilità, soggetti a temperature estreme, vibrazioni e urti meccanici. Le caratteristiche ambientali e ottiche dei moduli di visualizzazione

Rugged+ di KOE si basano sugli stessi rigorosi standard impiegati nelle specifiche dei display destinati all'industria automobilistica.

Il modulo di visualizzazione ha un controllo del dimming integrato, che consente di regolare la luminosità della retroilluminazione con un PWM o un ingresso di tensione CC.

www.koe-j-display.com



LTM4661, regolatore Boost μ Module compatto per sistemi ottici a bassa tensione

Analog Devices presenta l'LTM4661 Power by Linear, un regolatore μ Module step-up low power in package BGA di 6,25mm x 6,25mm x 2,42mm. Per completare un progetto sono necessari soltanto pochi condensatori e una resistenza, la soluzione occupa meno di 1cmq su PCB a singola faccia o 0,5cmq su doppia faccia. L'LTM4661 incorpora un controllore a commutazione DC/DC, i MOSFET, induttori e componenti

di supporto. L'LTM4661 funziona con tensioni di ingresso da 1,8V a 5,5V, e continua a operare fino a 0,7V dopo lo start-up. La tensione in uscita è regolabile tra 2,5V e 15V con



una sola resistenza. La combinazione di un package piccolo e sottile e di un ampio intervallo di tensioni di ingresso e uscita lo rende la soluzione ideale per svariate applicazioni, tra cui moduli ottici, apparati alimentati a batteria, sistemi di back-up, tensione di bias per amplificatori di potenza o diodi laser e piccoli motori in DC. L'LTM4661 può erogare 4A con continuità da 3,3VIN a 5VOUT e, sempre di continuo, 0,7A da 3,3VIN

a 12VOUT. L'LTM4661 usa il raddrizzamento sincrono, che raggiunge l'elevato livello di efficienza di conversione del 92% (da 3,3VIN a 5VOUT). La frequenza di commutazione è 1MHz ed è sincronizzabile a un clock esterno da 500kHz a 1,5MHz. L'architettura a due fasi con singola uscita dell'LTM4661 e lo switch a 1MHz garantiscono una risposta veloce ai transitori su variazioni di linea e di carico.

www.analog.com

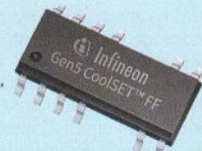
Infineon annuncia la 5^a generazione di dispositivi CoolSET per sistemi di alimentazione

Infineon Technologies annuncia la 5^a generazione di sistemi CoolSET a frequenza fissa con tensione di 700 / 800 V. Questa soluzione combina un controller PWM con gli ultimi MOSFET CoolMOS P7 da 700 V e 800 V in un unico package. La singola piattaforma supporta topologie di flyback isolate e non isolate. Il

nuovo CoolSET a 700 V / 800 V a frequenza fissa utilizza un MOSFET supergiunzione ad alta tensione in combinazione con un regolatore di corrente interno in una configurazione cascode. Questa combinazione offre un avvio rapido e una facile implementazione della protezione. Il MOSFET a 800 V integra supporto ingressi fino a 350 V

CA. Le perdite di commutazione sono ridotte grazie alla particolare modalità di funzionamento che riduce la frequenza di commutazione con carico medio e medio-basso.

Oltre alle protezioni nei confronti dei cortocircuiti in-uscita, del sovraccarico e della sovratensione, il CoolSET è in grado di rilevare anomalie nella linea



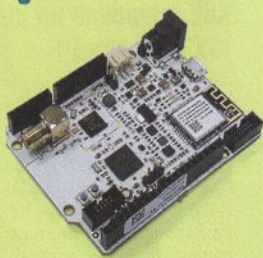
di ingresso. Tutte le modalità di protezione dispongono di ripristino automatico per ridurre al minimo le interruzioni.

www.infineon.com

Arrow Electronics presenta la nuova scheda SmartEverything Tiger

Arrow Electronics ha esteso la propria famiglia SmartEverything con la scheda Tiger, basata sulla MCU multi-protocollo wireless Kinetis KW41Z di NXP Semiconductors per dispositivi connessi a bassissimo consumo.

Pronta per un immediato avvio dello sviluppo, la scheda Tiger rappresenta una piattaforma ideale per prodotti quali la chiusura intelligente di porte, la tecnologia per le applicazioni portatili in ambito sanitario, i monitor indossabili per uso sportivo, i controlli RF a distanza, tastiere e mouse wireless per computer, controllori di videogiochi, sistemi di sicurezza per controllo di accessi, applicazioni di energia intelligente e apparecchiature di rete. I connettori compatibili Arduino offrono agli utenti la flessibilità per aggiungere nuove funzionalità a valore aggiunto ai loro progetti, mentre le funzioni di sicurezza integrate nella MCU Kinetis KW41Z, fra cui l'acceleratore crittografico hardware AES-256 e un generatore TRNG (True



Random Number Generator), consentono di avere la massima sicurezza dei dati e degli asset. Oltre all'efficiente core Arm Cortex-M0+, la MCU wireless Kinetis KW41Z contiene le due radio BLE (Bluetooth Low Energy) v4.2 e IEEE 802.15.4, e può operare con più protocolli wireless concorrenti. La scheda Tiger integra anche la funzionalità Wi-Fi, che estende ulteriormente le opzioni di connettività disponibili per gli utenti e supporta le applicazioni intelligenti end-node come ponte fra Wi-Fi e Thread o BLE.

www.arrow.com

Da Nordic Semiconductor la prima soluzione Zigbee per il SoC nRF52840

Nordic Semiconductor annuncia la sua prima soluzione Zigbee per il SoC multiprotocollo nRF52840. Si tratta di una release certificata Zigbee 3.0 che sarà disponibile dalla seconda metà del 2018. Come con altre soluzioni mesh di Nordic, ciò consente il funzionamento simultaneo di Zigbee 3.0 con Bluetooth Low Energy utilizzando il dispositivo S140 v6.0 SoftDevice. La soluzione multiprotocollo di Nordic per il SoC nRF52840 consente casi d'uso che sfruttano diverse tecnologie wireless. Gli esempi includono l'interazione con una rete di illuminazione intelligente Zigbee da uno smartphone tramite Bluetooth 5 o una rete di sicurezza Zigbee utilizzando i beacon Bluetooth per l'identificazione della posizione. L'SDK include diversi esempi tra cui il funzionamento simultaneo di Zigbee e Bluetooth 5 per lampade e inter-

ruttori intelligenti. Zigbee è una soluzione wireless globale basata su standard che sfrutta l'IEEE 802.15.4 PHY e opera a livello globale in bande senza licenza a 2,4 GHz (e in altre bande di frequenza in alcune zone geografiche). Le velocità di trasmissione dati raw di 250 kbps può essere raggiunta a 2,4 GHz (16 canali) e la distanza di trasmissione varia da 10 a 100 metri, a seconda della potenza e delle caratteristiche ambientali. La tecnologia supporta reti mesh di grandi dimensioni (che consentono ai nodi di inoltrare messaggi ad altri nodi) e operazioni a bassa potenza per prolungare la durata della batteria.

www.nordicsemi.com



MAX32660 e MAX32652, micro di Maxim dai consumi particolarmente bassi



I progettisti di sensori IoT, sensori ambientali, smartwatch, indossabili medicali o di pre-

venzione sanitaria ed altri dispositivi sottoposti a stretti vincoli dimensionali, possono oggi prolungare la durata delle batterie ed aumentare le funzionalità utilizzando i microcontrollori a bassissimo consumo MAX32660 e MAX32652 di Maxim Integrated. Questi microcontrollori sono basati sul processore Arm Cortex-M4 con FPU e consentono ai progettisti di sviluppare applicazioni avanzate in presenza di vincoli di consumo molto stringenti. Il MAX32660 offre ai progettisti una combinazione ideale, fornendo loro una quantità di memoria sufficiente per eseguire alcuni algoritmi avanzati e gestire sensori (256 KB di flash e 96 KB di SRAM). Offre inoltre un eccellente rapporto consumo/prestazioni (fino 50 μ W/MHz), dimensioni

incredibilmente piccole (1,6 mm x 1,6 mm in contenitore WLP) ed un prezzo contenuto. Con 3 MB di flash ed 1 MB di SRAM integrati sul chip ed un clock fino a 120 MHz, il MAX32652 costituisce una soluzione ad alta integrazione rivolta ai dispositivi IoT che puntano ad una maggiore capacità di elaborazione.

www.maximintegrated.com

ISL91302B, ISL91301A e ISL91301B, nuovi PMIC multifase di Renesas

Renesas Electronics annuncia tre nuovi dispositivi programmabili per la gestione dell'alimentazione (PMICs) che offrono il più elevato livello di efficienza ed il più piccolo footprint per applicazioni quali gli application processors ed i tablets. I nuovi PMIC sono stati progettati per alimentare processori per l'intelligenza artificiale (AI), FPGA e sono l'ideale per generare le alimentazioni richieste dai

drives allo stato solido (SSD), e transceivers ottici ed una ampia gamma di altri dispositivi. ISL91302B è un PMIC con

uscita singola e doppia multi fase in grado di fornire fino a 20 Ampere, con un livello di efficienza di picco che raggiunge il 94% in una soluzione che richiede una superficie di soli 70 mmq. Questo significa un ingombro del 40 per cento inferiore rispetto a quello dei PMIC della concorrenza. Oltre



Renesas ISL91301/02 Multiphase PMIC Deliver Highest Efficiency and Smallest Footprint

all'ISL91302B, ecco gli altri dispositivi presentati da Renesas: ISL91301A, PMIC con uscita tripla, e ISL91301B, PMIC con uscita quadrupla, sono entrambi in grado di fornire fino a 16 Ampere con un livello di efficienza di picco che raggiunge il 94%. I nuovi PMIC programmabili fanno leva sulla tecnologia di Renesas denominata R5 che consente di gestire una risposta veloce ai transienti

in un singolo ciclo, una compensazione digitale ed una frequenza di switching di ben 6 MHz durante la variazione del carico. Queste caratteristiche rendono più semplice la progettazione di alimentatori grazie alla possibilità di utilizzare induttori di dimensioni ridotte (solamente 2 mm x 2 mm con spessore di 1 mm), piccoli condensatori ed un numero limitato di componenti passivi.

www.renesas.com

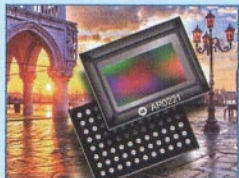
Sensore di immagini all'avanguardia per sensibilità e rapporto SNR

ON Semiconductor ha introdotto il primo sensore di immagine CMOS da 2,1 megapixel, formato 1/1,7 pollici con pixel BSI (Back Side Illuminated) da 4,2 μm di nuova concezione: l'AR0221 garantisce una sensibilità impareggiabile in condizioni di scarsa luminosità in ambito industriale.

L'AR0221 offre una eccezionale gamma High Dynamic Range (HDR) con una risoluzione del sensore di 1936H x 1096 V, supporta il formato 1080p a 30 fps e offre un eccezionale rapporto segnale-rumore (SNR) nelle gamma del visibile e nel vicino infrarosso (NIR). Il suo rapporto 16:9, i colori vividi e l'elevato contrasto lo rendono ideale per le più impegnative applicazioni industriali. Il sen-

sore offre una doppia interfaccia dati: quella MIPI CSI-2 4-lane e l'SLVs HiSpi. Progettato per soddisfare specifiche di livello industriale, l'AR0221 può operare in ambienti esterni gravosi, con la temperatura di funzionamento che può variare tra -30 °C e +85 °C. Disponibile in un package iBGA da 12 mm x 9 mm durevole, affidabile e robusto, presenta anche un rivestimento in vetro antiriflesso.

www.onsemi.com



STNRG011, controller digitale per circuiti di alimentazione con elevati standard di efficienza

Il controller digitale di potenza STNRG011 di STMicroelectronics semplifica e accelera lo sviluppo di alimentatori e adattatori da 90-300 W per computer, illuminazione a LED, apparecchiature medicali, telecomunicazioni e apparecchiature industriali, rispettando i più rigorosi standard relativi all'efficienza energetica.

L'STNRG011 combina un controller risonante LLC half-bridge con un controller PFC (Power-Factor Correction) multi-mode, gestito da algoritmi di controllo best-in-class. Le periferiche digitali integrate che sfruttano la tecnologia SMED (state machine event state) della ST e l'hardware analogico proprietario accelerano le prestazioni



del loop di controllo per un'eccellente risposta dinamica. La memoria non volatile viene fornita anche on-chip, per la memorizzazione di parametri specifici dell'applicazione.

La facilità di programmazione del controller digitale STNRG011 aiuta gli ingegneri a ottimizzare l'efficienza e le prestazioni nell'intero intervallo di carico nonché sfruttare il funzionamento in modalità burst con carichi leggeri per massimizzare l'efficienza.

www.st.com

TLP5832, fotoaccoppiatore per gate driver con corrente di 2,5A

Toshiba Electronics Europe ha annunciato il lancio di un nuovo fotoaccoppiatore per gate driver alloggiato in un package SO8L a basso profilo. Il TLP5832 fornisce una corrente di uscita di picco pari a 2,5A (IOPH, IOPL) e può pilotare direttamente IGBT e MOSFET in applicazioni come gli inverter per le applicazioni industriali, i condizionatori d'aria e le applicazioni fotovoltaiche, e inoltre i servoamplificatori.

L'adozione del package SO8L fornisce un dispositivo che è alto appena 2,3mm - circa il 54% in meno rispetto ai prodotti attuali di Toshiba che sono disponibili in package SDIP6 e DIP8. Il package SO8L assicura il supporto alle moderne applicazioni, nelle quali spesso l'altezza e lo spazio di montaggio sono limitati.

Nonostante le sue dimensioni ridotte, l'IC offre una tensione di



isolamento (BVS) di 5000Vrms e garantisce distanze di isolamento in aria e superficiali minime di 8,0mm, il che lo rende adatto per applicazioni che richiedono alte prestazioni di isolamento, inclusi i progetti critici per la sicurezza. Inoltre, il nuovo TLP5832 garantisce un tempo di ritardo di propagazione (tPLH, tPHL) di 200ns e una deviazione del ritardo di propagazione (tPsk) di $\pm 80\text{ns}$ lungo l'intero intervallo di temperature operative da -40°C e +110°C.

www.toshiba.semicon-storage.com

Controlla i tuoi dispositivi da remoto tramite LAN e Internet.



NOVITÀ

€ 88,00

Cod. FT1333M

SCHEDA ETHERNET CON 4 USCITE A RELÉ, 8 I/O DIGITALI E 4 INGRESSI ANALOGICI

- * 4 uscite a relé N.O. e N.C. con LED di stato;
- * Invio allarmi via e-mail;
- * Gestione uscite relé in modalità manuale o tramite programmazione oraria;
- * Possibilità di associare un ingresso digitale all'uscita relé di una scheda remota;
- * Alimentazione 7-18 Vdc;
- * Dimensioni 107 x 100 x 20mm;
- * Web server integrato;
- * Fornita già montata e collaudata.

SCHEDA ETHERNET CON 8 RELÉ 16A

- * 8 uscite a relé N.O. e N.C. con LED di stato;
- * Alimentazione 12Vdc;
- * Dimensioni 143 x 84 x 20mm;
- * Fornita già montata e collaudata.

€ 96,00

Cod. ETH008

€ 68,00

Cod. ETH-RLY02

SCHEDA ETHERNET CON 2 RELÉ 16A

- * 2 uscite a relé N.O. e N.C. con LED di stato;
- * Alimentazione 12Vdc;
- * Dimensioni 85 x 66 x 20mm;
- * Disponibile applicazione gratuita per Android;
- * Fornita già montata e collaudata.

SCHEDA ETHERNET CON 4 RELÉ 16A

- * 8 I/O digitali e 4 ingressi analogici;
- * 4 uscite a relé N.O. e N.C. con LED di stato;
- * Alimentazione 12Vdc;
- * Dimensioni 143 x 84 x 20mm;
- * Disponibile applicazione gratuita per Android;
- * Fornita già montata e collaudata.

€ 96,00

Cod. ETH484

Prezzi IVA inclusa.

CeBIT

Un CEBIT tutto nuovo ci aspetta a giugno: la più grande manifestazione dell'Information Technology cambia pelle e si trasforma in un Business Festival dell'Innovazione.

CEBIT SI FA PALADINO DELLA DIGITAL TRANSFORMATION. A PARTIRE DAL LOOK.

di ARSENIO SPADONI

Una T-shirt con una sgargiante scritta gialla "CEBIT" invece della tradizionale giacca e cravatta: così si è presentato Hartwig von Saß - Capo Ufficio Stampa della manifestazione - alla conferenza stampa di presentazione dell'edizione 2018 del CEBIT. Proprio a significare anche nel look la svolta che Deutsche Messe, che organizza la manifestazione, vuole impri-

mere a una delle più importanti manifestazioni tecnologiche del mondo. Una fiera che fino all'anno scorso potevamo classificare tra quelle di Information Technology e che, sull'onda delle trasformazioni in atto, vuole diventare la fiera della Digital Transformation un Business Festival dell'Innovazione. Un cambiamento drastico, e una scelta sicuramente coraggiosa

che tenta di conciliare una raffigurazione del nuovo che sta arrivando con un modello di business tradizionale, anche se in evoluzione, offrendo nuovi motivi di interesse per espositori e visitatori. Una strada che anche altre manifestazioni simili, come il CES di Las Vegas, stanno percorrendo, anche se con cambiamenti meno radicali di quelli preannunciati per CEBIT 2018. Un CEBIT, dunque, completamente rinnovato: nuovi temi, nuovi format, nuovo layout espositivo, nuova data alle soglie dell'estate. Mantiene il suo ruolo di appuntamento per il business, ma promette nuove emozioni. Al centro della sua attenzione ci sono gli otto temi forti della digitalizzazione: intelligenza artificiale, Internet delle Cose, realtà virtuale e aumentata, sicurezza, blockchain, droni e sistemi a pilotaggio remoto, mobilità futura e robotica umanoide.

"Con la sua nuova, triplice fisionomia di vetrina delle innovazioni, di palcoscenico di conferenze di alto profilo e di evento per un networking informale, il nuovo CEBIT offrirà una panoramica a 360 gradi delle soluzioni per la digitalizzazione delle aziende, della pubblica amministrazione e della società", spiega Oliver Frese, Responsabile CEBIT nel CdA di

Deutsche Messe. "Oltre che esperti dell'IT, addetti ai lavori del settore e decision maker di alto livello del mondo digitale e dell'industria, con il nuovo concept vogliamo però interpellare sempre di più soprattutto la futura generazione dei decision maker delle aziende".

QUATTRO AREE PER LA PRESENTAZIONE DI INNOVAZIONI E VISIONI DEL FUTURO

Il nuovo concept di CEBIT prevede quattro aree o ambiti centrali: **d!conomy**, **d!tec**, **d!talk** e **d!campus**. Nell'area **d!conomy** professionisti dell'IT e decision maker delle aziende, del settore pubblico e del commercio troveranno tutto quello che è necessario per digitalizzare con successo le attività dell'economia e della pubblica amministrazione. I settori espositivi di quest'area sono Digital Business, Digital Administration, Internet of Things, Security, Communication & Networks, Channel & Distribution, Mobile Solutions e Data Center. Protagonisti di **d!tec** saranno invece sviluppatori, ricercatori e startup; **d!talk** sarà lo spazio dei visionari, dei pensatori trasversali, degli esperti e dei creativi di tutto il mondo. Ma il cuore pulsante del



nuovo CEBIT è **d!campus** – che prevede ampi spazi per chi voglia fare networking in un clima informale e rilassante, gustando street food e ascoltando musica dal vivo.

CEBIT prenderà il via lunedì 11 giugno con un fitto programma di conferenze mentre l'esposizione vera e propria aprirà martedì 12.

IL FITTO CALENDARIO DI CONFERENZE

Durante il "Take-off Monday" – il lunedì di apertura – relatori di alto profilo esploreranno il tema della digitalizzazione in vari format presso il Convention Center (CC) e alla Welcome Night in calendario la sera nel padiglione 11.

Il primo highlight della giornata è una leggenda del mondo della realtà virtuale: lo statunitense **Jaron Lanier**, che terrà la keynote inaugurale del "Take-off Monday". Lanier è un pioniere della realtà virtuale e uno dei critici più feroci di Internet. Informatico, imprenditore, compositore e saggista, lavora come scienziato per Microsoft Research e ha ricevuto nel 2014 il prestigioso Premio tedesco per la Pace come riconoscimento per la sua critica fondamentale ai modelli di business delle maggiori società di Internet.

Il forum Future Work spalanca gli scenari del lavoro futuro.

Il forum, che si terrà subito dopo la keynote di Lanier, affronterà il tema della valorizzazione delle possibilità offerte dalla digitalizzazione dell'economia e del lavoro, allo stesso tempo dimostrando l'infondatezza dei timori di perdita del posto di lavoro e di qualificazione per l'uomo. Discuteranno tra l'altro del futuro del lavoro Christiane Benner, del Sindacato tedesco metalmeccanico IG Metall, e Dieter Spath, Presidente dell'Accademia nazionale tedesca di scienze e tecnologia.

A chi serve la digitalizzazione?

Un altro punto del programma del Take-off Monday è il CEBIT Executive Talk, che affronterà il problema di come utilizzare le sempre più potenti tecnologie digitali al servizio dello sviluppo economico, del benessere e della coesione sociale. Interverranno tra gli altri Dieter Kempf, Presidente dell'Associazione federale dell'industria tedesca (BDI), Reimund Neugebauer, Presidente della Società Fraunhofer, e Frank Riemensperger, CEO di Accenture.

Altri forum e occasioni di dibattito sul tema della digitalizzazione con la presenza di esponenti



di alto profilo del mondo dell'economia e della politica sono attualmente in fase di programmazione.

PIONIERI DI NUOVE IDEE E ICONE DELL'IT DAL VIVO SUL PALCO

Al CEBIT è atteso l'autore del "Diario segreto di Steve Job", **Dan Lyons**, che il 14 giugno accompagnerà il pubblico dietro le quinte del settore dell'high-tech rivelando quello che ha vissuto come dipendente nell'"inferno delle startup" (è l'espressione da lui stesso utilizzata). **Ranga Yogeshwar**, il più famoso giornalista scientifico tedesco, risponderà il 13 giugno alla provocatoria domanda "Chi programma chi?", esponendo le sue opinioni più personali sul futuro del rapporto tra uomo e computer.

L'intervento di **Julia Shaw** promette invece un appassionante viaggio nella psiche umana: il 13 giugno la ricercatrice londinese spiegherà infatti come la memoria dell'uomo possa essere manipolata e che cosa questo significhi in termini pratici. Il 12 giugno, **Mikko Hypponen**, il famoso esperto internazionale di sicurezza informatica che è anche Chief Research Officer dello specialista finlandese di IT Security F-Secure, illustrerà ai visitatori di CEBIT i maggiori rischi connessi al crescente successo di Internet of Smart Things e spiegherà come le aziende possono proteggersi dagli attacchi informatici.

Lunedì 11 giugno è prevista anche **Welcome Night** con la presenza di personaggi di spicco del mondo della politica tedesca ed estera, evento che preparerà il terreno per le tre giornate successive, dedicate al business nel quartiere fieristico. Il "Venerdì digitale" (15 giugno) CEBIT sarà aperto, con interessanti eventi e a una tariffa d'ingresso ridotta, a un più ampio pubblico generico interessato al digitale. La comunità internazionale dei blogger si darà convegno a d!talk

e a CEBIT!Signals. Sono inoltre previste particolari attività per programmatori e sviluppatori (Coder & Developer) con format speciali come gli hackathon, e inoltre gare di volo per piloti di droni e occasioni di recruiting per studenti e giovani professionisti.

CEBIT BLOCKCHAIN SUMMIT, UNA CORSA ALL'ORO VIRTUALE SULLA RETE

La blockchain ha il potenziale di rivoluzionare interi settori dell'economia. Il Forum Economico Mondiale prevede che entro il 2027 circa il 10 per cento del PIL globale sarà gestito con questa tecnologia. Nel suo "Hype Cycle for Emerging Technologies" – il quotatissimo grafico sul ciclo di vita delle tecnologie emergenti – anche la società di ricerca, analisi e consulenza Gartner Research, prevede la piena affermazione della blockchain nei prossimi cinque-dieci anni. Ecco perché CEBIT dedica a questa tecnologia un'intera giornata tematica con il CEBIT Blockchain Summit, che si terrà giovedì 14 giugno al Grand Central Stage del padiglione 27.

Il Summit sarà inaugurato dal professor Wolfgang Prinz dell'Istituto Fraunhofer per la tecnologia dell'informazione applicata (FIT), che spiegherà ai visitatori di CEBIT che cosa si nasconde dietro il successo della blockchain e perché questa tecnologia ha un potenziale così elevato. Prinz si occupa da più di due anni delle basi tecniche della blockchain, e di sviluppo e analisi di applicazioni basate su blockchain.

IL FUTURO DEL COMMERCIO

CEBIT si occuperà anche del futuro del commercio, sicuramente digitale, anche se molti commercianti non sono ancora pronti per la svolta. Secondo un'indagine condotta di recente in Germania tra gli addetti al commercio dall'Associazione tedesca del settore digitale (Bitkom), l'11 per cento degli intervistati non ha ancora un proprio sito web e addirittura il 77 per cento ha dichiarato di non essere aggiornato in tema di digitalizzazione. Esiste dunque un notevole potenziale ancora non sfruttato.

Dall'11 al 15 giugno CEBIT proporrà nuove idee e soluzioni che stimolino la trasformazione digitale – sia di piccoli dettaglianti sia di grandi gruppi commerciali, portando un "pieno di energia digitale" per il settore con un'offerta ben articolata e mirata che prevede informazioni sul futuro del commercio, su strategie di marketing e di vendita orientate alla clientela, supporto e

consigli per la creazione di sistemi multichannel, una rassegna di casi di *best practice* e colloqui con operatori del settore. I punti di incontro più importanti al riguardo saranno gli spazi "Multichannel & Web Presence" e "Customer Centricity" nei padiglioni 16 e 17 dell'area d'economy di CEBIT.

Le soluzioni cloud NetSuite di Oracle

La piattaforma business pienamente integrata NetSuite è la prima del suo genere sviluppata esclusivamente per il cloud ed è quindi facile da implementare, semplice da usare e scalabile praticamente per ogni esigenza. Assicura una funzionalità internazionale B2B e B2C che può essere anch'essa facilmente e prontamente implementata, così da creare una brand experience indifferenziata e ininterrotta attraverso l'integrazione della gestione dei clienti, degli ordini e delle giacenze, del marketing, della gestione finanziaria, dei servizi per il Point of Sale e dello shopping online. A CEBIT Oracle presenterà significativi esempi di applicazioni di NetSuite.

Gli esperti Alexa di Amazon al "Customer Centricity Summit"

Giovedì 14 giugno due forum saranno dedicati, nel padiglione 17, al commercio del futuro. Il primo appuntamento – il "Customer Centricity Summit", che si terrà sul Center Stage – presenterà soluzioni di marketing e di vendita centrate sulle attese e sulle esigenze del cliente. Si parlerà tra l'altro di analitica dei dati, di intelligenza artificiale, di CRM, di realtà virtuale e aumentata, di impiego di video.

LA PRESENZA ITALIANA

Tra le partecipazioni più significative al prossimo CEBIT quella di Zucchetti, prima azienda italiana per fatturato software, che presenterà a imprese e operatori IT la sua nuova offerta di soluzioni tecnologiche per il mercato internazionale.

Il Gruppo Zucchetti ha raggiunto delle dimensioni veramente importanti: 466 milioni di fatturato nel 2017, 150.000 clienti attivi, oltre 900 business partner in Italia e 200 all'estero che operano in 50 differenti paesi.

Soprattutto il grande successo riscontrato negli ultimi anni dalle soluzioni Zucchetti per le grandi imprese, ha convinto la software house italiana a intensificare la propria espansione all'estero, sia mediante acquisizioni e/o accordi

di partnership con importanti player internazionali, sia con una presenza diretta sul territorio. Rientra in questo disegno la nascita nel 2015 di Zucchetti GmbH, società con sede a Voelklin-gen, nel distretto di Saarbrücken, in Germania, diventata il punto di riferimento per l'offerta del gruppo dedicata al mercato tedesco.

"CEBIT è la più importante fiera internazionale del mondo ICT - ha dichiarato Dirk Schwindling, CEO di Zucchetti GmbH - e rappresenta quindi la 'migliore vetrina' per i vendor come Zucchetti che intendono non solo presentare a un pubblico molto qualificato il valore di un'offerta realmente innovativa, ma anche comunicare, in particolare alle aziende tedesche, la nostra vicinanza al cliente dal punto di vista prettamente territoriale, dalla gestione della trattativa, all'installazione delle applicazioni fino al supporto post vendita".

"PRONTI - PARTENZA - VIA!": GIOVANI AZIENDE TENTANO IL DECOLLO

L'obiettivo del nuovo CEBIT - generare business, contatti e nuove idee - ne fa la cornice ideale per giovani aziende emergenti in cerca di partner d'affari e desiderose di presentarsi al mercato internazionale. Il loro punto d'incontro sarà "scale11": ormai collaudato hub per star-

tup, investitori e aziende affermate, che verrà allestito, dall'11 al 15 giugno, all'insegna dello slogan "Ready - steady - go" ("Pronti - partenza - via!") nel padiglione 27 all'interno dell'area di d!tec.

La nuova Tv delle startup

Per la prima volta, verrà mandata in onda ogni giorno di fiera nel primo pomeriggio dal padiglione 27 una trasmissione TV live della durata di un'ora circa dedicata a newcomer, temi e retroscena del mondo delle startup. La trasmissione verrà diffusa attraverso i canali di comunicazione dei media partner dell'Associazione delle startup.

Il Founders Fight Club seleziona le migliori startup

Chi è abbastanza coraggioso da potersi misurare con altri giovani imprenditori su di un palco aperto, potrà prendere parte alla "battaglia verbale" del Founders Fight Club. In ogni match due giovani imprenditori si contenderanno i favori di una giuria di esperti in una schermaglia verbale di carattere imprenditoriale. Altri momenti del programma del Founders Fight Club sono i panel tematici e di esperti e i "lanci" delle startup sullo "Startup Stage" di scale11. La



partecipazione sarà aperta a tutte le startup che esporranno a scale11.

Le FuckUp Night presentano storie di fallimenti

Non sempre le giovani aziende hanno successo. Possono anche fallire. Ecco quindi che le conferenze delle cosiddette FuckUp Night dimostreranno come fallire i propri obiettivi faccia parte della vita e come da questa esperienza si possa anche imparare molto. In uno spazio temporale massimo di 15 minuti, mercoledì 13 giugno le startup racconteranno le storie dei loro fallimenti e daranno a tutti i presenti preziosi consigli per il futuro.

Le FuckUp Night sono un noto evento globale, un format internazionale nato in Messico nel 2012: un incontro nel quale gli imprenditori condividono apertamente con il pubblico le proprie storie di fallimento e i propri errori. Forse anche Yannick Kwik potrà imparare qualcosa da questi incontri. Il famosissimo imprenditore che è alla guida delle FuckUp Night in tutto il mondo visiterà CEBIT il prossimo giugno per farsi un'idea della creatività, dei successi e dei fallimenti delle startup presentate.

Investors Club: un'area VIP per potenziali investitori

A scale11 sarà prevista un'area speciale per le aziende e i potenziali investitori che intendano approfittare della possibilità di avere colloqui riservati con startup di loro interesse in un ambiente esclusivo e tranquillo.

Grandi attese per scale11

La community delle startup guarda con grande interesse alla formula di scale11, alla possibilità di avere colloqui con aziende affermate, di trovare investitori e di confrontarsi con altre realtà analoghe alla propria.

"Abbiamo partecipato attivamente a scale11 fin dai suoi inizi al CEBIT, perché offre la preziosa opportunità di incontrare decision maker del B2B e di proporre i nostri servizi per l'innovazione aziendale. Non vediamo l'ora di essere parte di scale11 a giugno", dice Philippe Thiltges, co-fondatore e managing partner dell'agenzia viennese WhatA-Venture.

Greta Kreuzer, CEO di Cosinuss di Monaco di Baviera, una giovane azienda che opera nel settore medicale, sottolinea: *"Siamo molto lieti di partecipare a scale11 e a CEBIT, perché lì ci si sente*

parte di una 'famiglia', di una rete che si impegna con tutte le sue forze ad aiutare le startup".

LAYOUT E ORARI

È nuovo anche il layout espositivo: i padiglioni da 9 a 13, da 14 a 17 e da 25 a 27, e l'area scoperta intorno al Tetto EXPO accoglieranno interessanti iniziative, mostre ed eventi di networking. Gli stand all'interno dei padiglioni saranno aperti dalle 10.00 alle 19.00. d!campus rimarrà aperto invece fino alle 23.00 – con spazi dedicati alla cultura digitale, a innovativo street food e ad esibizioni dal vivo di artisti famosi come la hip-hop star tedesca Jan Delay, attesissimo per la sera del 13 giugno.

CEBIT IN BREVE	
DOVE:	Centro Fieristico di Hannover (Germania)
QUANDO:	da lunedì 11 a venerdì 15 giugno 2018.
ORARI:	Lunedì: 10.00 - 18.00 (Take-off-Monday: giornata riservata alle conferenze con pre-registrazione). 18.00 - 20.00 Welcome Night (solo su invito) Martedì - Giovedì: 10.00 - 19.00 apertura del fronte espositivo nei padiglioni 10.00 - 23.00 programma d!campus Venerdì: 10.00 - 17.00 apertura dei padiglioni 10.00 - 17.00 programma d!campus
AREE/PROGRAMMA:	d!conomy Host, Connect & Secure your Business (padiglioni 11-13) Digital Enterprise, Digital Office and Digital Administration (padiglioni 14-17) d!tec Game Changing Technologies, Emerging Business Cases & Partner Events (padiglioni 25-27) d!talk Conferenze nei padiglioni 12, 13, 14, 16, 17, 19, 20, 26 e 27 d!campus Mostre, musica dal vivo e Digital Street Festival nell'area scoperta
TEMI CENTRALI:	Intelligenza artificiale, blockchain, mobilità future, robotica, realtà virtuale / aumentata, sicurezza, droni / sistemi a pilotaggio remoto, Internet of Things
Sito: www.cebit.de	

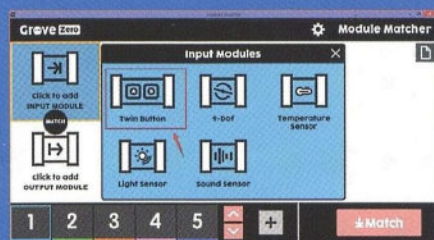
Programmare diventa un gioco!

€125,00

cod. GROVEZEROSTEM

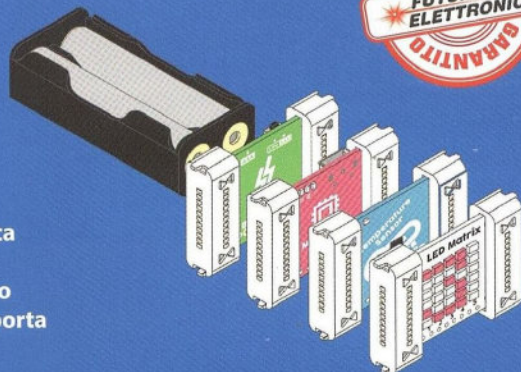
Per piccoli e grandi,
per avvicinarsi
facilmente al mondo
della programmazione

Grove Zero



Grove Zero Starter Kit

Crea progetti facilmente e rapidamente con Grove Zero, un sistema modulare orientato alla prototipazione, composto da una board base a microcontrollore (che ne rappresenta il cervello) e tantissime schede dedicate contenenti sensoristica, interfacce di I/O, link wireless, display e altro ancora, specializzate per vari utilizzi (automazione, domotica, strumentazione, robotica ecc.). Il kit permette di realizzare varie configurazioni semplicemente attaccando insieme le board come fossero mattoncini LEGO®. La programmazione è semplice e supporta Module Matcher, Microsoft MakeCode e JavaScript.



Prezzo IVA inclusa.

**FUTURA
ELETTRONICA®**

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

Caratteristiche tecniche
e vendita on-line su:

www.futurashop.it

Fishino

Design With Simplicity

I TUOI PROGETTI DIVENTANO WIRELESS

FISHINO UNO

- Alimentazione:
 - da 7 a 12 Vdc (tramite plug)
 - 5 Vdc (tramite porta USB)
- Compatibile al 100% con Arduino UNO
- Modulo WiFi
- Interfaccia per scheda MicroSD
- Modulo RTC con batteria di mantenimento
- Sezione di alimentazione a 3,3 V potenziata
- Compatibile con shield e schede millefori
- Peso 25 g
- Dimensioni 76,5 x 53,5 x 14 mm

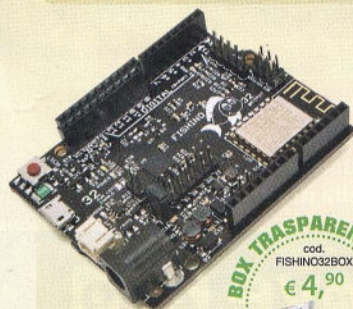


cod. FISHINOUNO

€ 36,00

cod. FISHINOMEGA

€ 49,90



FISHINO32

- Controller PIC32 bit
- Alimentazione:
 - 3,6 Vdc (tramite batteria esterna al litio)
 - da 3,5 a 20 Vdc (tramite plug o sull'ingresso Vin)
 - 5 Vdc (tramite porta USB)
- Circuito di ricarica per batteria al litio
- Compatibile con Arduino UNO
- Modulo WiFi
- Modulo RTC con batteria di mantenimento
- Clock 120 MHz, riducibile via software
- Interfaccia per scheda microSD
- Codec audio stereo
- Peso 29 g
- Dimensioni 76,5 x 53,5 x 14 mm

cod. FISHINO32

€ 59,00



Fishino UNO, Fishino MEGA, Fishino GUPPY, Fishino32 e Fishino PIRANHA sono schede di sviluppo Arduino-compatible dotate di WiFi e lettore di schede microSD. Grazie ad un completo pacchetto di librerie (scaricabile dal sito www.fishino.it), si ha la possibilità di gestire tutti i propri progetti dal web in maniera semplice e veloce!

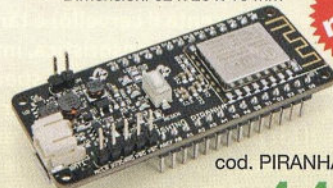


FISHINO MEGA

- Alimentazione:
 - 3,6 Vdc (tramite batteria esterna al litio)
 - da 3,5 a 20 Vdc (tramite plug)
 - 5 Vdc (tramite porta USB)
- Circuito di ricarica per batteria al litio
- Compatibile al 100% con Arduino MEGA
- Modulo WiFi
- Interfaccia per scheda MicroSD
- Modulo RTC con batteria di mantenimento
- Stadio di alimentazione switching (5 Vdc e 3,3 Vdc)
- Compatibile con shield e schede millefori
- Peso 37 g
- Dimensioni 101,5 x 53,5 x 15 mm

FISHINO PIRANHA

- Alimentazione:
 - 3,7 Vdc (tramite batteria esterna al litio)
 - da 3,5 a 20 Vdc (tramite plug)
 - da 3,5 a 20 Volt sull'ingresso Vin
 - 5 Vdc (tramite porta USB)
- Circuito di ricarica per batteria al litio
- Compatibile al 100% con Arduino MKR1000
- Modulo WiFi
- Interfaccia per scheda MicroSD
- Peso 10 g
- Dimensioni 62 x 25 x 16 mm

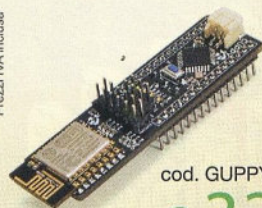


cod. PIRANHA

€ 44,90

FISHINO GUPPY

- Alimentazione:
 - 3,6 Vdc (tramite batteria esterna al litio)
 - da 6,5 a 20 Vdc (tramite plug)
 - 5 Vdc (tramite porta USB)
- Circuito di ricarica per batteria al litio
- Compatibile al 100% con Arduino NANO
- Modulo WiFi
- Interfaccia per scheda MicroSD
- Peso 10 g
- Dimensioni 75 x 20 x 18,5 mm



cod. GUPPY

€ 33,90

Prezzi IVA inclusa

FUTURA ELETTRONICA

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

Caratteristiche tecniche
e vendita on-line su:

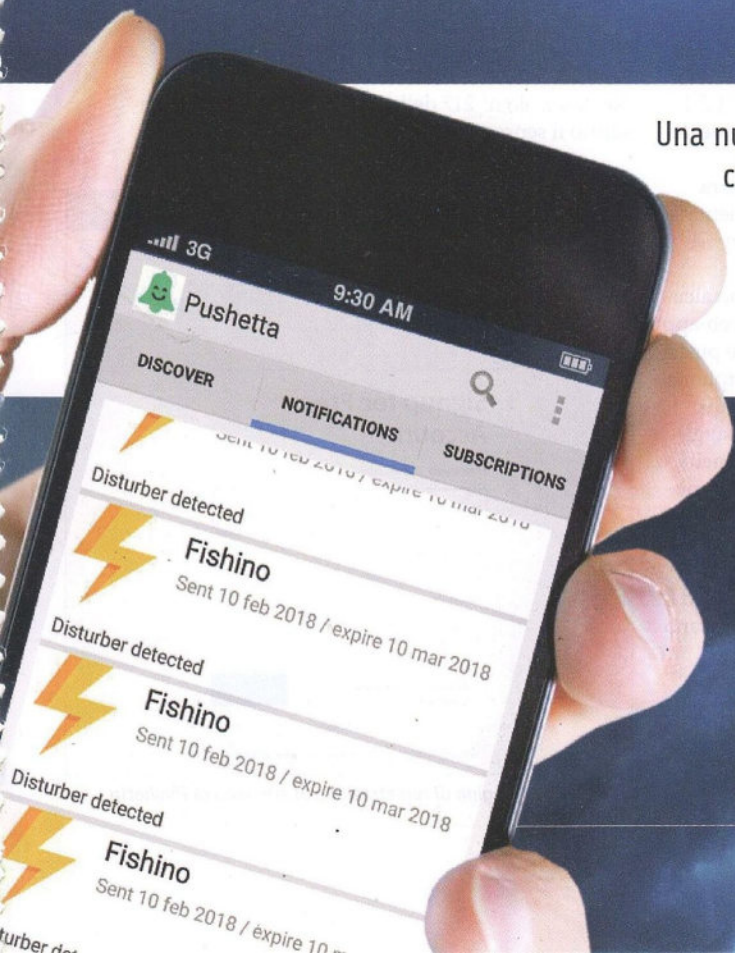
www.futurashop.it

FISHINO, RILEVA I FULMINI E INVIA NOTIFICHE PUSH

Una nuova libreria per le board Fishino consente di inviare con semplicità notifiche push multi-device utilizzando Pushetta. Proviamola in abbinamento al rilevatore di fulmini.

di EMANUELE SIGNORETTA

Dalla presentazione della prima scheda della serie Fishino, abbiamo sviluppato e proposto nuove versioni e arricchito il supporto software, con l'aggiunta di funzionalità nel firmware e nuove librerie che rendono le board "col pesciolino" sempre più competitive e complete. In queste pagine vogliamo descrivervi una nuova libreria chiamata Fishetta. Il nome deriva dalla fusione di Fishino e Pushetta, la quale è una web



Listato 1

```
bool Fishetta::sendPushNotification(String text)
{
  if (client.connect(_serverName, 80))
  {
    //Send data to the server
    client.print("POST /api/pushes/");
    client.print(channel);
    client.println("/ HTTP/1.1");
    client.print("Host: ");
    client.println(_serverName);
    client.print("Authorization: Token ");
    client.println(apikey);
    client.println("Content-Type: application/json");
    client.print("Content-Length: ");
    client.println(text.length() + 46);
    client.println();
    client.print("{ \"body\" : \"");
    client.print(text.c_str());
    client.println("\", \"message_type\" : \"text/plain\" }");
    client.println();
    client.stop();
  }

  return true;
}
```

app che consente di inviare notifiche push a più dispositivi utilizzando una semplice API KEY. Senza entrare nel dettaglio (rimandiamo, per la descrizione completa di Pushetta e del suo funzionamento, al relativo articolo pubblicato nel fascicolo n° 194) ricordiamo che per inviare una notifica push da un dispositivo come un Arduino (o Fishino nel nostro caso) è necessario inviare una richiesta HTTP GET all'indirizzo api.pushetta.com, contenente il nome del canale, la chiave API e il testo del messaggio.

Ciò detto, passiamo alla descrizione della nostra nuova libreria, che si può scaricare gratuitamente dal sito www.fishino.it, oppure dalla repository GitHub [www.github.com/signorettaefishetta](https://github.com/signorettaefishetta).

Per comprendere meglio Fishetta richiamiamo alcuni concetti base di Pushetta: si tratta di una web-app tutta italiana che consente di inviare notifiche push a più dispositivi contemporaneamente: smartphone, tablet, smart TV, ecc. I vantaggi rispetto ai tradizionali mezzi di comunicazione (e-mail, messaggi) sono numerosi: tracciabilità, gratuità e istantaneità. Non è necessario inserire alcun numero di telefono e accedendo alla dashboard possiamo vedere le statistiche come numero di utenti che hanno sottoscritto il canale e numero delle notifiche inviate. Possiamo anche variare le impostazioni del canale: ad esempio nascondere in modo che altri non possano sottoscrivere o renderlo accessibile al pubblico.

COME FUNZIONA

Per utilizzare Fishetta nella nostra applicazione dobbiamo aggiungere nello sketch corrispondente la riga `#include<Fishetta.h>` e dichiarare due

costanti: il nome del canale Pushetta e la chiave API, che è univoca per ogni utente. Dichiariamo successivamente l'oggetto della libreria in modo tale da associare anche le due costanti. L'oggetto dichiarato è così strutturato:

Fishetta push (CHANNEL,APIKEY);

per inviare la notifica push non è necessario altro che richiamare la funzione

push.sendPushNotification(String);

Bisogna precisare che questa funzione è del tipo *boolean*, quindi riceviamo un valore "true" in caso di effettivo invio e "false" in caso contrario.

Ma cosa fa la nostra Fishino quando invia una notifica? Ebbene, nel Listato 1 possiamo vedere il codice (dal file *Fishetta.cpp*) che servirà per effettuare la richiesta HTTP GET.

Fishetta risulta utilissima perché non richiede molte risorse e non deve instaurare una connessione HTTPS con il server, riducendo all'osso l'uso della RAM del modulo WiFi ESP2866 installato sulle board Fishino.

LA NOSTRA APPLICAZIONE

Nel fascicolo n° 217 della nostra rivista abbiamo descritto il sensore AS3935, utile per rilevare fulmini

Fig. 1 - Pagina di registrazione al sito web di Pushetta.

Schema di
cablaggio di Fishino e della
breakout board rilevatrice
di fulmini.

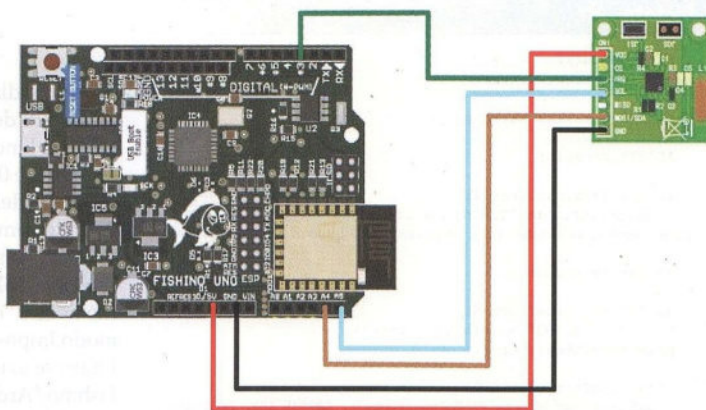


Tabella 1 - Connessioni del circuito.

Fishino	AS3935
GND	GND
5V oppure 3,3V	Vcc
D3	IRQ
A4 (SDA)	SDA
A5 (SCL)	SCL

Listato 2

```
#define AS3935_ADDRESS 0x03
#define IRQ_PIN 3
FishinoAS3935 AS3935(AS3935_ADDRESS, CS_PIN, digitalPinToInterrupt(IRQ_PIN));
Fishetta push(CHANNEL, APIKEY);
```

fino a 40 km di distanza. In questo articolo andremo ad associare ad un rilevamento l'invio di una notifica push. Dobbiamo innanzitutto precisare che si è resa necessaria la scrittura di un'altra libreria per poter utilizzare il sensore in I²C-Bus. Quelle reperibili in rete, infatti, si sono dimostrate incomplete. La libreria FishinoAS3935 è scaricabile dal sito di

Fishino e consente di utilizzare il sensore con SPI, o I²C.

Dunque, analizziamo insieme l'hardware e lo sketch che ne permette il funzionamento.

Creazione del canale

Per creare un canale è necessario recarsi al sito www.pushetta.com registrarsi, scegliere uno username e una password e inserire il proprio indirizzo e-mail (Fig. 1). Dopo esserci registrati, entriamo nell'area riservata e andiamo nella sezione "Channels", successivamente premiamo su "Add a channel". Compare una schermata come quella nella Fig. 2; a questo punto non ci resta che compilare tutti i campi, inserire un'immagine che identifichi ciascun canale e premere su "Create". Andiamo nella sezione "Dashboard" e annotiamo la chiave API che ci servirà successivamente.

L'hardware necessario

Per realizzare la nostra applicazione dobbiamo interconnettere la breakout board sensore di fulmini descritta nel fascicolo n° 217 secondo lo schema di cablaggio mostrato in cima a questa pagina. Da esso vediamo che il pin Vcc del sensore è collegato al 5V della nostra Fishino Uno (in teoria è possibile alimentarlo anche a 3,3V). I pin GND, SDA e SCL del sensore sono collegati ai reciproci della board. Il pin IRQ della breakout board basata su AS3935 è collegato al pin D3 del Fishino; questo servirà a generare il segnale di interrupt ogni qualvolta verrà rilevato

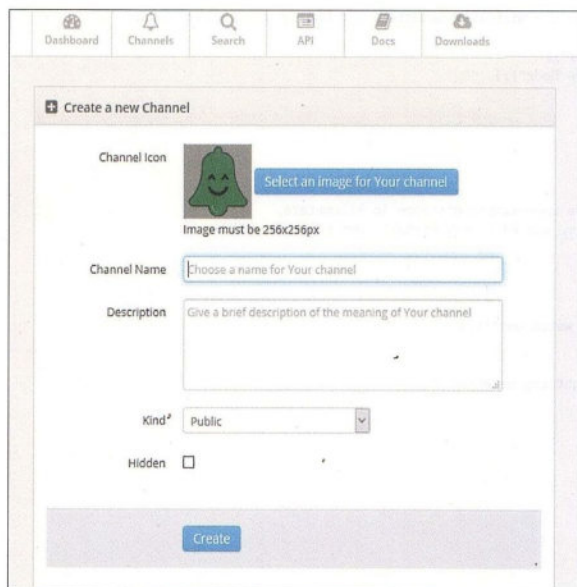


Fig. 2 - Creazione nuovo canale.

Listato 3

```
Wire.begin();
Wire.setClock(400000);

Serial.println("Starting");

AS3935.reset();

if (!AS3935.calibrate())
  Serial.println("Tuning out of range, check your wiring,
your sensor and make sure physics laws have not changed!");

AS3935.setIndoors();

AS3935.setNoiseFloor(6);
  AS3935.enableDisturbers();
printAS3935Registers();

AS3935IrqTriggered = 0;
attachInterrupt(digitalPinToInterrupt(3), AS3935_ISR, RISING);
```

un fulmine o un disturbo radioelettrico consistente. Tutto ciò che serve a cablare le due unità è riassunto nella Tabella 1.

LO SKETCH ARDUINO

Passiamo dunque ad analizzare il codice dell'applicazione proposta, che riportiamo nel Listato 2; come

vedete, andiamo a definire alcune costanti e a creare gli oggetti delle nostre librerie: `AS3935_ADDRESS` identifica l'indirizzo I²C-Bus del modulo AS3935, che di norma è 0x03; `IRQ_PIN` è il pin che va collegato al pin D3 della scheda. Non utilizzeremo il pin D5 definito come `SI_PIN`, perché il suo collegamento si rende necessario solo in caso si voglia utilizzare il sensore in SPI. Definendo l'oggetto AS3935 possiamo notare "*digitalPinToInterrupt(IRQ_PIN)*": in tal modo impostiamo il pin D3 come pin di interrupt. Infatti se avessimo ommesso questa parte, la nostra Fishino/Arduino non sarebbe in grado di riconoscere l'interrupt. Il codice riportato nel Listato 3 è incluso nella setup dello sketch e serve ad instaurare la connessione con il sensore, eseguire il reset dei registri, calibrarlo e selezionare le modalità di lavoro. Nel caso in cui il sensore sia installato in un luogo chiuso non è necessario apportare alcuna modifica al codice. Se invece volessimo utilizzarlo all'esterno, dovremmo modificare `AS3935.setIndoors()`; in `AS3935.setOutdoors()`; in tal modo il sensore regolerà in automatico il guadagno. Passiamo ora

Listato 4

```
if (AS3935IrqTriggered)
{
  // reset the flag
  AS3935IrqTriggered = 0;
  // first step is to find out what caused interrupt
  // as soon as we read interrupt cause register, irq pin goes low
  int irqSource = AS3935.getInterruptSource();
  // returned value is bitmap field, bit 0 - noise level too high, bit 2 - disturber detected, and finally bit 3 - lightning!
  if (irqSource & 0b0001){
    Serial.println("Noise level too high, try adjusting noise floor");
    push.sendPushNotification("Noise level too high, try adjusting noise floor");
  }
  if (irqSource & 0b0100){
    Serial.println("Disturber detected");
    // push.sendPushNotification("Disturber detected");
  }
  if (irqSource & 0b1000)
  {
    // need to find how far that lightning stroke, function returns approximate distance in kilometers,
    // where value 1 represents storm in detector's near vicinity, and 63 - very distant, out of range stroke
    // everything in between is just distance in kilometers
    int strokeDistance = AS3935.getLightningDistanceKm();
    if (strokeDistance == 1){
      Serial.println("Storm overhead, watch out!");
      push.sendPushNotification("Storm overhead, watch out!");
    }
    if (strokeDistance == 63){
      Serial.println("Out of range lightning detected.");
      push.sendPushNotification("Out of range lightning detected.");
    }
    if (strokeDistance < 63 && strokeDistance > 1)
    {
      String stroke = "Lightning detected ";
      stroke += (strokeDistance, DEC);
      stroke += " kilometers away.";
      Serial.println(stroke);
      push.sendPushNotification(stroke);
    }
  }
}
```

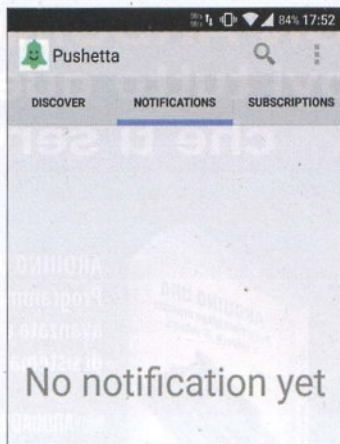



Fig. 3 - Schermata al primo avvio dell'applicazione.

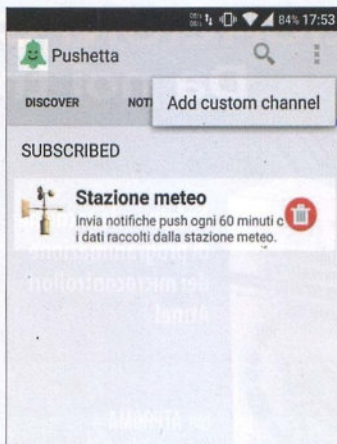


Fig. 4 - Registrazione ad un canale.

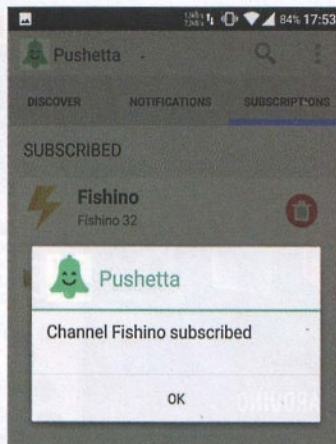


Fig. 5 - Il pop-up al termine della registrazione ad un canale.

ad analizzare il codice contenuto nel loop del nostro sketch, che potete vedere nel **Listato 4**. All'avvio di ogni loop, la scheda Fishino controlla che non ci sia un cambiamento di stato sul pin di interrupt, per poi riportarne il valore a zero. Andrà poi a leggere il valore sul registro del sensore in modo da determinare cosa sia stato a determinare questo cambiamento. Il pin IRQ arriverà ad un livello logico basso. In base al valore ottenuto da registro possiamo determinare se è stato rilevato un fulmine, un disturbo o se il livello del rumore è troppo alto. In caso dovessimo rilevare un fulmine, non dovremmo fare altro che stabilire la distanza in cui è scoccato: se il valore è 1, riceveremo una notifica che ci avverte di prestare attenzione e che il fulmine è sopra le nostre teste. Se il valore è compreso tra 1 e 63, ci viene inviata una notifica che ci indica la distanza dal fulmine. Se il valore è maggiore di 63, il fulmine è fuori portata, ma ne saremo comunque informati.

CONFIGURAZIONE DELL'APP

Passiamo adesso all'impostazione dell'app: prendiamo lo smartphone e dopo aver scaricato l'app "Pushetta" dal Play Store o da iTunes, configuriamola per ricevere le notifiche push. Al primo avvio ci comparirà la schermata di **Fig. 3**; dopo esserci spostati nella sezione "Subscriptions", premiamo sul pulsante in alto a destra nello schermo e poi su "Add custom channel" (**Fig. 4**). Ora inseriamo il nome del nostro canale e premiamo Invio sulla tastiera. Se tutto è andato a buon fine, ci comparirà un pop-up come quello mostrato nella **Fig. 5**. A questo punto il nostro dispositivo è pronto a ricevere le notifiche dal canale al quale si è sottoscritto. La **Fig. 6** mostra una schermata dell'app che riporta le letture dei fulmini.

CONCLUSIONI

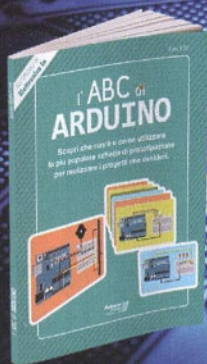
Termina qui la nostra presentazione della nuova libreria per le board Fishino; ora non dovete fare altro che sperimentare per semplificare e rendere la vostra vita sempre più connessa in rete. ■



Fig. 6 - Schermata di lavoro dell'app.

VUOI SVILUPPARE LE TUE APPLICAZIONI CON ARDUINO?

Da noi trovi tutto quello
che ti serve!



L'ABC di ARDUINO

COD. ABCARDU

€ 9,90



Arduino e le tecniche
di programmazione
dei microcontrollori
Atmel

COD. ATPROMA

€ 15,00



ARDUINO UNO
Programmazione
avanzata e librerie
di sistema

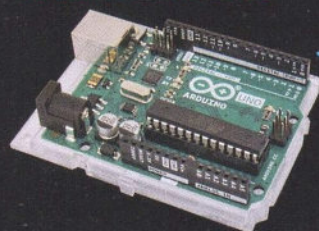
COD. ARDUADVANCED

€ 22,00

Arduino Uno Rev3

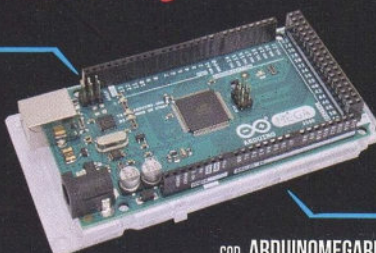
Arduino Mega

Arduino Due



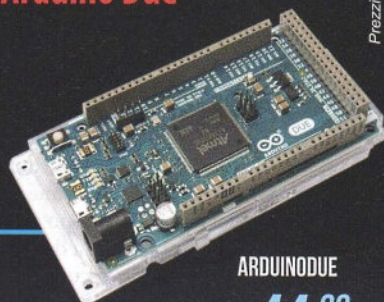
COD. ARDUINOUNOREV3

€ 24,50



COD. ARDUINOMEGAREV3

€ 43,00



ARDUINODUE

€ 44,00

Prezzi IVA inclusa

kit V5 con Arduino Uno REV3

Set contenente tutti i componenti
necessari per realizzare gli esperimenti
descritti nel libro
"L'ABC di Arduino".



COD. ARDUKITV5

€ 59,00

Starter kit per robot con Arduino Uno

Set contenente le parti necessarie per realizzare il Robot
descritto nel libro incluso.

€ 155,00



COD. ARDUKITV4

**FUTURA
ELETTRONICA®**
www.futurashop.it

Futura Group srl • via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

Caratteristiche tecniche di questi prodotti
e acquisti on-line su www.futurashop.it

ANTENNINO: L'ARDUINO CON L'ANTENNA

Scopriamo
come caricare
il bootloader
nella nostra
scheda per IoT,
preparandola
all'utilizzo.
Seconda e
ultima puntata.



dell'Ing. MIRKO UGOLINI

Avete conosciuto Antennino leggendo la prima puntata di questo articolo, nella quale abbiamo presentato il progetto e descritto l'hardware nei dettagli. Riprendiamo quindi da dove ci siamo fermati, ossia dalla programmazione del suo microcontrollore allo scopo di caricare il bootloader. Questo perché, come tutte le board basate su architettura Arduino, Antennino prevede nell'ATmega un bootloader, che è un firmware eseguito nella fase di bootstrap (avvio) del microcontrollore e che permette di eseguire alcune operazioni preliminari, la più importante delle quali è gestire il caricamento dei nostri sketch tramite l'IDE di Arduino impiegando un economico convertitore USB/seriale. Esistono tanti bootloader per Arduino: potreste anche caricare quello classico che viene impiegato normalmente in Arduino Uno o anche scegliere quello del progetto Moteino realizzato da Felix Ruso, che riteniamo più interessante perché ha la possibilità di gestire l'aggiornamento wireless

del firmware. Per fare questo, Ruso ha dovuto aggiungere del codice, quindi la dimensione del bootloader è passata da 512 byte a 1 kB, cosa che riduce lo spazio nella Flash memory utilizzabile dagli sketch; questa rinuncia viene, però, ripagata largamente dai benefici della programmazione da remoto. Ruso ha anche personalizzato la libreria del modulo RFM69.

CARICHIAMO IL BOOTLOADER

Vediamo subito cosa ci serve per caricare il bootloader sul nostro Antennino: la programmazione richiede un programmatore con interfaccia ICSP (In Circuit Serial Programming, ossia che permette la programmazione del chip montato nel circuito dove viene impiegato) ed un software che lo gestisca; infatti non basta il semplice convertitore USB/seriale, che si usa quando nell'ATmega è già installato il bootloader.

Anche qui abbiamo diverse alternative, una delle

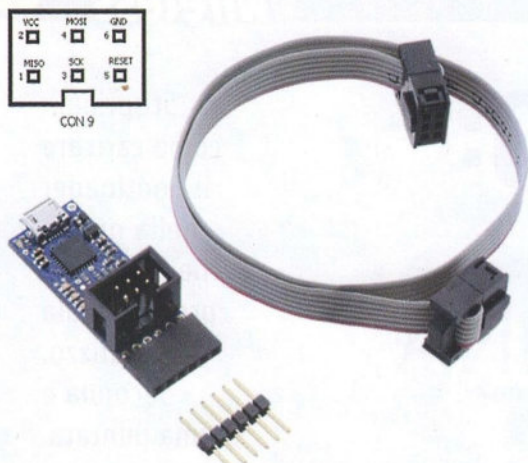


Fig. 1 - Connettore e cavo di programmazione ICSP usato per caricare il BootLoader.

quali è la possibilità di impiegare un semplice Arduino Uno come programmatore di altri Arduino, sul quale deve essere caricato uno *sketch* specifico e ci serve un cavo piatto a sei poli terminante con un connettore avente la piedinatura indicata in Fig. 1. Nel fascicolo n° 164 abbiamo spiegato l'utilizzo di Arduino come ISP con Arduino (ed anche Internet è prodiga di suggerimenti); può anche essere utile il volume Arduino e le tecniche di programma-

zione dei microcontrollori Atmel (acquistabile su www.futurashop.it). Volendo utilizzare un programmatore commerciale, potremmo adottare quello disponibile sul sito di Futura elettronica: <http://www.futurashop.it/programmatore-usb-avr-3-in-1-usbavrprog2> (Fig. 1).

Per quanto riguarda il software anche qui le possibilità sono molteplici. Alcuni utilizzano un software della Atmel (ora Microchip) che si chiama Atmel Studio. Ma è un software pachidermico (soprattutto nelle ultime versioni) e se la vostra necessità è solo quella di caricare il bootloader su Antennino vi consigliamo vivamente di utilizzare un software molto più leggero. La scelta cade forzatamente su Avrdude, un software open source che deve essere impiegato a riga di comando. Naturalmente usarlo a riga di comando non è il massimo della semplicità, quindi quello che vi proponiamo di utilizzare è BitBurner. Il cuore di questo programma è sempre Avrdude, però dotato di GUI (interfaccia grafica) che ne agevola enormemente l'utilizzo. Il software è scaricabile da questo indirizzo: <https://sourceforge.net/projects/bitburner>. Il programma è essenziale: si presenta con una sola interfaccia con quattro tab con cui è possibile fare tutto. Ricordatevi che il programmatore ICSP deve essere settato per lavorare a 3,3 V. Di seguito analizziamo i passi della programmazione con BitBurner.

COLLEGARE IL PROGRAMMATORE E INDIVIDUARE IL MICRO

Per evitare conflitti con il bus SPI, dobbiamo attivare il resistore di pull-up collegato al pin di Chip-Select sul modulo RFM69 chiudendo il ponticello J9. Rimuoviamo (se presenti) le batterie da Antennino, colleghiamo il programmatore ISP alla porta USB del nostro PC, inseriamo il connettore ICSP femmina del programmatore alla porta ICSP maschio di Antennino facendo attenzione al verso di inserimento, quindi mandiamo in esecuzione BitBurner.

Selezioniamo il primo TAB (AVRdude) e nella schermata che si apre (Fig. 2) selezioniamo il nostro Chip (ATmega328P) dalla combo AVR Device; dalla combo Programmer, scegliamo il nostro programmatore (ad esempio Atmel AVR ISP mkII) e la porta di comunicazione (USB).

Ora viene la parte più delicata, quindi seguite passo-passo le varie fasi. Andiamo sul Tab "AVR Fuses" di Fig. 3 e clicchiamo sul pulsante "Read": con questa operazione andremo a leggere la configurazione attuale (predefinita) dei vari Fuse Bit del nostro microcontrollore. Ci sono moltissimi para-

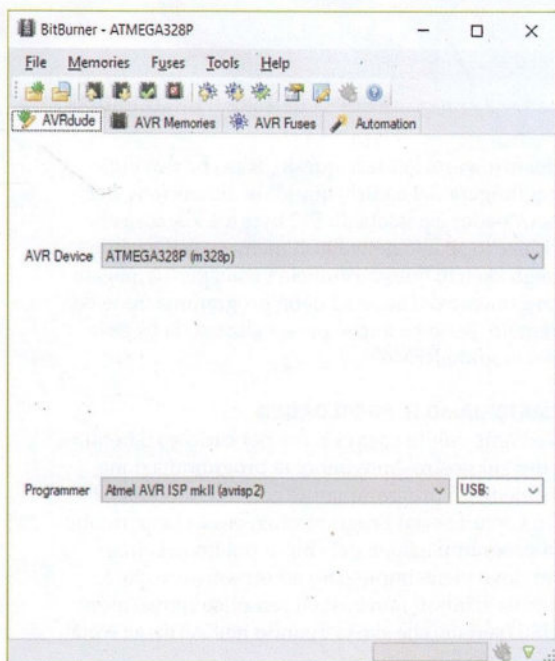


Fig. 2 - Tab AVRdude di BitBurner.

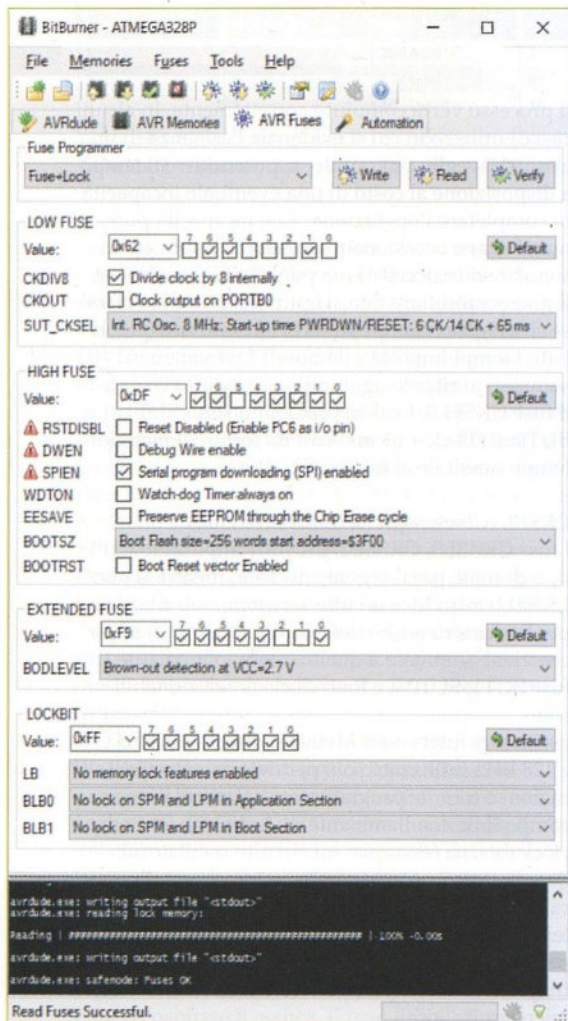


Fig. 3 - Tab AVRFuses predefinita di BitBurner.

metri, ma non fatevi spaventare. Poi spiegheremo nel dettaglio le varie opzioni.

Se tutto funziona come previsto dovrebbe apparire un messaggio nel box di log, dove, a conferma che bitBurner utilizzi AVRDUDE, viene visualizzato lo script di AVRDUDE che viene invocato ed il relativo risultato. Se viene riportato "Fuses OK" "Read Fuses Successful" significa che il nostro chip (Device signature 0x1e950f) è stato individuato in modo corretto.

PROGRAMMARE I FUSE BIT

I fuse bit sono particolari registri del microcontrollore che permettono di configurare il comportamento di alcuni aspetti dello stesso. Per esempio consentono di definire quale sorgente deve essere

utilizzata per il clock di sistema: l'oscillatore interno, una sorgente di clock esterna, oppure un quarzo o risuonatore esterno, la porzione di memoria riservata al bootloader, il livello minimo di tensione di alimentazione che permette di fare partire il micro ed altre impostazioni essenziali per il corretto funzionamento del microcontrollore.

Questa fase è molto delicata perché se sbagliamo a settare i valori dei Fuse Bit potremmo trovarci con un ATmega328P che non ne vorrà più sapere di partire; ad onor del vero esiste una procedura di sblocco ma richiede del lavoro supplementare.

Un errore molto comune è quello di impostare il micro "dicendogli" di utilizzare come sorgente di clock una sorgente esterna e quindi il chip all'avvio si attende che su un certo pin sia presente un clock con una certa frequenza; e se non gli forniamo tale segnale, il nostro caro ATmega328p non si scomoderà proprio a partire.

Bene, fatta questa piccola premessa utile a rendervi consapevoli della delicatezza e importanza dei fuse bit, mettiamoci al lavoro.

I fuse sono suddivisi in quattro blocchi: "LOW FUSE", "HIGH FUSE", "EXTENDED FUSE", "LOCK BIT". Questi registri sono detti Fuse, "Fusibili", anche se in pratica non c'è nulla che venga bruciato materialmente nella memoria del microcontrollore: sono detti così perché essi sono salvati nella Flash, quindi nella memoria non volatile del microcontrollore: essi perciò non perdono il loro valore quando viene tolta l'alimentazione. In pratica si tratta di quattro registri da un Byte (8bit) ciascuno. Per definirne lo stato di questi registri possiamo usare la notazione binaria oppure quella esadecimale. Quando diciamo o ci riferiamo al fatto che un valore sia settato/non settato, programmato/non programmato, bisogna prestare attenzione al fatto che 1 significa **non settato** (non programmato) mentre 0 significa **settato** (programmato) quindi, in un certo senso la logica è invertita. Questo modo di operare apparentemente illogico

Bit	Nome	Note
7	CKDIV8	Quando settato divide il clock per 8
6	CKOUT	Quando settato il clock è presente sul pin CLK0 PBO
5	SUT1	Imposta il tempo di avvio del microcontrollore
4	SUT0	
3	CKSEL3	Seleziona la fonte del clock
2	CKSEL2	
1	CKSEL1	
0	CKSEL0	

Tabella 1 - I Low Fuse.

trae origine dall'architettura interna delle memorie Flash, le cui celle dopo che sono state cancellate (formattate) hanno tutte il valore 1. Questo spiega la strana logica usata dai fuse per memorizzare i valori.

Low Fuse

Iniziamo con una carrellata sui Low Fuse, riportati sinteticamente nella **Tabella 1**, e con la loro impostazione.

CHDIV8 (Clock Divide): in alcuni casi si rende necessario abbassare il clock di sistema dividendone il valore per 8. Per esempio se scegliessimo come sorgente di clock quella interna a 8MHz e il flag CKOUT fosse settato, avremmo un clock di sistema a 1MHz. Il valore predefinito è CKOUT=0 (settato).

CKOUT (Clock Out): questa impostazione permette di fare apparire il segnale di clock di sistema sul Pin PB0 permettendo di pilotare altri circuiti. Il valore predefinito è CKOUT=1 (non settato).

SUT1/SUT0 (Start Up Time): questi due fuse impostano il tempo di avvio del microcontrollore dopo un reset oppure dopo l'uscita da una condizione di sleep (risparmio energetico). Quando viene fornita alimentazione ad un circuito oscillatore, questo necessita di un certo periodo di tempo nonché di un certo numero di oscillazioni prima che il clock fornito sia stabile ed il suo valore corrispondente a quello nominale. Analogamente, il ripristino da una condizione di stand-by, introduce le stesse problematiche perché il microcontrollore richiede un certo tempo e alcuni numeri di cicli di clock per riportare a piena operatività tutte le periferiche interne. Le combinazioni dei due fuse SUT1 e SUT0 determinano l'intervallo da attendere prima che il processo di avvio/ripresa si possa considerare completato (in forma di un'attesa in ms e di un'attesa in cicli di clock): più tale intervallo è lungo, più tempo quindi viene concesso al microcontrollore per avviarsi/risvegliarsi, più saremo sicuri che

il processo verrà compiuto correttamente. In alcuni casi di utilizzo in cui si evidenzia l'esigenza di un avvio/risveglio più rapido, si può ridurre il tempo a disposizione al costo di una eventuale incapacità di completare l'operazione. Tale incapacità può manifestarsi occasionalmente se i valori di avvio sono brevi (valori 0-1) ma può capitare anche che il microcontrollore non sia più in grado di avviarsi se si scelgono i tempi più brevi in assoluto (valori 0-0). I tempi impostati da questi fuse vengono sommati al ritardo aggiuntivo introdotto tramite il fuse CKSEL0. I valori predefiniti sono SUT1=1 e SUT0=0 (14 ck + 65 ms) così da fornire il maggior tempo possibile al microcontrollore.

CKSEL (Clock Sources/Clock Selection)

I fuse CKSEL3, CKSEL2, CKSEL1 impostano il tipo di fonte per il segnale di clock, mentre il fuse CKSEL0 introduce un ulteriore tempo di ritardo per l'avvio/risveglio del microcontrollore, tempo che viene sommato a quello ottenuto mediante i fuse SUT1, SUT0. Le fonti di clock selezionabili sono:

oscillatore interno a 8 MHz; oscillatore interno a 128 kHz (utilizzato solo per determinate applicazioni e tramite programmatori particolari, non compatibile con l'ambiente di Arduino); fonte di clock esterna (esempio: un circuito oscillatore); quarzo o risonatore ceramico esterno (con diversi campi di frequenza). Le combinazioni sono molteplici: vengono presi in considerazione diversi parametri, tra cui il consumo del quarzo e la sua frequenza, tutte riportate sulla scheda tecnica del microcontrollore.

Nota: nel programma BitBurner, per agevolare l'utente la configurazione relativa alla frequenza del clock e ai ritardi di startup si selezionano dalla combo box **SUT_CKSEL** in cui sono elencate tutte le possibili combinazioni.

High Fuse

Passiamo adesso all'impostazione degli High Fuse, che sono descritti nella **Tabella 2**.

RSTDISBL (External reset disable): il pin PC6 è deputato al reset del micro. Portando tale pin a massa si ottiene il reset del microcontrollore e si può reinizializzare il funzionamento se qualche cosa va storto nel codice. In qualche caso particolare può essere necessario usare il pin PC6 come un normale Pin di I/O e quindi in questa eventualità il Flag RSTDISBL deve essere settato (=0). Attenzione però che con questa impostazione si perde la possibilità di riprogrammare il chip. Un motivo

Bit	Nome	Note
7	RSTDISBL	Reset Esterno disabilitato
6	DWEN	Debug Wire abilitato
5	SPIEN	SIP e data Download Abilitato
4	WDTON	WatchDog sempre Abilitato
3	EESAVE	Previene la cancellazione della EEPROM
2	BOOTSZ1	Setta la Dimensione di memoria del Boot Loader
1	BOOTSZ0	
0	BOOTRST	Setta il vettore di reset del Boot Loader

Tabella 2 - Gli High Fuse degli ATmega.

BOOTSZ1	BOOTSZ0	Boot Size	Pages	Application Start	Application End	Reset Address
1	1	256 word	4	0X0000	0X3EFF	0X3F00
1	0	512 word	8	0X0000	0X3DFF	0X3E00
0	1	1024 word	16	0X0000	0X3BFF	0X3C00
0	0	2048 word	32	0X0000	0X37FF	0X3800

Tabella 3 - Possibili valori per la dimensione del Bootloader.

per disabilitare il reset potrebbe anche essere la necessità di impedire la riprogrammazione o lo scaricamento del firmware per ragioni di copyright. Valore predefinito (RSTDISBL=1) non settato.

DWEN (debugWIRE enable): abilitando questo fuse si accede alla possibilità di eseguire quella che viene indicata con on-chip debuggin, ma è richiesto un particolare programmatore (tipo AVR Dragon). Quindi anche questo Flag non deve essere alterato rispetto al valore predefinito altrimenti si perde il controllo sulla possibilità di riprogrammare il chip.

SPIEN (Enable Serial programming and Data Downloading): come sappiamo, i chip Atmega possono essere programmati tramite l'interfaccia ISP. Se disabilitiamo questa opzione non saremo più in grado di farlo e quindi è necessario lasciare questa impostazione settata. Il valore predefinito è SPIEN=0 (settato).

WDTON (Watchdog Timer Always On): il WatchDog è in linea di principio un timer che provoca il reset del controllore se non riceve un segnale di conferma del funzionamento corretto del chip entro uno specifico intervallo di tempo. Il Watchdog può inoltre essere settato via software. Il valore predefinito WDTON =1 (non settato).

EESAVE (Preserve EEPROM memory): quando i chip Atmega sono programmati la memoria eeprom è cancellata come lo è la memoria destinata ad ospitare il codice. Il fuse EESAVE può essere usato per notificare al chip di preservare la EEPROM. Questo può essere utile quando si vuole aggiornare il firmware ma si desidera mantenere le configurazioni memorizzate nella EEPROM. Siccome l'impostazione predefinita è EESAVE=0, la EEPROM viene cancellata durante l'aggiornamento del firmware.

BOOTSZ1 & BOOTSZ0 (Boot loader Size): il bootloader è utilizzato per inizializzare i dispositivi e verificare le comunicazioni esterne. In generale, Arduino usa un bootloader per comunicare con un PC e capire se deve caricare uno sketch. Il bootloader è memorizzato nella stessa memoria in cui è caricato il nostro sketch e siccome il bootloader può avere diverse dimensioni, noi possiamo e dobbiamo dire al chip quanto spazio riservare per il bootloader. Notate che le dimensioni specificate sono in word (1 word=2 byte) quindi 512 word=1.024 byte.

BOOTRST Select reset vector: quando questo fuse

è settato, nella fase di reset, l'esecuzione del codice inizierà dalla posizione iniziale del bootloader altrimenti se il fuse non è settato il programma inizierà dalla locazione di memoria di partenza del programma 0x0000. Quindi anche se abbiamo configurato i fuse **BOOTSZ1 & BOOTSZ0** non settando il fuse **BOOTRST** la presenza del bootloader verrà ignorata. Il valore predefinito è **BOOTRST=1** (non settato).

Extended Fuse

Procediamo con gli Extended Fuse, riportati nella **Tabella 4** e vediamo che gli unici significativi sono i BODLEVEL0, 1 e 2, meglio descritti nella **Tabella 5**. Questi Fuse (sul chip Atmega 328p) sono utilizzati solamente per settare il livello del BOD (Brown-Out Detector). Vediamo di cosa si tratta: i chip Atmega possono diventare instabili o inaffidabili quando la tensione di alimentazione non è sufficiente. Per esempio un 328P può lavorare in modo corretto a 16MHz se la sua tensione di alimentazione raggiunge almeno i 4V. Per valori di tensione sotto i 4V il chip ha una alta probabilità di comportarsi in modo imprevedibile. Per assicurarsi che il chip abbia una tensione adeguata per svolgere il suo lavoro, viene in aiuto il registro relativo ai BODLEVEL con il quale è appunto possibile specificare la tensione minima di lavoro del chip. La tensione viene monitorata da un blocco operativo

Bit	Nome	Note
7		Non usato/riservato
6		Non usato/riservato
5		Non usato/riservato
4		Non usato/riservato
3		Non usato/riservato
2	BODLEVEL2	Setta il livello di tensione del rilevatore di Brown-out
1	BODLEVEL1	
0	BODLEVEL0	

Tabella 4 - Gli Extended fuse.

BODLEVEL2,1,0	Tensione
111	BOD Disabled
110	1,8V
101	2,7V
100	4,3V

Tabella 5 - Possibili valori del BODLEVEL.

LOW Byte Fuses		B01100010 = 0x62			
Bit	Nome	Descrizione	Valore		Note
7	CKDIV8	Divide clock per 8	0	Settato	14 ck + 65 ms
6	CKOUT	Clock su pin PB0	1	Non settato	
5	SUT1	Seta il ritardo di startup	1	Non settato	
4	SUT0		0	Settato	
3	CKSEL3	Clock Source	0	Settato	Clock interno 8MHz
2	CKSEL2		0	Settato	
1	CKSEL1		1	Non settato	
0	CKSEL0		0	Settato	

Tabella 6 - Valori predefiniti per i Low Fuse.

interno al chip stesso. Se la tensione scende sotto il valore prestabilito il chip viene resettato. Il valore predefinito è BODLEVEL2, BODLEVEL1, BODLEVEL0 (=1 non settato). A causa del fatto che alcuni bit non sono utilizzati, il loro valore può risultare "fluttuante" per cui dopo la programmazione del fuse esteso la rilettura del registro può dare un errore di verifica del valore scritto in memoria. In questo caso si consiglia di considerare solo il valore dei primi 3 bit e di impostare a 0 i bit dal 3° al 7°.

Lock Bit

Le impostazioni di questo registro permettono di disabilitare la possibilità di leggere o scrivere nella memoria ed impedirne quindi la riprogrammazione. Questo registro è molto delicato e vi consigliamo di lasciarne le impostazioni inalterate (0x3F) altrimenti correte il serio rischio di ritrovarvi con chip completamente bloccato.

I valori predefiniti

Ciò detto, diamo uno sguardo all'impostazione predefinita dei Fuse Bit nei microcontrollori ATmega 328/328P vergini, ovvero come sono impostati in fabbrica. La **Tabella 6** riporta le impostazioni predefinite per i Low Fuse, la **Tabella 7** riguarda gli High Fuse e la **Tabella 8** riporta i valori predefiniti degli Extended Fuse. Su BitBurner Individuiamo la sezione "Fuses & Lock bits" e clicchiamo sui due pulsanti Read. Questa operazione permette

di acquisire i valori presenti in questi "registri" e probabilmente non saranno quelli che servono a noi. Andiamo quindi a scrivere i valori seguenti: L:0xDE H:0xDC E:0x05 LB:0x3F e clicchiamo sui rispettivi pulsanti Write. Nell'Area di log sottostante dovremmo avere dei messaggi. Al termine proviamo a verificare i dati scritti e clicchiamo nuovamente su Read: se tutto è andato a buon fine dovremmo ritrovare i valori che abbiamo inserito noi e nessun messaggio di errore nella finestra di log.

SCRIVIAMO IL BOOTLOADER SUL CHIP

Scarichiamo i file dal repository GitHub accessibile da <https://github.com/LowPowerLab/DualOptiboot> cliccando sul pulsante verde "Clone or Download" come visibile in Fig. 2.

Scarichiamo lo zip **DualOptiboot-master.zip**, quindi scompattiamolo in una cartella di appoggio: ad esempio C:\TMP\. Quello che ci interessa è il file che si trova in questo percorso:

```
C:\TMP\DualOptiboot-master\DualOptiboot_V5.0_
atmega328_BlinkD9.hex
```

Il file indicato è il bootloader adatto al nostro microcontrollore ATmega328p, che prevede il LED di notifica settato sul pin di I/O D9.

Su BitBurner individuiamo il Tab con l'indicazione "AVR Memories" (Fig. 3), selezioniamo il file del bootloader da caricare, poi nella combo box "Memory Programmer" selezioniamo "Flash Only" e clicchiamo sul pulsante "Write".

A questo punto nella finestra di Log dovremmo veder apparire una barra di progressione che indica la percentuale di caricamento ed al termine dell'operazione, un messaggio rassicurante che ci indica che il caricamento è andato a buon fine.

Se tutto va così, siamo a posto: il nostro Antennino ora ha un cuore pulsante ed è pronto ad ospitare i nostri Sketch.

HIGH Byte Fuses		B11011001 = 0xDF			
Bit	Nome	Descrizione	Valore		Note
7	RSTDISBL	Reset Esterno disabilitato	1	Non settato	
6	DWEN	Debug Wire abilitato	1	Non settato	
5	SPIEN	SIP abilitato	0	Settato	Permette programmazione SIP
4	WDTON	Watdog Timer ON	1	Non settato	
3	EESAVE	Preserva eeprom	1	Non settato	Cancella eeprom quando carico programma
2	BOOTSZ1	Bootloader memory size	1	Non settato	256 Word = 512 KB
1	BOOTSZ1		1	Non settato	
0	BOOTRST	Boot Loader reset Vector	1	Non settato	

Tabella 7 - Valori predefiniti degli High Fuse Fuses.

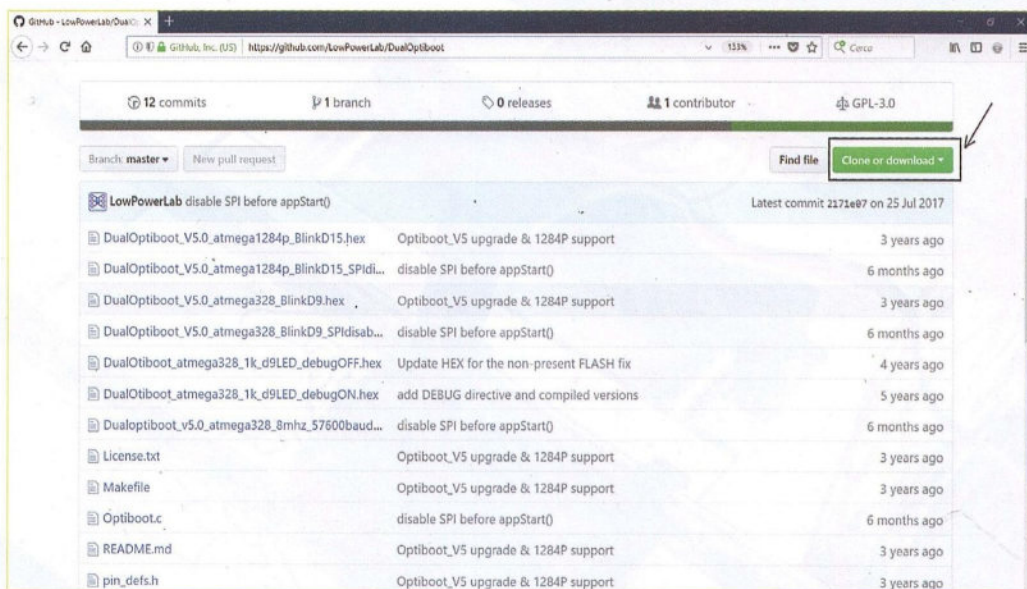


Fig. 2 - Pagina Github del Progetto Moteino.

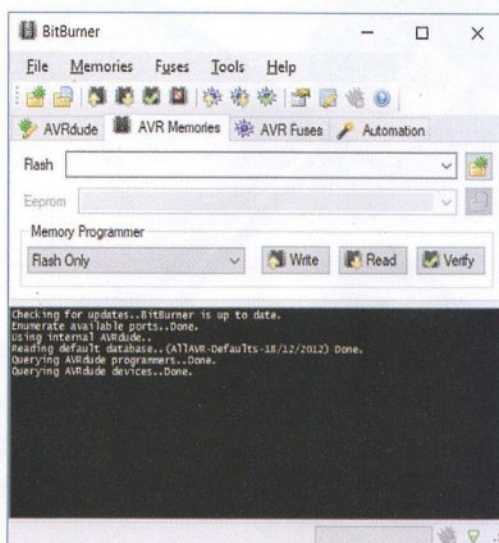


Fig. 3 - Tab AVRMemories in BitBurner.

Extended Fuses		B11111001 = 0xF9			
Bit	Nome	Descrizione	Valore		Note
7		Non usato/riservato	1	Non settato	
6		Non usato/riservato	1	Non settato	
5		Non usato/riservato	1	Non settato	
4		Non usato/riservato	1	Non settato	
3		Non usato/riservato	1	Non settato	
2	BODLEVEL2		0	Settato	
1	BODLEVEL1	BOD Level	0	Settato	BOD Level 2,7V
0	BODLEVEL0		1	Non settato	

Tabella 8 - Valori di fabbrica degli Extended Fuse.

CONCLUSIONI

Bene, si chiude qui questa seconda ed ultima puntata di Antennino; abbiamo spiegato che cos'è il bootloader e tutte le sue particolarità, scendendo in ogni dettaglio, quindi siamo passati alla scelta del bootloader, del programma con cui caricarlo nel microcontrollore di Antennino e alla procedura di installazione per fare in modo che Antennino sia gestibile dall'IDE di Arduino attraverso la connessione USB del computer. Ora che avete il bootloader potrete installare i vostri sketch.

In un prossimo articolo inizieremo a programmare Antennino, iniziando con uno sketch Hello World che però utilizzerà due moduli: uno per simulare un nodo sensore ed uno per implementare un semplice gateway.

Dunque, restate "connessi" e scoprirete nuovi ed utili sviluppi del progetto.



per il MATERIALE

La scheda Antennino (cod. ANTENNINO) è in vendita montata e collaudata presso Futura Elettronica al prezzo di Euro 39,00. È disponibile, separatamente, anche il display OLED I²C da 0,96" (cod. OLED096C) al prezzo di Euro 11,00. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 • Fax: 0331-792287 - www.futurashop.it

EASY PLOTTER

dell' Ing. MIRCO SEGATELLO

Disegna e scrive a tratto mediante un pennarello a punta fine, su fogli di carta ed altro materiale, guidato da una meccanica a due assi autocostruibile con una spesa ridotta.

Come spesso accade, da semplici sperimentazioni amatoriali si giunge ad un prodotto commerciale di larga diffusione: è il caso delle stampanti 3D e delle macchine CNC, facilmente acquistabili ora in kit di montaggio, che in poche di lavoro diventano veri e propri strumenti di lavoro che poco hanno da invidiare a prodotti commerciali ben più costosi. L'utilizzo di software open-source, completamente gratuito, offre lo spunto agli appassionati di sviluppare le proprie personalizzazioni, favorendo maggiormente la diffusione di queste tecnologie. Un giorno ci è capitato tra le mani un set composto da un motore passo-passo e relativo driver di comando, che ci ha prontamente ispirato.

Neanche il tempo di bere un caffè, che già l'idea si era materializzata nella nostra mente: una scheda Arduino, un tondino di acciaio, alcune parti stampate in 3D e ci possiamo autocostruire un plotter a due assi dal costo irrisorio.

Vediamo quindi come procedere.

ASSEMBLAGGIO

La meccanica è senz'altro il punto cruciale di tutto il progetto e normalmente quello che incide maggiormente sui costi di realizzazione. Se ci accontentiamo di realizzare un semplice plotter in cui non sono richiesti particolari sforzi meccanici, come invece avviene in una fresa, possiamo permetterci alcune semplificazioni che ci permettono di abbattere notevolmente i costi.

Ecco come realizzare il plotter: per prima cosa procuratevi un tondino di acciaio rettificato del diametro di 8 mm e ricavate due pezzi da 21 cm e due da 29 cm. Per realizzare i supporti, tra le varie soluzioni, la stampa 3D ci è sembrata la soluzione più economica e idonea, ma se proprio non avete la possibilità di utilizzare una stampante 3D, qualche ora di lavoro di " falegnameria " potrebbe andare ugualmente bene.

La stampa 3D potrebbe comportare alcune leggere bave sui pezzi stampati, che vanno rimosse con cura con un tagliarino o della carta vetrata a grana fine; eventualmente ripassate i fori dove si incastrano i tondini, con una punta da trapano dal diametro di 8,25mm. Il piatto deve scorrere agevolmente ma senza avere dei giochi, normalmente questa funzione è svolta da dei cuscinetti lineari, ma per

Tabella 1 - Materiale occorrente alla realizzazione.

Componente	Codice
Parti meccaniche stampate in 3D	-
Barra rettificata diametro 8mm lunghezza 1m	2828-BARRA8
N°2 Motore passo passo 28BYJ-48 con driver ULN2003	2846-SM2003DB
N°1 Scheda Arduino UNO	7300-ARDUINOUNOREV3
N°1 Servo da modellismo da 9gr	7300-SERVO206
N°1 Alimentatore Switching 5 VDC / 2 A uscita plug	8822-AL5V2A
1m di filo da pesca da 0,5mm di diametro	-
Jumper di collegamento	-
base di supporto 35x25cm	facoltativo

semplificare lasciamo che la guida in acciaio scorra direttamente sulla plastica, tanto le velocità in gioco sono molto basse. Per la movimentazione dei due assi, utilizzeremo due economici motori passo-passo unipolari siglati 28BYJ-48 che vengono forniti assieme al loro driver, mentre il movimento del pennino di stampa è affidato a un comune servo da modellismo. La Tabella 1 riporta le parti occorrenti a realizzare il nostro mini-plotter: la maggior parte si trova presso Futura Elettronica e nella colonna Codice trovate il relativo codice prodotto.

L'assemblaggio è abbastanza intuitivo; potete prendere spunto dalle foto in queste pagine, tenendo pre-

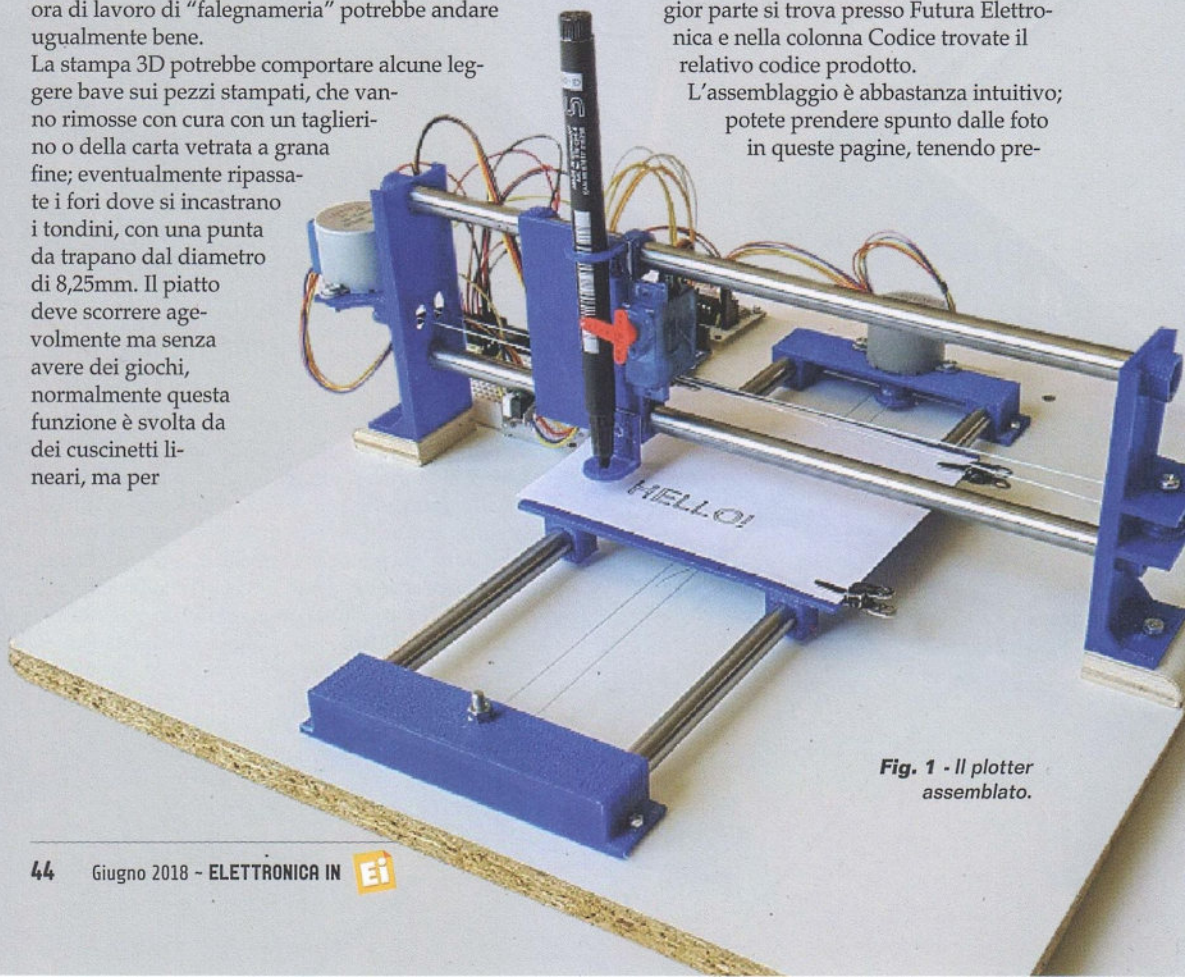


Fig. 1 - Il plotter assemblato.

sente che sarà necessario aggiungere qualche vite e potrebbe essere necessario qualche foro aggiuntivo non previsto in fase di progettazione; al termine potete fissare l'intera struttura su di una base in legno. Il risultato è un'area di stampa di circa 15x15 cm con una risoluzione migliore di 0,01 mm; il tutto, con una spesa decisamente contenuta.

Per ridurre ulteriormente i costi non abbiamo utilizzato le solite pulegge abbinate a una cinghia, bensì del semplicissimo filo da pesca, che, grazie alla sua elasticità, permette la giusta aderenza alle pulegge utilizzate per far spostare il pennino. Usate l'accortezza di avvolgere più volte il filo sulla puleggia, affinché non slitti, come potete vedere nella Fig. 2, che mostra anche in che modo tale filo dev'essere fissato al piatto. Nelle Fig. 3 e Fig. 4 potete vedere il fissaggio relativo ai motori dei due assi.

Per quanto riguarda il pennino, abbiamo previsto due piccoli supporti adatti ad accogliere un classico pennarello indelebile, di quelli usati, ad esempio, per scrivere sui CD/DVD. I due supporti (i relativi file per la stampa 3D, scaricabili dal nostro sito www.elettronica.in.it, sono Pensupport e Pensupport2) una volta stampati, vanno posizionati sul blocco di movimento dell'asse X, con della semplice colla.

Per sollevare ed abbassare il pennino viene utilizzato un comune servo da modellismo, che va fissato con della colla in modo che la squadretta si trovi a ridosso del pennino (Fig. 5). Una piccola molletta sarà sufficiente affinché la squadretta possa alzare ed abbassare il pennino.

Conclusa la meccanica potete procedere con il cablaggio elettrico, che per comodità nel nostro prototipo abbiamo eseguito con dei semplicissimi jumper terminati; seguite lo schema di cablaggio mostrato nella Fig. 6 per interconnettere elettricamente tutti gli elementi.

Diciamo subito che il sistema supporta anche i due fincorsa di zero macchina, che però in questo progetto non utilizziamo. Una piccola attenzione va posta nel corretto verso di avvolgimento del filo nelle pulegge, infatti invertendolo il piatto di muoverà al contrario. Se volete, potete invertire il verso di rotazione elettricamente, semplicemente scambiando i fili IN1 con IN4 e IN2 con IN3 nel driver dei motori. Tutto il circuito funziona a 5V ma l'assorbimento in corrente può raggiungere gli 800mA, pertanto sconsigliamo di alimentare il tutto dalla USB del PC, è invece preferibile alimentare almeno i motori ed il servo con un piccolo alimentatore stabilizzato, e lasciare la sola scheda Arduino alimentata dalla porta USB. Un piccolo alimentatore da 5V (1 o 2 ampere) tipo quelli da cellulare va più che bene, in

Fig. 2 - Fissaggio del filo da pesca sul piatto.

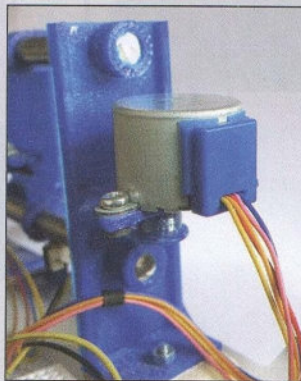
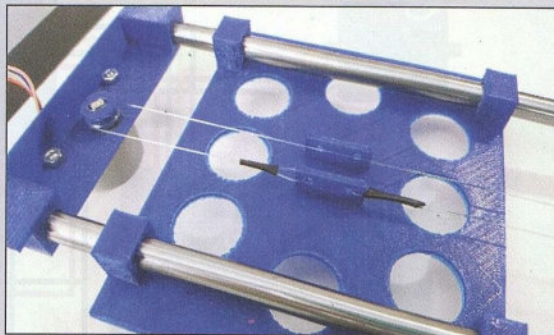


Fig. 3 - Particolare fissaggio motore asse X.

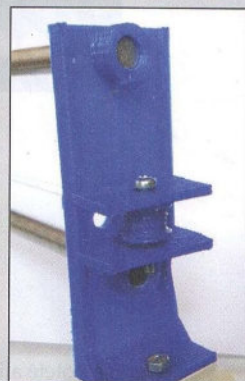


Fig. 4 - Particolare fissaggio puleggia asse X.



Fig. 5 - Particolare fissaggio pennino.

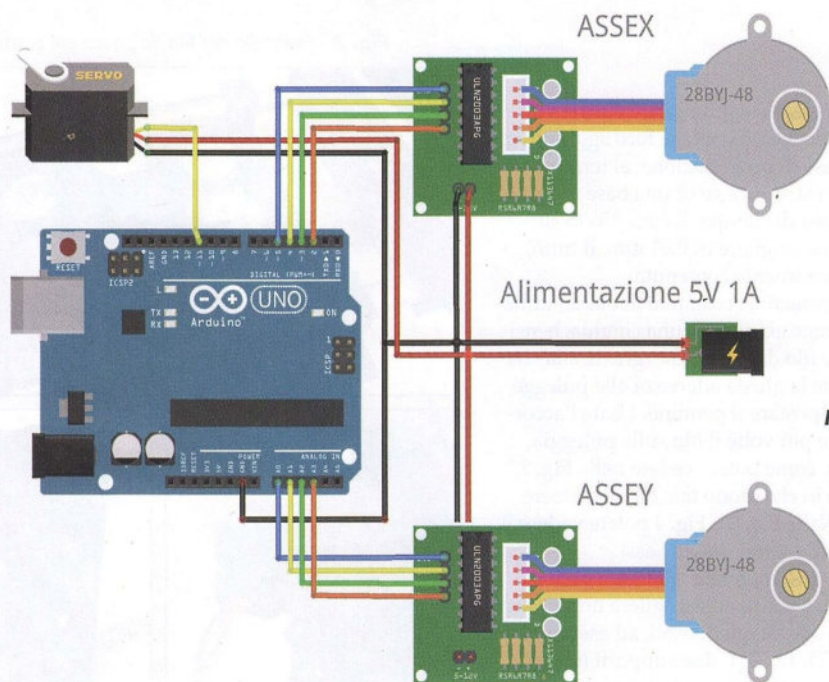


Fig. 6 - Schema di cablaggio.

alternativa potete alimentare il tutto tramite il plug di Arduino ma con un alimentatore switching da 9VDC/0,8A, come quello distribuito dalla Futura Elettronica con il codice 8220-PSSE0908.

IL SOFTWARE

Per utilizzare il nostro plotter sarà necessario installare un apposito firmware sulla scheda Arduino ed un software di gestione lato PC. Il firmware che utilizzeremo è una versione derivata dal famoso GRBL, software open source creato per gli hobbisti che desiderano realizzare plotter o macchine CNC. Questo nasce per l'impiego in macchine a controllo numerico a tre assi con motori bipolari e driver dedicati. Per i nostri scopi serve una versione modificata adatta all'uso di motori unipolari e controllo dell'asse verticale con un servo da modellismo. La pagina di riferimento è la seguente: <https://github.com/ruizivo/GRBL-28byj-48-Servo>. Potete scaricare i file dal sito indicato oppure utilizzare i file della rivista, una volta scompattati, vi ritroverete con una cartella di nome GRBL-28byj-Servo che contiene a sua volta la cartella di nome *grbl*, che andrà copiata per intero nella cartella delle librerie di Arduino. Avviate l'IDE di Arduino, aprite l'esempio di nome *grblUpload.ino* e caricatelo, come si farebbe con un normale sketch, nella scheda Arduino. Per gestire la stampa è necessario un apposito software che possa dialogare con Arduino e

le invii gli appositi comandi; per questo utilizziamo il software open-source UniversalGcodeSender-v1.0.9 (UGS) che potete scaricare dal repository <https://github.com/winder/Universal-G-Code-Sender>. Si tratta di un software multi-piattaforma scritto in Java che non necessita di installazione; per avviarlo è sufficiente cliccare sul file *UniversalGcodeSender.jar*. A questo punto siete pienamente operativi e potete iniziare i primi collaudi e tarature.

Il firmware GRBL dispone di impostazioni predefinite che gli permettono un uso immediato; è tuttavia necessario verificare il corretto verso di rotazione dei motori e la corretta distanza di movimento. Connettete la scheda Arduino al PC ed avviate il software UGS, impostate la porta di comunicazione, il firmware alla versione GRBL e la velocità di comunicazione a 115 kbaud, quindi avviate la connessione con il plotter: il firmware risponderà con la conferma di avvenuta connessione e fornirà la versione installata.

Prima di sperimentare con i comandi di spostamento è necessario impostare alcuni parametri di funzionamento; per questo sfruttiamo la possibilità di inviare comandi al firmware GRBL tramite il tab *Commands*. Si tratta di inviare semplici comandi di testo similmente ai comandi AT dei modem. L'invio del comando \$ permette di visualizzare l'help dei comandi, mentre con il comando \$\$ si visualizzano le impostazioni attuali. Per impartire comandi è

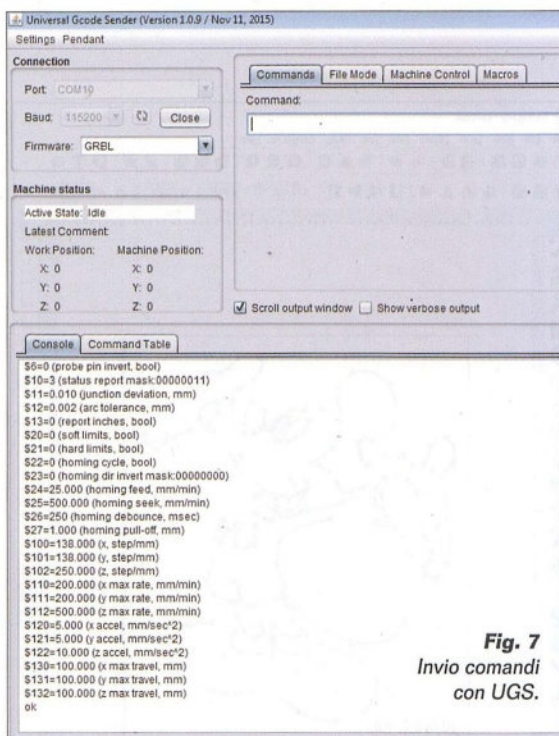


Fig. 7
Invio comandi
con UGS.

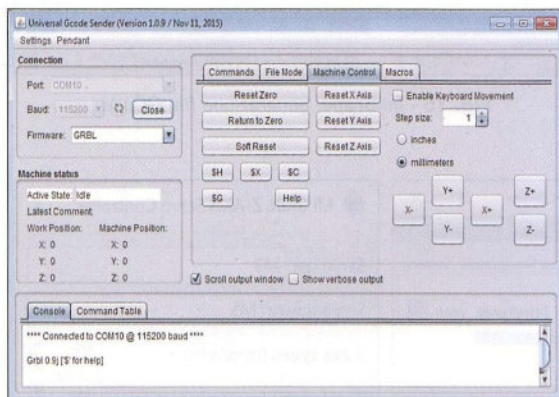


Fig. 8 - Gestione manuale del plotter con UGS.

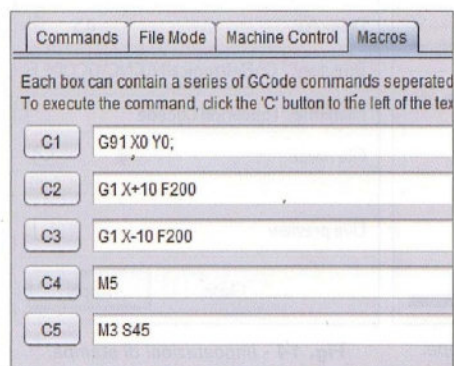


Fig. 9
Gestione
Macros
con UGS.

sufficiente indicare il codice del parametro da modificare ed il valore da impostare (Fig. 7). Per prima cosa riduciamo l'accelerazione dei motori inviando i comandi \$120=5 e \$121=5, poi modifichiamo la massima velocità di movimento con i comandi \$110=200 e \$111=200, i valori dell'asse Z sono ininfluenti. Modifichiamo anche i parametri 100 e 101 che specificano il numero di step necessari per far muovere il pennino di 1mm: nel nostro caso 138 impulsi. Ci sono anche comandi appositi per gestire manualmente il servo: con il comando M5 riportiamo il servo nella posizione zero, mentre con il comando M3 45 spostiamo il servo di 45° (ovviamente potremmo variare quest'ultimo valore secondo necessità). Con questi comandi potete anche verificare il corretto funzionamento del pennino ed impostare la corretta rotazione della squadretta. Il pennino toccherà il piatto per caduta libera, perché il servo serve solo per alzarlo dal piatto.

Dopo questa prima fase passiamo al tab *Machine Control*, dal quale sarà possibile interagire con il plotter manualmente (Fig. 8); ricordate che il punto zero macchina si trova in basso a sinistra del piatto. Agendo sulle frecce potete eseguire piccoli movimenti; in questa fase verificate che premendo la freccia X+ il pennino si sposti effettivamente verso destra, se così non fosse è necessario invertire il verso di rotazione dei motori come indicato precedentemente. Fate una prima verifica impostando uno spostamento di almeno 50mm e con una stecca verificate se effettivamente il movimento coincide, se così non fosse ritoccare i parametri 100 e 101 per gli assi X e Y rispettivamente. Nella scheda *Macros* (Fig. 9) è possibile impostare alcuni comandi facilmente richiamabili al bisogno, nel nostro caso abbiamo specificato rispettivamente: ritorno a zero (G91 X0 Y0), spostamento a destra di 10mm alla velocità di 200mm/sec (G1 X+10 F200), spostamento a sinistra di 10mm alla velocità di 200mm/sec (G1 X-10 F200), servo in posizione di riposo (M5) ed infine servo ruotato di 45° (M3 S45).

STAMPA

Il software UGS accetta file di stampa in formato GCode, che sono semplicemente file di testo che contengono le coordinate e i comandi per il plotter. Assieme ai file del progetto, sul nostro sito troverete alcuni esempi di file GCode, compresi alcuni utili per la taratura del plotter; il file *square50x50.gcode* stamperà un quadrato di 50mm di lato. In questa fase potete verificare ulteriormente la precisione degli spostamenti.

Passiamo ora alla stampa vera e propria spiegando

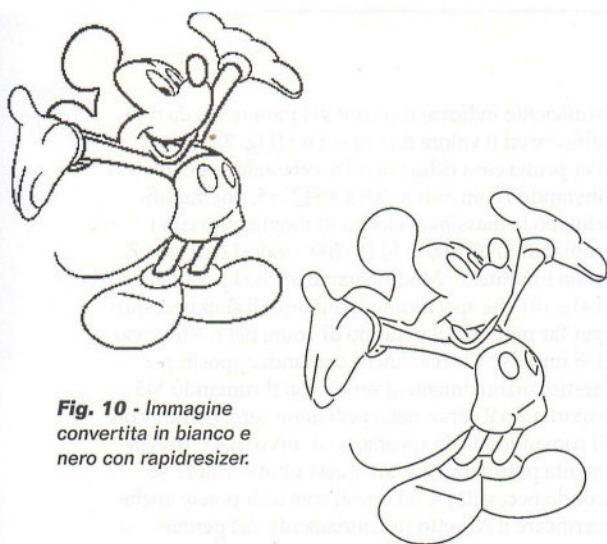


Fig. 10 - Immagine convertita in bianco e nero con rapidresizer.

Fig. 12 - Immagine solo contorno elaborata con rapidresizer.

passo-passo il procedimento tramite un esempio: supponiamo di voler stampare il popolare Topolino, partendo da un file immagine a colori che lo ritrae; siccome il nostro plotter stampa a un solo colore, non possiamo stamparlo direttamente ma è necessario prima convertirlo in bianco e nero; per fare questo utilizziamo un servizio on-line disponibile all'indirizzo <http://online.rapidresizer.com/photograph-to-pattern.php>. Il risultato della conversione lo vedete in Fig. 10; le impostazioni predefinite vanno già benissimo, quindi salvate il file elaborato in formato PNG. Dal disegno in bianco e nero è necessario ricavare un disegno con in evidenza solo le linee da tracciare (Fig. 11); per fare questo utilizziamo il servizio online all'indirizzo <http://online.rapidresizer.com/tracer.php>, nelle impostazioni selezionare centerline (Smoothness a metà) e salvate l'immagine elaborata in formato SVG. Se disponete di un'immagine tipo silhouette, entrambi questi passaggi non servono. A questo punto dovete utilizzare un software in grado di generare i file GCode adatti al nostro plotter: tra quelli free di

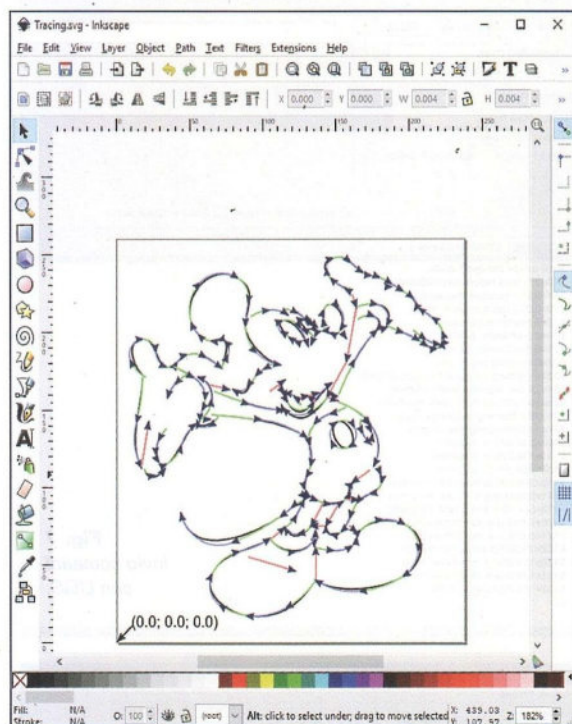


Fig. 15 - Inkscape: generazione file GCode.

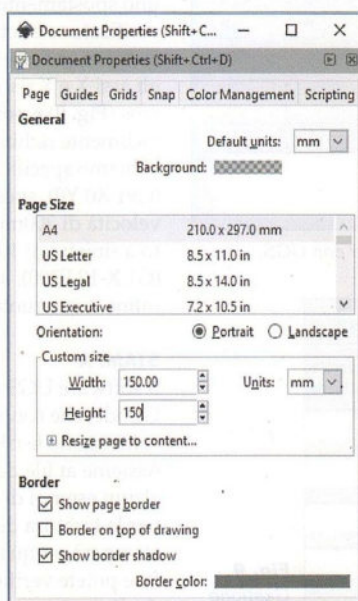


Fig. 13 - Impostazioni Inkscape.



Fig. 14 - Impostazioni di stampa.



Fig. 15
Alcune stampe
realizzate.

avviare la stampa: riapriamo UGS e nel tab file, tramite il pulsante Browse, selezionate il file GCode appena generato; quindi avviate la stampa cliccando sul pulsante Send. Se cliccate su Visualize, vi sarà possibile vedere anche a video il movimento del pennino. Durante la stampa il software mostrerà i comandi inviati, il tempo di stampa e quello rimanente. A patto di accettare qualche compromesso sulla qualità di stampa è possibile stampare i PCB direttamente su vetronite ramata (non presensibilizzata), ottenendo un supporto da poter trattare direttamente con acido per la realizzazione dei circuiti stampati. La larghezza delle piste coincide con lo spessore del pennarello usato, che andrà scelto di conseguenza.

CONCLUSIONI

A differenza delle stampanti, il plotter è idoneo alla stampa su superfici rigide come cartoncini e tavole di legno; un'idea potrebbe essere sostituire il pennino con un LASER per poter eseguire incisioni di precisione sui materiali più disparati. Non è invece possibile sostituire il pennino con una fresa, perché la struttura meccanica non è sufficientemente rigida e il filo non consente di trasmettere alla testa la forza necessaria.

Parametro	Significato	Valore consigliato
Servo UP	Comando per alzare il pennino	M3
Servo down	Comando per abbassare il pennino	M5
X axis speed	Velocità avanzamento pennino asse X	200
Y axis speed	Velocità avanzamento pennino asse Y	200
Angle for servo	Angolazione del servo quando pennino basso	45
Delay	Ritardo per staccare il pennino dal foglio	0.5

Tabella 2 - I parametri di stampa più rilevanti.

facile utilizzo trovate Inkscape, scaricabile gratuitamente all'indirizzo <https://inkscape.org/en/>, a noi serve la specifica versione 0.48. Per poter gestire i file GCode è necessario installare l'apposito plug-in che potete scaricare all'indirizzo www.mediafire.com/file/9x1m60w1x6tdhaw/MI+Inkscape+Extension.zip. Per installare questo plug-in è sufficiente scompattare i file e copiarli nella cartella `\App\Inkscape\share\extensions` di InkScape (Fig. 12 e Fig. 13). Avviate il software ed importate il file SVG precedentemente ricavato, oppure disegnatene uno a vostro piacimento; impostate come unità di misura i mm e settate l'area di lavoro in 150x150mm, quindi scalate l'immagine a seconda della reale dimensione di stampa desiderata. Dal menu path selezionate la voce object to path e per ultimo dal menu Extensions cliccate su MI GRBL Z axis Servo controller, si aprirà una finestra di dialogo in cui potete impostare alcuni parametri di stampa, il cui significato è riportato nella Tabella 2. Cliccate su Apply e il software provvederà a generare il file Gcode (Fig. 14); ad operazione conclusa, sullo schermo verrà mostrato il tracciato di stampa. Non ci resta che



per il MATERIALE

Il materiale per realizzare questo progetto è di facile reperibilità; quello elencato di seguito è invece disponibile presso Futura Elettronica. La board Arduino Uno Rev3 (cod. ARDUINOUNOREV3) è in vendita a Euro 24,50, lo stepper motor a 5 fili + driver con ULN2003 (cod. SM2003DB) è disponibile a Euro 4,90, la barra rettificata con diametro 8 mm lunga 1 metro (cod. BARRA8) costa Euro 8, il servo analogico 9 g - 23x12,5x30 mm (cod. SERVO206) è in vendita a Euro 7,50 e l'alimentatore switching 5 VDC / 2 A con plug DC (COD. AL5V2A) è disponibile a Euro 8,50. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 • Fax: 0331-792287 - www.futurashop.it

Futura Energy

Fai il pieno di energia con le batterie alcaline
di **FUTURA ELETTRONICA!**



**ELEVATA CAPACITÀ
AD UN PREZZO
IMBATTIBILE!**

Prezzi IVA inclusa.



**BATTERIE
ALCALINE
LR6 AA 1,5V**

Blister
da 4 pezzi

€ 1,50

Cod. LR6-AA-4



**BATTERIE
ALCALINE
LR3 AAA - 1,5V**

Blister
da 4 pezzi

€ 1,50

Cod. LR03-AAA-4



**BATTERIA
ALCALINA
6LR61 - 9V**

Blister
da 1 pezzo

€ 1,80

Cod. 6LR61-9V-1



**BATTERIE
ALCALINE
23A - 12V**

Blister
da 5 pezzi

€ 2,50

Cod. 12V-23A-5



**BATTERIE
ALCALINE
LR14 C - 1,5V**

Blister
da 2 pezzi

€ 2,30

Cod. LR14-C-2



**BATTERIE
ALCALINE
LR20 D - 1,5V**

Blister
da 2 pezzi

€ 3,20

Cod. LR20-D-2

**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

SAFETY SEA: SALVA L'UOMO IN MARE

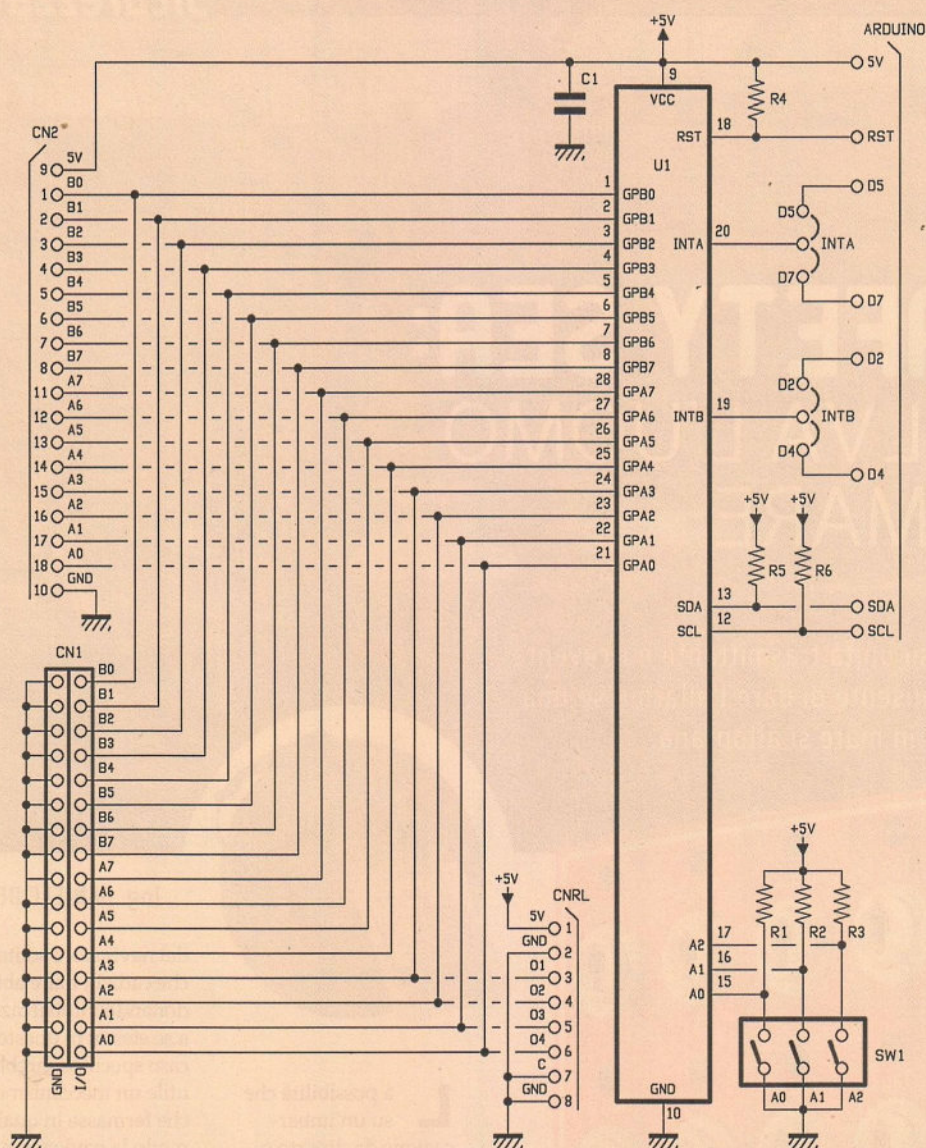
Basato su unità trasmettente e ricevente LoRa, consente di dare l'allarme se una persona in mare si allontana.



Ing. DANIELE DENARO

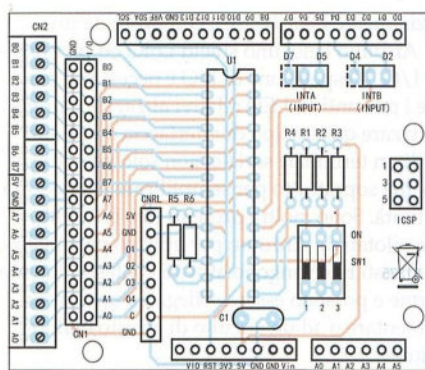
del navigatore solitario che cada in mare abbandonando l'imbarcazione a se stessa; in questo caso specifico sarebbe utile un meccanismo che fermasse in qualche modo la navigazione, come ad esempio una cordicella da fissare al corpo, che venendo tirata dall'allontanamento dalla barca fa spegnere il motore. Tale soluzione meccanica può essere sostituita con una elettronica, applicabile anche a imbarcazioni di grandi dimensioni; un esempio è il progetto qui descritto, che è basato sui moduli

La possibilità che su un'imbarcazione da diporto o su un peschereccio si presenti una condizione di pericolo a causa di avverse condizioni meteorologiche è tutt'altro che remota; in particolare può capitare che qualcuno cada in mare senza che altri membri dell'equipaggio se ne rendano conto in tempo per provvedere al salvataggio. Situazione anche più pericolosa è quella



radio LoRa. In realtà i sistemi LoRa, come fa intendere il loro nome (Long Range) sembrerebbero poco indicati a segnalare l'allontanamento dell'apparato a poca distanza da una centralina di controllo, a meno di non voler utilizzare anche un modulo GPS che possa segnalare direttamente la posizione. Ma per questo progetto, che abbiamo chiamato Safety Sea, volevamo privilegiare la semplicità e l'economicità, per cui ci siamo basati esclusivamente sul raggio di ricezione, giocando sulla grande flessibilità e configurabilità dei moduli LoRa. Questo progetto è una delle tante applicazioni del sistema LoRa e non è da

considerarsi come sostituto dei sistemi di sicurezza che ogni imbarcazione dovrebbe avere. Nella Fig. 1 abbiamo schematizzato il modo di funzionamento del Safety Sea, che si compone di una centralina che interroga, ogni 15 secondi, dei risponditori (*Safety Tag*, Fig. 2) indossabili e da inserire in involucri stagni. Nel caso che uno di questi *Tag* (fino ad 8) non risponda, viene attivata una sirena d'allarme e con essa due relè che potete utilizzare per azionare altri utilizzatori (un lampeggiante, per esempio) o interrompere l'alimentazione del motore della barca. Mentre la centralina è stata realizzata

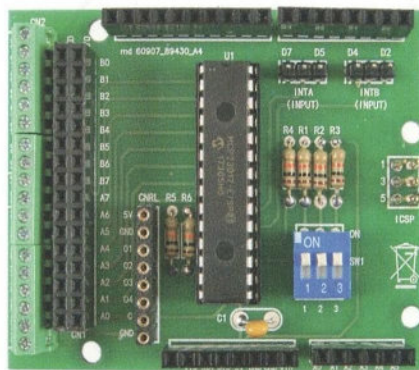


Elenco Componenti:

C1: 100 nF ceramico
R1÷R6: 4,7 kohm
SW1: Dip-Switch 3 vie
U1: MCP23017-E/SP

Varie:

-Zoccolo 14+14
-Strip maschio 3 vie (2 pz.)



-Jumper (2 pz.)
-Strip maschio 8 vie (5 pz.)
-Morsetto passo 2.54mm 6 vie (3 pz.)
-Strip M/F 6 vie
-Strip M/F 8 vie (2 pz.)
-Strip M/F 10 vie
-Strip M/F 2x3 vie
-Circuito stampato S1375 (69x55 mm)

con un Arduino corredato di shield LoRa (ed altra componentistica), per il risponditore è stata utilizzata la scheda "Maduino" composta da un Arduino e da un modulo LoRa. Maduino è stata presentata nell'articolo della nuova libreria LoRa (fascicolo n° 225) e in questa applicazione viene alimentata da un piccolo "PowerBank" a uscita USB. Il consumo del Tag è molto basso ed un piccolo "PowerBank" può durare anche diversi giorni. Per ridurre la portata radio abbiamo portato la potenza di emissione al minimo (circa 1,5 mW) e fissato i parametri *Spreading Factor* e *Band Width* ai seguenti valori: SF=7 e BW=8. In questo modo la sensibilità si è ridotta a circa -120dBm ed il raggio di azione si attesta intorno ai 20m. L'immunità ai disturbi propria della modalità di trasmissione LoRa e l'utilizzo di un protocollo digitale, garantiscono una notevole affidabilità. Lo sketch del risponditore (*Safety Tag*) è abbastanza semplice (lo proponiamo nel **Listato 1**; N.B. I listati del progetto fanno riferimento alla libreria Arduino per i moduli LoRa scaricabile all'indirizzo: <https://github.com/open-electronics/LoRa>); non fa altro che attendere la richiesta di risposta da parte del server (senza time-out). Per ridurre i consumi dopo aver risposto si mette in riposo per i primi 8 secondi, confidando sull'intervallo di scansione del server che è impostato sui 15 secondi. Lo sketch è

unico, per cui l'indirizzo del risponditore è definito da un micro-switch a 3 bit. Questo switch viene letto alla partenza e forma un numero binario "n" da 0 a 7 che sarà utilizzato per formare l'indirizzo n+2, cioè da 2 a 9 perché l'indirizzo 0 non è utilizzabile dalla libreria e l'indirizzo 1 è destinato al server. Sulla centralina, infine, i Tag sono nominati da 1 a 8, per semplicità (ovvero n+1).

CENTRALINA

La centralina, anche detta Server del nostro sistema, è alimentata a 12V, perché 12 volt è la bassa tensione tipicamente presente nelle imbarcazioni. Questa tensione alimenta Arduino Uno e tramite il regolatore interno ad Arduino, si ottengono i 5V necessari a far funzionare le schede di espansione. I 12 volt possono essere utilizzati da una sirena. La centralina è stata pensata per essere posizionata nell'imbarcazione in modo da essere facilmente accessibile e di semplice utilizzo; ha infatti 8 grandi pulsanti che abilitano o disabilitano il controllo di altrettanti "Safety Tag". Ogni "Safety Tag" ha un indirizzo configurabile tramite un dip-switch presente sul contenitore, affinché possa essere rilevato distintamente; va da sé che ciascuno dei tag deve avere un indirizzo univoco, vale a dire differente da quello degli altri abbinati al Server.

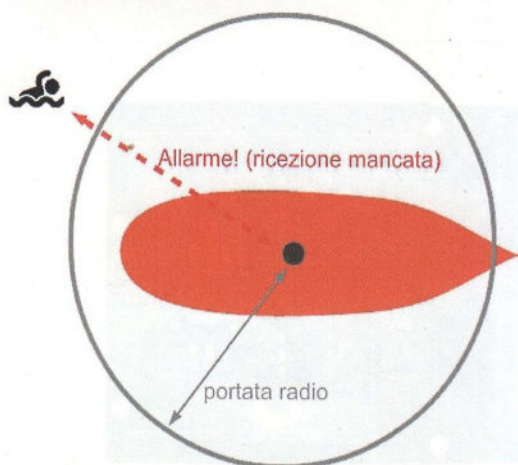
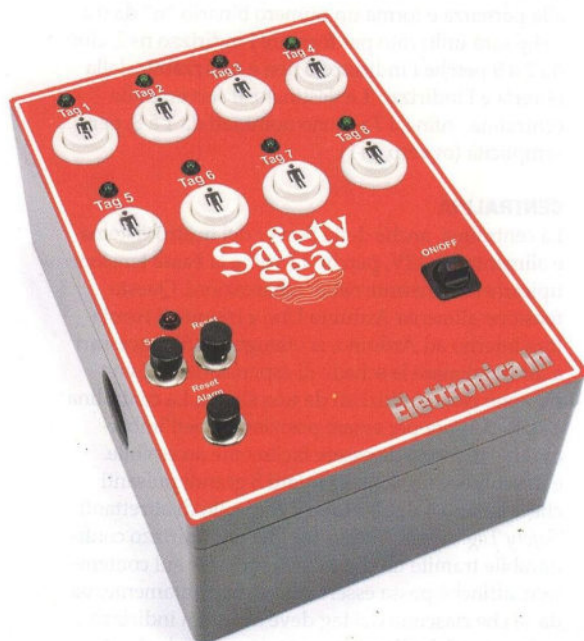


Fig. 1 - Principio di funzionamento.

Come detto, gli indirizzi sono in forma binaria da 0 a 7, ma vengono normalmente indicati sulla centralina con i valori da 1 a 8. I Tag attivi possono essere salvati nella EEPROM di Arduino Uno, cosicché all'accensione della centralina non c'è bisogno di attivarli di nuovo.

Nella **Tabella 1** vedete l'impostazione dei quattro dip-switch di ogni Tag per ottenere gli otto possibili indirizzi: da sinistra a destra, i dip sono 1, 2, 3, 4.



Sostanzialmente l'hardware è composto da una scheda Arduino Uno, uno shield LoRa ed un espansore di I/O. L'espansione di I/O è necessaria per pilotare i pulsanti e i LED relativi ai Tag. Si è preferito utilizzare questo tipo di interfaccia a pulsanti invece di un terminale seriale, non solo per ridurre il costo, ma soprattutto per semplificare al massimo l'operatività. Sono inoltre presenti due relè optoisolati e pilotati da Arduino per utilizzo personalizzabile. Questi relè vengono attivati contestualmente all'allarme e possono essere collegati a una sirena supplementare o ad un circuito di arresto dell'imbarcazione.

Come vedete nel relativo schema elettrico in queste pagine, la scheda di espansione di I/O è basata sull'integrato MCP23017, perché di esso esistono librerie per Arduino. Questo componente comunica con Arduino tramite il bus I²C e di esso (quindi dello shield) se ne possono collegare al bus stesso fino ad 8, in quanto l'integrato è indirizzabile con 3 bit. La scheda utilizzata ha un dip-switch a 3 posizioni con indirizzo prefissato pari a 0. La libreria da utilizzare è quella della Adafruit ed è scaricabile al seguente indirizzo: <https://github.com/adafruit/Adafruit-MCP23017-Arduino-Library>; essa permette di trattare i 16 I/O aggiunti con lo stesso approccio

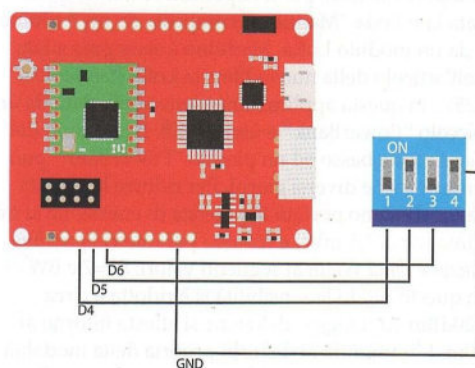


Fig. 2 - A sinistra un'immagine della centralina, qui sotto il Safety Tag (risponditore).



Listato 1 - Firmware risponditore

```
#include "LowPower.h"           //Include la libreria power-down
#include "LoraNode.h"           //Include la libreria Lora
LoraNode Node();                //Istanza del nodo

#define MASTERADD 1             //Indirizzo del server (centralina)

#define pin0 4                  // microswitch per definire l'indirizzo
#define pin1 5                  // "
#define pin2 6                  // "

void setup()
{
  Node.begin();                 //Start del modulo radio
  Node.LR.setLowPower(2);        //Potenza al minimo (2 dBm)
  Node.setSpreadingFactor(7);    //SpreadingFactor cod. 7
  Node.setBandWidth(8);          //Band Width cod. 8
  Node.setNodeAdd(setNodeNum()); //Definisce il suo indirizzo
}

void loop()                     //Quando arriva la richiesta (da 1) risponde
{
  if (Node.newMessAvailable(MASTERADD,0)) {imhere();}
  nodesleep();                  //Dorme per 8 secondi
}

void imhere()                   //Risponde
{
  Node.sendMessage(MASTERADD, "OK", 500);
}

void nodesleep()
{
  Node.LR.setSleepState(true);
  LowPower.powerDown(SLEEP_8S, ADC_OFF, BOD_OFF);
}

byte setNodeNum()
{
  pinMode(pin0, INPUT_PULLUP);
  pinMode(pin1, INPUT_PULLUP);
  pinMode(pin2, INPUT_PULLUP);
  byte num=0;
  if (digitalRead(pin0)==LOW) bitSet(num,0);
  if (digitalRead(pin1)==LOW) bitSet(num,1);
  if (digitalRead(pin2)==LOW) bitSet(num,2);
  return num+2;                 //Indirizzo del tag
}
```

dei pin di Arduino, quindi è possibile definire ogni singolo pin come ingresso o uscita ed è altresì possibile leggere o scrivere il valore digitale con funzioni simili a quelle di base di Arduino.

Lo sketch della centralina (che riportiamo nel Listato 2) non fa altro che interrogare i Tag attivi ciclicamente con un intervallo temporale di 15 secondi; tra una scansione e l'altra controlla se è stato attivato o disattivato un Tag. Alla partenza, in ogni caso, verifica se è stata salvata una configurazione di Tag

attivi ed in questo caso la ricarica. La configurazione dei Tag attivi può essere salvata in ogni momento in memoria EEPROM tramite il pulsante dedicato. Se un Tag non risponde viene attivato l'allarme. Questo può essere disabilitato dal pulsante di reset. Sono stati utilizzati come input digitali anche un paio di pin analogici A0 (D14) e A1 (D15); infatti, in Arduino, i pin analogici possono essere gestiti come pin digitali chiamandoli con la numerazione da D14 a D19.

ON	ON	ON	ON	ON	ON	ON	ON
1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
Tag 1	Tag 2	Tag 3	Tag 4	Tag 5	Tag 6	Tag 7	Tag 8

Tabella 1 - Posizione dei dip-switch per identificare i tag.

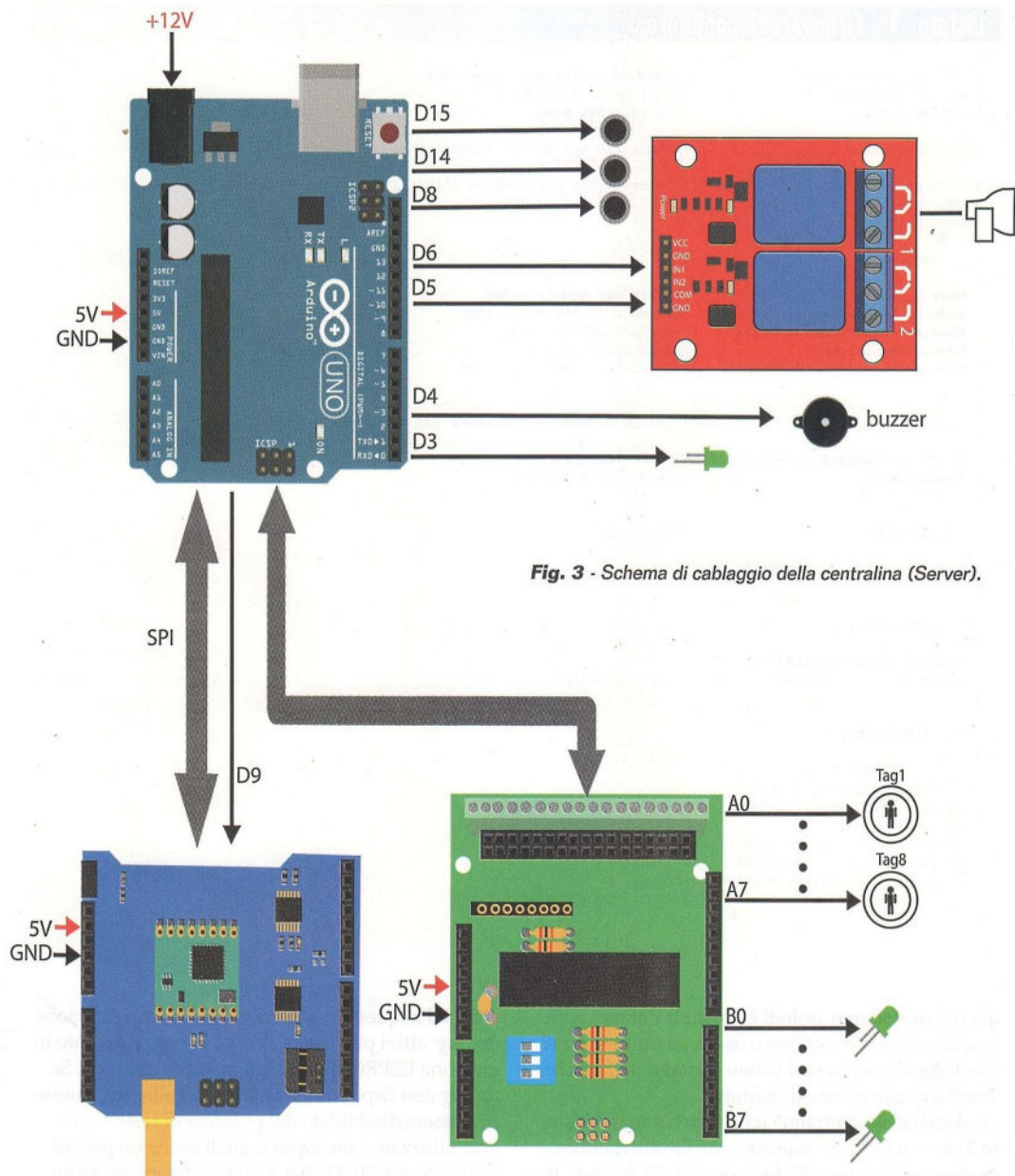


Fig. 3 - Schema di cablaggio della centralina (Server).

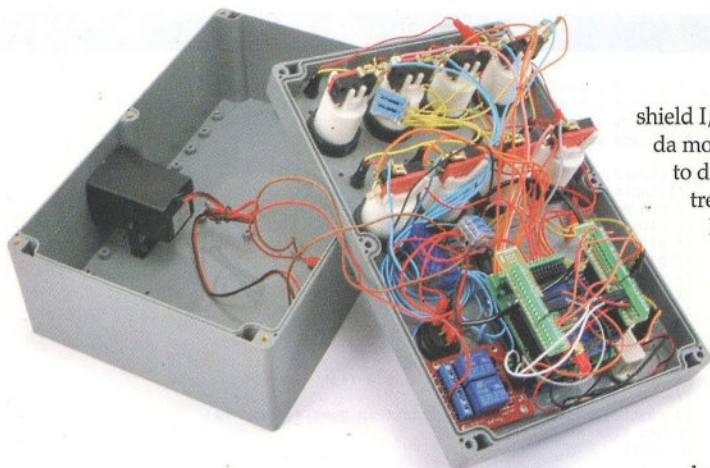
Lo sketch utilizza la classe *Tag* (vedere Listato 2) creata per maggiore leggibilità del codice; questa classe definisce 8 oggetti *Tag* che contengono le caratteristiche dei singoli risponditori (indirizzo, attivazione ecc.) e le funzioni che li manipolano.

Nella Fig. 4 riportiamo il diagramma di flusso corrispondente all'algoritmo utilizzato nel Server.

Come accennato, lo sketch del server utilizza una classe separata la cui istanze rappresentano gli 8 *Tag*; la classe tiene conto delle caratteristiche del *Tag* come,

ad esempio, se è attivo, oppure qual è il suo numero di indirizzo o la porta corrispondente al pulsante di abilitazione/disabilitazione o, ancora la porta corrispondente al LED di segnalazione, ecc. Infine la classe *Tag* contiene i metodi (funzioni) che verificano se il pulsante è stato premuto, oppure che accendono/spengono il LED corrispondente e le altre funzioni che modificano o leggono lo stato del *Tag*.

Nel Listato 3 trovate il codice corrispondente alla classe *Tag*.



REALIZZAZIONE PRATICA

Passiamo adesso alla costruzione del sistema, che sarà composto da una centralina da assemblare in un contenitore (per il prototipo abbiamo utilizzato una scatola grigia per impianti elettrici) secondo lo schema in Fig. 3 e da un certo numero di tag, anch'essi racchiusi in un contenitore per elettronica individualmente; i tag dovranno poi essere inseriti in una custodia impermeabile, magari da tenere al collo, come quella che vedete nelle foto in queste pagine, che è della Tribord.

Iniziamo con la centralina, la cui elettronica è composta da una scheda Arduino Uno sulla quale sono innestati, nell'ordine, uno shield I/O Expander e uno LoRa Shield ultima versione; nelle foto vedete che agli header di tale shield abbiamo applicato due schede di conversione da essi a morsettiere miniatura, ma potete farne a meno se per il cablaggio utilizzate solo jumper per Arduino.

L'alimentazione, prelevata da una presa jack da pannello montata su un lato del contenitore, va portata alla scheda Arduino mediante un cavetto bipolare rosso-nero terminante con un jack adatto a quello del DC-in dell'Arduino Uno. Agli input Ax dello

shield I/O Expander collegate i pulsanti previsti, da montare a pannello (magari prendete spunto dal prototipo che abbiamo realizzato) mentre alle uscite Bx collegate i LED di segnalazione. Alle porte D5e D6 connettete gli ingressi di una scheda relé a due canali, che nel caso specifico è il MODULO 2 RELÉ 5 VDC 10 A della Futura Elettronica, contraddistinto dal codice prodotto 2846-RELAY-2CH. I relé di bordo serviranno a comandare una sirena a 12V

(da montare su un lato della scatola dopo aver realizzato - con una sega a tazza montata su un trapano - il foro per affacciarla all'esterno) ed un carico ausiliario che eventualmente aggiungerete in base al bisogno. Sul pannello frontale, cioè dove avete montato i pulsanti, disponete un LED per ciascuno di quelli dedicati ai tag. Quanto al tag, è composto da una board Maduino della quale portiamo all'esterno il dip-switch a 4 vie su un pezzetto di basetta millefori, così da renderli accessibili dal frontalino del contenitore; naturalmente è una soluzione tra quelle possibili, perché ognuno di voi può decidere di disporre il dip-switch come meglio crede.



Il tag con il cavo USB per il Power Bank, chiuso in una custodia impermeabile.

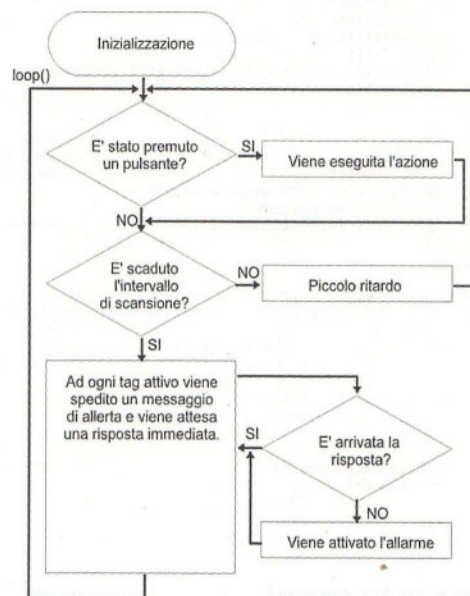


Fig. 4 - Algoritmo del server.

Listato 2 - Safety Sea Server

```
#include "LoraNode.h" //Include libreria Lora
#include "Adafruit_MCP23017.h" //Include libreria IO-expansion
#include "Tag.h" //Include classe Tag
#include <EEPROM.h> //Include libreria EEPROM
#define thisNode 1
LoraNode Node(thisNode); //Istanza nodo 1 (server)
Adafruit_MCP23017 mcp; //Istanza espansione
#define psound 4 // pin per il cicalino
#define pinf 3 // pin per il led rosso di segnalazione
#define pinR1 6 // pin per il relè 1
#define pinR2 5 // pin per il relè 2
#define areset 8 // pin per il pulsante di reset allarme
#define slist 15 // (A1) pin per il pulsante che salva la lista
#define rlist 14 // (A0) pin per il pulsante che cancella la EEPROM
#define SCANTIME 15000 // intervallo (millisecondi) di scansione tag
boolean SHIELD=true;
#define ntag 8
Tag termlist[ntag]; // array di istanze tag (vedi classe Tag)
char Alert[]="active"; //messaggio spedito ai tag
boolean fctl=true; // controllo sì/no (non utilizzato in questa versione)
boolean alarm=false; // flag allarme off
unsigned long T; // timer

void setup()
{
  Serial.begin(9600); // nel caso si colleghi un terminale seriale
  configure(); // configura pin e istanze tag
  if (!Node.begin()) {Serial.println("No LoRa shield detected! Stop!"); SHIELD=false; return;}
  Serial.println("SafetySea LoRa server.");
  Node.LR.setLowPower(2); // minima potenza (2 dBm: 1.5 mW)
  Node.setSpreadingFactor(7); // configurazione di minima sensibilità
  Node.setBandwidth(8); // "
  Node.printAddresses(); // in caso di collegamento a terminale seriale
  Node.printConfig();
  Serial.println("Start surveillance server");
  startcontrol(); // predisporre il controllo periodico
  T=millis()+SCANTIME; // Inizializza il timer
}

void configure()
{
  digitalWrite(pinf,LOW);
  digitalWrite(psound,LOW);
  digitalWrite(pinR1,LOW);
  digitalWrite(pinR2,LOW);
  pinMode(pinf,OUTPUT);
  pinMode(psound,OUTPUT);
  pinMode(pinR1,OUTPUT);
  pinMode(pinR2,OUTPUT);
  pinMode(areset,INPUT_PULLUP);
  pinMode(slist,INPUT_PULLUP);
  pinMode(rlist,INPUT_PULLUP);
  mcp.begin();
  for (int i=0;i<ntag;i++) termlist[i].setTag(i); // Inizializza le istanze Tag
  loadlist();
}

void loop()
{
  termconfig(); // controllo continuo periodico
  if (millis()>T) // controllo pulsanti
  { // Se intervallo scaduto scansione dei tag
    T=millis()+SCANTIME;
    if (fctl) terminalcontrol(); // scansione dei tag
  }
}

void termconfig()
{
  for (int i=0;i<ntag;i++) termlist[i].checkButton();
  if (alarm) // se allarme ON controlla se deve essere disattivato
  { if (digitalRead(areset)==LOW) // solo se Reset premuto a lungo
    {delay(1000); if (digitalRead(areset)==LOW) resetalarm();}
  }
  if (digitalRead(slist)==LOW) savelist();
  if (digitalRead(rlist)==LOW) cancllist();
}

void terminalcontrol() // controllo di tutti i tag attivi
{ int TAGADD=1;
```

UTILIZZO DEL SISTEMA

Dopo aver collegato la centralina all'alimentazione a 12 volt, la si accende con l'interruttore luminoso.

A questo punto, dopo aver alimentato tutti i Tag collegandoli ai PowerBank, li si attiva sulla centralina pigiando il relativo bottone. Alla scansione dei Tag


```

for (int i=0;i<ntag;i++)
{ if (!termist[i].isActive()) continue;
  TAGADD=termist[i].getAddress();
  Node.sendMessage(TAGADD,Alert,500); // spedisce messaggio
  if (!Node.newMessAvailable(TAGADD,1000)) setalarm(i); // no risposta= allarme
  else termist[i].blinkled(); // altrimenti breve blink
}
}
/***** Funzioni accessorie *****/
void startcontrol()
{ fctl=true; Serial.println("Control started!"); listtag(); }

void listtag() // in caso di collegamento a terminale seriale
{ int count=0;
  for (int i=0;i<ntag;i++) {if (termist[i].isActive()) count++;}
  if (count==0) {Serial.println("No tag active!");return;}
  Serial.print(count);Serial.print(" active tags: ");
  for (int i=0;i<ntag;i++)
  { if (!termist[i].isActive()) continue; Serial.print(termist[i].getName()); Serial.print(" "); }
  Serial.println();
}

void setalarm(int tag) // attiva i due relè e il cicalino
{ alarm=true;
  termist[tag].setAlarm(true);
  Serial.print("Alarm! On tag ");Serial.println(tag+1);
  digitalWrite(pinR1,HIGH);
  digitalWrite(pinR2,HIGH);
  sound(0,0,0); // suono continuo del cicalino
}

void resetalarm() // disattiva i due relè ed il cicalino
{ alarm=false;
  digitalWrite(pinR1,LOW);
  digitalWrite(pinR2,LOW);
  digitalWrite(psound,LOW);
  for (int i=0;i<ntag;i++) termist[i].setAlarm(false);
}

void blinkpinf(int time,int n)
{ int i;
  byte p=1;
  for (i=0;i<n;i++) {digitalWrite(pinf,p);delay(time); p=p^1;}
  digitalWrite(pinf,0);
}

void sound(int timeon,int timeoff,int n)
{ int i;
  if (n==0) {digitalWrite(psound,1);return;}
  for (i=0;i<n;i++)
  {digitalWrite(psound,1);delay(timeon);digitalWrite(psound,0);delay(timeoff);}
  digitalWrite(psound,0);
}

void savelist() // salva i tag attivi su EEPROM (a partire dalla posizione 20)
{ digitalWrite(pinf,HIGH);
  for (int i=0;i<ntag;i++)
  { EEPROM.write(20+i,termist[i].isActive());}
  blinkpinf(300,8);
  digitalWrite(pinf,HIGH); // segnala che in memoria c'è una lista
}

void loadlist() // se la posizione 20 non ha valore 255 carica gli 8 valori successivi
{ if (EEPROM.read(20)==255) return;
  for (int i=0;i<ntag;i++)
  {termist[i].setActive(EEPROM.read(20+i));}
  blinkpinf(1000,4);
  digitalWrite(pinf,HIGH); // segnala che in memoria c'è una lista
}

void canclist() // pone a 255 il valore della posizione 20 rendendo nulla la lista
{ EEPROM.write(20,255);
  blinkpinf(100,12);
  digitalWrite(pinf,LOW); // segnala che in memoria non c'è una lista
}

```

corrisponderà un breve lampeggio di ogni *Tag* abilitato. Nel caso il *Tag* non fosse raggiungibile si attiverà l'allarme. Per spegnerlo si può tenere premuto il

pulsante di reset per circa un secondo. Ogni *Tag* può essere disabilitato premendo di nuovo il relativo pulsante un po' più a lungo fino a che il led verde

Listato 3 - Classe Tag

```
#ifndef Tag_h
#define Tag_h

extern Adafruit_MCP23017 mcp; // Espansione di IO

class Tag
{
public:
    void setTag(int index) // istanza con configurazione dei parametri
    {
        // index=indice dell'array (da 0 a 7)
        address=index+2; // indirizzo usato per la trasmissione (da 2 a 9)
        name=index+1; // nome del tag sulla consolle (da 1 a 8)
        active=false; // inattivo per definizione
        alarm=false; // (non utilizzato in questa versione)
        pinbutton=index; // indice della porta pulsante MCP (da 0 a 7)
        pinled=index+8; // indice della porta led MCP (da 8 a 15)
        ledstat=0; // stato del led
        // configurazione pulsante e led con MCP23017
        mcp.pinMode(pinbutton, INPUT);
        mcp.pullUp(pinbutton, HIGH); // pull-up pulsante
        mcp.pinMode(pinled, OUTPUT);
        mcp.digitalWrite(pinled, LOW); // led spento
        oldb=1; // valore di partenza della porta pulsante
    }

    // funzioni pubbliche (metodi)
    int getAddress(){return address;}
    void setActive(bool act)
    {active=act;if (active) {led(1);seron();}else {led(0);seroff();}}
    bool isActive(){return active;}
    void setAlarm(bool al){alarm=al;}
    bool isAlarm(){return alarm;}
    void led(int state){mcp.digitalWrite(pinled,state);ledstat=state;}
    bool checkButton()
    {
        if (!active) {if (button()) setActive(true); return;}
        else {if (buttonlong()) setActive(false); return;}
    }
    void blinked() // cambia stato due volte
    {
        ledstat=ledstat^1;mcp.digitalWrite(pinled,ledstat);
        delay(300);
        ledstat=ledstat^1;mcp.digitalWrite(pinled,ledstat);
    }
    int getName(){return name;}

private:
    int address;
    bool active;
    bool alarm;
    int name;
    int pinled;
    byte ledstat;
    int pinbutton;
    byte oldb;

    bool button() // vero solo se passa da alto a basso
    {byte b=mcp.digitalRead(pinbutton);
    byte ret=(b^oldb)&oldb; oldb=b;return ret;}

    bool buttonlong() // vero solo se rimane premuto a lungo {
        oldb=mcp.digitalRead(pinbutton);
        if (oldb==1) return false;
        int i;
        for (i=0;i<10;i++)
        {
            oldb=mcp.digitalRead(pinbutton);
            if (oldb!=0) break;
        }
        if (i>6) return true;
        else return false;
    }

    void seron() // utile solo se c'è un terminale seriale collegato
    {Serial.print("Tag ");Serial.print(name);Serial.println(" ON");}
    void seroff()
    {Serial.print("Tag ");Serial.print(name);Serial.println(" OFF");}
};

#endif
```


Listato 4 - Safety Sea per un solo Tag

```
#include "LoraNode.h" //Include libreria LoRa

#define thisNode 1
LoraNode Node(thisNode); //Istanza del nodo 1
#define TAGADD 2 // indirizzo del nodo Tag

#define pinR1 6 // pin rele1
#define pinR2 5 // pin rele2

#define SCANTIME 10000 // tempo di scansione
// (millisecondi)

char Alert[]="active";
unsigned long T;

void setup()
{
    configure();
    Node.begin();

    Node.LR.setLowPower(2);
    Node.setSpreadingFactor(7);
    Node.setBandWidth(8);

    T=millis()+SCANTIME;
}

void configure()
{
    digitalWrite(pinR1,LOW);
    digitalWrite(pinR2,LOW);
    pinMode(pinR1,OUTPUT);
    pinMode(pinR2,OUTPUT);
}

void loop()
{
    if (millis()>T) // Controlla l'unico
                    // Tag, se il timer è scaduto
    {
        T=millis()+SCANTIME;
        terminalcontrol();
    }
    else delay(100);
}

void terminalcontrol()
{
    Node.sendMessage(TAGADD,Alert,500);
    if (!Node.newMessAvailable(TAGADD,1000))
    setalarm();
}

void setalarm()
{
    digitalWrite(pinR1,HIGH);
    digitalWrite(pinR2,HIGH);
}
```

pulsanti e dell'espansione di I/O. Lo sketch sarebbe composto di un semplice loop dove viene verificata la risposta dell'unico Tag. Nella Fig. 5 è schematizzato l'hardware necessario, mentre nel Listato 4 è descritto lo sketch che girerà nel server, il quale, come vedete, è semplificato di molto rispetto a quello del Listato 2. L'hardware e il software del Tag rimangono immutati. La versione semplificata ad un solo tag può essere utile, ad esempio, a chi naviga da solo: in

questo caso lo proteggerà dalla dispersione in mare, dato che quando si è da soli in barca non c'è nessun altro in grado di provvedere al salvataggio. Allo scopo si utilizzeranno i relé per attivare un teleallarme GSM e per spegnere il motore tramite interruzione del carburante o del circuito della chiave di avviamento.

CONCLUSIONI

Con questo progetto abbiamo voluto proporvi un'applicazione inconsueta dei moduli LoRa che può risolvere una situazione di pericolo in mare, ma che dimostra la flessibilità e praticità dei moduli stessi. Una fase di ulteriore sviluppo potrebbe prevedere l'utilizzo di moduli GPS per la localizzazione a distanza dell'uomo in mare. In questo caso il tag verrebbe dotato di un piccolo ricevitore GPS e la scheda Maduino trasmetterebbe la posizione da esso rilevata, quindi l'allontanamento sarebbe determinato, dalla centrale, non da una perdita di segnale radio, ma da un dato sulla distanza del tag dall'imbarcazione (si potrebbe attivare una funzionalità tipo il geo-fence). In questo caso il link LoRa lavorerà a portata elevata, permettendo di geo-localizzare il Tag anche a distanza chilometrica, sfruttando propriamente la potenzialità della trasmissione LoRa.

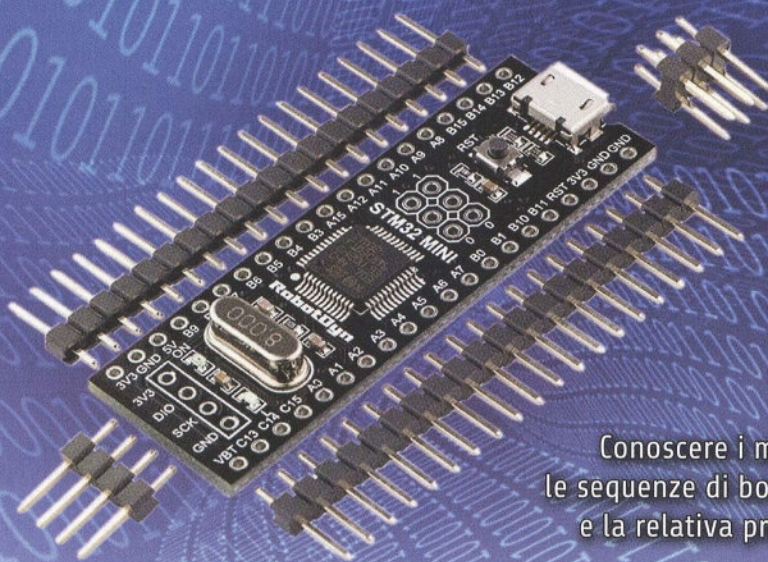


per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono di facile reperibilità; quelli elencati di seguito sono disponibili presso Futura Elettronica. La board Arduino Uno Rev3 (cod. ARDUINOUNOREV3) è disponibile a Euro 24,50, l'Arduino I/O Expander (cod. FT1375K) costa Euro 9,00, lo shield LoRa a 433 MHz per Arduino (cod. LORASHIELD433) è in vendita a Euro 31,00, il modulo a 2 relé 5 VDC / 10 A (cod. RELAY2CH) costa Euro 8,00 e la board ATmega328 con LoRa 433 MHz (cod. MADUINOLORA433) è in vendita a Euro 31,00. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 • Fax: 0331-792287 - www.futurashop.it



Conoscere i microcontrollori,
le sequenze di boot del firmware
e la relativa programmazione.

L'IMPORTANZA DEL BOOTLOADER

di PIERO BOCCADORO

Nella programmazione dei microcontrollori esistono codici, routine e framework di riferimento, che consentono l'esecuzione sequenziale di comandi che vanno a comporre il funzionamento complessivo del sistema, quando è completo il suo sviluppo. Ma, per poter programmare un generico microcontrollore, è necessario utilizzare una sequenza di operazioni, codificate in un certo ordine, così da agire su registri specifici, le cui corrette impostazioni consentono l'effettiva esecuzione del codice.

Quello che vi proponiamo in questo articolo è uno studio sulla programmazione dei microcontrollori,

con un focus sulle necessarie routine. Descriveremo Bootloader e sequenze di boot, per comprendere come funzionano i microcontrollori. E per capirne completamente ogni aspetto, faremo degli esempi analizzando alcuni casi particolarmente utili e significativi, ovvero soluzioni targate ST Microelectronics, con particolare riferimento alle funzionalità di bootloading built-in in microcontrollori ARM Cortex, ed altre, targate Infineon, afferenti alla famiglia XMC1000.

Dal confronto emergeranno tratti comuni e differenze la cui comprensione è fondamentale per poter programmare praticamente qualunque sistema.

Infine, analizzeremo questo argomento in chiave "making" per vedere se e come il movimento dei maker sia davvero riuscito a semplificare concetti, pratiche e software, rendendoli effettivamente appetibili anche per chi non è specialista del campo.

IL BOOTLOADER

In informatica, il bootloader è il programma che, nella fase di avvio (boot) del computer, carica il kernel del sistema operativo dalla memoria "secondaria" (di archiviazione, o di massa) alla memoria "primaria" (per intenderci, la RAM), permettendone l'esecuzione da parte del processore ed il conseguente avvio del sistema.

Il termine deriva dal fatto che il processo di avvio di un computer viene chiamato bootstrap (dal detto inglese to lift oneself by one's own bootstrap, "tirarsi su per le fibbie degli stivali"). Nel caso di più sistemi operativi installati sulla stessa macchina, il caricamento da parte del bootloader è preceduto dalla selezione del sistema operativo desiderato da parte dell'utente, tramite un apposito boot manager (una soluzione molto nota agli estimatori di Linux è GRUB).

La funzione fondamentale di un bootloader è dunque caricare ed eseguire un kernel di sistema operativo, unitamente ai processi ed ai servizi secondari. Nella maggior parte dei casi, questo richiede di accedere alla memoria di massa, per leggere il kernel del sistema operativo, e, potenzialmente, anche altri file. Nei casi più semplici, il bootloader contiene l'indirizzo dei blocchi di memoria in cui sono memorizzati i file da caricare, e deve, quindi, essere aggiornato se questi file vengono modificati. L'accesso al disco spesso avviene attraverso le funzioni fornite dal firmware (il BIOS nei sistemi PC IBM compatibili).

Alcuni bootloader posseggono la capacità di interpretare uno o più file system per trovare i file da caricare; in questo caso, possono anche caricare un file di configurazione dal disco, o permettere ad un utente (avanzato) di esplorare il disco alla ricerca di file da caricare per l'avvio selettivo di specifici servizi. Inoltre alcuni bootloader sono in grado di utilizzare le funzionalità fornite da alcune schede di rete (tipicamente ethernet) per scaricare un kernel dalla rete.

Esistono, inoltre, alcune funzionalità aggiuntive presenti in alcuni bootloader:

- molti kernel supportano la possibilità di ricevere parametri di avvio per configurare il loro comportamento; il bootloader si occupa di passarli al kernel, e, in alcuni casi, consente all'utente di

editarne alcuni;

- alcuni kernel prevedono che altri file siano resi disponibili durante il boot (per esempio nei sistemi Unix-like è comune l'uso di initrd. Il loro caricamento viene gestito dal bootloader;
- un bootloader può mostrare all'utente un menù di possibili kernel da caricare, con relativi parametri, permettendogli di scegliere quale sistema operativo avviare;
- un bootloader può richiedere una password per consentire l'avvio del sistema.

Il bootloader viene, in molti casi, installato in una posizione specifica di un dispositivo di memoria di massa - tipicamente nel primo blocco del primo disco ("primo" rispetto all'ordine con cui i dispositivi sono collegati al calcolatore) - da dove viene caricato dal firmware.

Ad esempio, nell'architettura del PC IBM-compatibile, il bootloader può essere memorizzato nel Master Boot Record (MBR) del disco di avvio, oppure nel primo settore della partizione che ospita il sistema operativo (nel qual caso deve essere richiamato da un boot manager installato nel MBR).

Può essere collocato nel primo settore di un volume avviabile, come ad esempio un ambiente di ripristino (un esempio è Windows Recovery Environment). Esistono alcune limitazioni imposte dalla tecnica e dai produttori di hardware e firmware, per aggirare le quali molti bootloader moderni operano ricorrendo al cosiddetto boot in due stadi: lo stage 1 e lo stage 2. Essi possono essere descritti come segue:

- lo **stage 1** è quello cui poc'anzi ci siamo riferiti chiamandolo boot manager e si occupa di mansioni molto elementari che possono essere codificate in relativamente poco spazio, come, ad esempio, ricercare il primo volume di sistema sul disco di boot, cercare, caricare il secondo stadio, più complesso ed esteso, in memoria e passargli il controllo;
- lo **stage 2** è il vero e proprio gestore di avvio del programma esecutivo o del sistema operativo, è molto più sofisticato, complesso ed esteso del primo e si occupa principalmente di scegliere quale kernel caricare in memoria per poi cedergli il controllo; esso comincia ad assomigliare ad un vero e proprio sistema operativo, per quanto minimale, sebbene sia più corretto chiamarlo *programma esecutivo*, dal momento che manca di molte delle caratteristiche dei sistemi operativi moderni, come ad esempio la multiprogrammazione e la gestione delle risorse.

Si tratta di una suddivisione logica che rimarca

diversi momenti della fase di avvio. Quanto alle limitazioni, per esempio vi è la possibilità di caricare dal dispositivo di boot nella memoria centrale solo una piccola quantità di byte e trovarla posizionata in un punto piuttosto sconveniente della memoria, come accade nei sistemi x86, dai primi 512 byte a partire dall'indirizzo 0x0000:0x7C00 in avanti (ultimo byte in 0x0000:0x7DFF).

Vale la pena di specificare che un singolo bootloader è in grado di caricare molti tipi kernel (e quindi molti sistemi operativi) differenti su partizioni diverse. Naturalmente può essere caricato un solo kernel per volta.

SUI MICROCONTROLLORI

Lo sviluppatore embedded si occupa di scrivere codice che deve essere caricato e messo in esecuzione su dispositivi molto più piccoli di un Personal Computer. E allora, cosa cambia tra un microprocessore e un microcontrollore? Il bootloader è lo stesso?

Naturalmente no! Il bootloader svolge funzioni simili, lavora in modo equivalente, ma quello per i microcontrollori è molto più semplice.

Infatti, un bootloader è un programma che viene eseguito all'interno del microcontrollore al fine di poterlo programmare. Sembra una definizione un po' ricorsiva ma fondamentalmente si tratta di una routine che riceve il nuovo codice da una sorgente esterna, attraverso un bus di comunicazione dedicato, al fine di poterlo correttamente caricare nella memoria del microcontrollore.

Una delle nozioni che è importante tenere a mente della programmazione di un microcontrollore tramite componenti esterni specializzati e dedicati è il fatto che funziona a prescindere dal contenuto già presente all'interno. Infatti, i microcontrollori iniziano il processo di riscrittura delle aree di memoria in maniera del tutto agnostica rispetto al contenuto precedentemente presente all'interno della loro memoria.

Previo opportuna verifica che la dimensione del nuovo firmware possa essere recepita per intero dal microcontrollore in oggetto, alla circuiteria già presente è possibile pensare di affiancare hardware dedicato alla scrittura.

L'interfacciamento potrebbe essere svolto in maniera diretta, se i componenti elettronici fossero saldati direttamente sulla scheda, oppure tramite interfacce con un PC, per esempio RS232, USB o magari Ethernet. Nel caso si desideri utilizzare un computer, il programma che dialoga con il bootloader dovrebbe essere opportunamente configurato per inviare un file in formato *.bin oppure di tipo *.hex.

Detta così, in effetti, sembra davvero molto semplice. Ma che succede se c'è un errore nella comunicazione? Che succede se i dati, oppure parte di essi, vengono persi? E se per caso la comunicazione viene persa, che cosa accade alle aree di memoria? E se c'è un bug oppure il bootloader si corrompe? Queste sono tutte problematiche che vengono gestite proprio dal bootloader.

Si parte con una prima istruzione finalizzata al reset del microcontrollore, che è fondamentale per interrompere l'esecuzione del codice eventualmente già presente. Il bootloader comunica con l'host, il quale risponde con l'eventuale nuovo codice; non appena quest'ultimo viene inviato, quello precedentemente presente in memoria viene sovrascritto.

All'interno di ciascun codice è sempre inserito un meccanismo di verifica che prevede un checksum, per il controllo dell'integrità dei dati; questa è una delle funzionalità che viene affiancata al semplice bootloader. Nel caso in cui l'immagine ricevuta non sia effettivamente intatta, deve poter essere gestito il caso di sostituzione o ripristino della vecchia immagine. Proprio per questo motivo, solitamente sono previsti due moduli distinti: l'uploader e il bootloader. Il primo, infatti, è responsabile della scrittura all'interno della memoria della nuova immagine solo nel caso in cui essa sia stata ricevuta intatta e sia stato effettivamente scelto di sostituire la vecchia immagine con quella aggiornata.

L'Uploader è parte dell'applicazione principale ed interagisce con l'host al fine di poter caricare un nuovo firmware. Ciò richiede aree di memorie separate che devono essere necessariamente distinte da quelle dove c'è il programma principale. A tale scopo si utilizzano, per esempio, memorie EEPROM esterne collegate tramite interfaccia seriale. È anche possibile che siano impiegati altri microcontrollori, che dispongono di ulteriore spazio di memorizzazione disponibile.

L'Uploader si occupa "semplicemente" della scrittura del nuovo file ricevuto ma non è coinvolto nella sua esecuzione. Nel momento in cui è stato effettuato il caricamento, perché il programma possa essere eseguito, esso ha bisogno che il bootloader sia effettivamente in esecuzione; quest'ultimo, infatti, si occuperà del controllo della memoria, della gestione delle istruzioni e della loro esecuzione sequenziale.

È importante notare che ciascun microcontrollore differisce dagli altri, in molti casi anche all'interno della stessa famiglia (per esempio la PIC 18F); per questo motivo, non è possibile pensare ad un bootloader che vada bene per tutti microcontrol-

lori esistenti. La cosiddetta soluzione "one size fits all" non esiste per via del fatto che i comandi sono differenziati, le aree di memoria hanno indirizzi diversi, le istruzioni necessarie sono differenti e, sebbene le operazioni elementari, atomicamente in senso informatico coincidano (per esempio l'arresto del microcontrollore, il reset ed il riavvio) servono routine differenziate affinché ciascuna possa essere eseguita correttamente.

Le soluzioni a queste problematiche, globalmente, possono essere quelle descritte qui di seguito.

- **Basic:** il dispositivo ha una linea di comunicazione seriale e vuole connettersi ad un host. Il bootloader viene eseguito a partire dall'istruzione di reset per poi iniziare uno scambio di messaggi al fine di iniziare la scrittura delle aree di memoria per poi eseguire il nuovo codice. Di solito, dopo un intervallo temporale di 500 ms in cui non ci sono state comunicazioni complete (per esempio, non sono avvenute risposte), il bootloader smette di effettuare "tentativi" ed esegue il codice caricato in precedenza nelle aree di memoria. In tal caso, il flashing dovrà essere eseguito nuovamente.
- **Program memory uploader:** in questo caso, la memoria disponibile sul microcontrollore viene divisa in maniera tale da avere sia il programma già in memoria, sia la nuova immagine sia il bootloader. Quest'ultimo funzionerà in maniera tale da effettuare il reset ed il caricamento della nuova immagine nelle prime aree di memoria.
- **External EEPROM Image:** come nel caso precedente, è possibile pensare all'utilizzo di una memoria esterna come repository della nuova applicazione, tuttavia, fondamentalmente non cambia molto rispetto al caso precedente, a meno che non si stia pensando di cambiare il microprocessore o il microcontrollore in favore di uno con una maggiore disponibilità di memoria oppure con un tipo o numero di periferiche differente.
- **TCP Bootloader.** Esso implica l'utilizzo del protocollo TCP, e quindi di una connessione tramite indirizzo IP, per il caricamento dell'immagine. Tipicamente, questa modalità di caricamento viene scelta nel caso in cui si utilizzino macchine server dedicate per l'intero sistema. È possibile che questa modalità sia impiegata quando bisogna programmare più microcontrollori diversi. In questo caso, prima di procedere alla scrittura del dispositivo successivo, verrà effettuato un controllo delle operazioni svolte. Ciò pone alcuni problemi tecnici in termini di eventuali

ritardi nelle verifiche, che devono poter essere gestite opportunamente. Naturalmente, lo stesso bootloader, in questo caso, richiede di essere aggiornabile "sul campo", il che implica che esso stesso deve contenere lo stack TCP.

Ciascuna di queste possibilità, inoltre, deve essere scelta sulla base di un'accurata valutazione di parametri quali:

- la velocità di trasmissione;
- il costo dei componenti;
- la facilità d'uso;
- la manutenibilità.

BOOTLOADER SU ARM CORTEX

Una delle caratteristiche più interessanti dei microcontrollori della famiglia STM32 di ST Microelectronics è la presenza del bootloader built-in, almeno per quanto riguarda la loro revisione più recente. Per tutti coloro che di programmazione, scrittura di firmware e debug fossero un po' a digiuno, precisiamo che si tratta di una delle funzionalità più utili che lo sviluppatore possa immaginare, perché software e hardware necessari sono già presenti e vanno solo configurati.

In particolare, è noto come e perché l'interfaccia JTAG rappresenti uno dei principali, se non il principale, standard di riferimento per le comunicazioni e l'interfacciamento ai microcontrollori: essa viene innanzitutto utilizzata per effettuare la programmazione del microcontrollore. Dopo che il codice è stato compilato, infatti, esso può essere caricato utilizzando l'interfaccia JTAG sul microcontrollore tramite una sequenza di controllo specifica a seconda della versione e della revisione del singolo circuito integrato.

Ma i vantaggi nell'utilizzo di questa interfaccia riguardano soprattutto il controllo ed il monitoraggio del funzionamento, anche a Runtime; il JTAG, infatti, consente l'accesso diretto alle aree di memoria del microcontrollore in maniera precisa, ovvero consente di leggere, byte per byte, il contenuto presente.

Concordemente è anche possibile effettuare controlli per la verifica dell'integrità delle stesse aree di memoria, così da verificare che siano sempre accessibili e che siano integri anche i dati contenuti al loro interno.

Grazie al JTAG, inoltre, è possibile verificare anche le funzionalità del microcontrollore in termini di memoria RAM e, velocità di accesso e comunicazione. Per poter portare a termine tutte queste operazioni, però, di solito è necessario utilizzare:

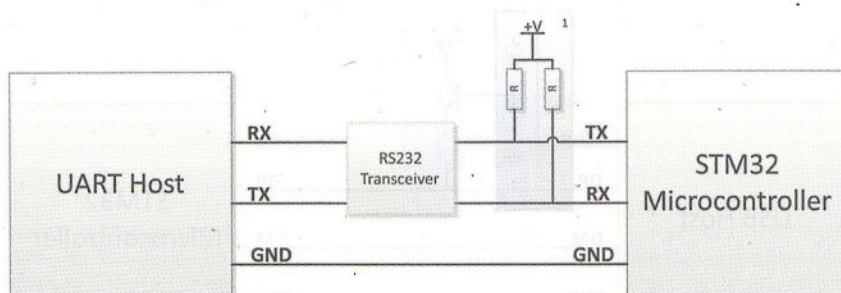


Fig. 1
Connessione tramite
interfaccia USART.

- (opzionali) componenti hardware aggiuntivi;
- un tool software dedicato;
- un header (connettore fisico) aggiuntivo che consente l'interfacciamento fisico, qualora la scheda in oggetto non lo preveda.

Questo elenco risulta completo per tutti i dispositivi che siano già dotati di bootloader, il che però non è sempre vero. Esistono, infatti, microcontrollori che ne sono dotati ed altri che non hanno il bootloader interno. È molto interessante ed utile, quindi, capire che cosa cambia in un caso o nell'altro.

Una delle funzionalità avanzate più interessanti abilitate per l'inclusione del bootloader sul microcontrollore è il cosiddetto "debug in-circuit": esso consente di eseguire la verifica del funzionamento senza dover interrompere una routine in esecuzione. È possibile, in particolare, far procedere il microcontrollore nel suo funzionamento in modalità step-by-step, così sarà possibile verificare opportunamente ogni singola operazione, ogni singola funzione chiamata ed il valore di ciascuna variabile. Questo tipo di funzionalità è estremamente utile quando si cerca di verificare il corretto funzionamento del circuito, ma soprattutto consente di isolare un eventuale malfunzionamento, identificandone più precisamente la causa.

La famiglia STM32 è dotata di questa funzionalità ed è molto interessante vedere come può essere utilizzata. Dal punto di vista fisico, ci sono due pin che possono essere impiegati per decidere la sorgente del codice: essi possono essere utilizzati in configurazioni differenti che restano funzionalmente allocate. In particolare, abbiamo:

Se scartiamo l'idea di eseguire il codice che risiede nella memoria RAM, Boot1 può essere mantenuto a livello logico basso e possiamo aggiungere uno switch oppure un jumper a BOOT0.

Se dopo il reset BOOT0 è a livello logico alto, il processore si avvierà utilizzando il codice che risiede nella memoria di sistema: in pratica, direttamente dal bootloader. Mantenendo il pin a livello logico

BOOT1	BOOT0	Boot Mode
X	0	Boot from user flash
0	1	Boot from System Memory
1	1	Boot from embedded RAM

basso, il sistema si avvia con il codice contenuto nella memoria Flash.

Una volta capito come effettuare la scelta, è necessario studiare la realizzazione dei collegamenti. Per farlo, è indispensabile prima di tutto capire che tipo di interfaccia si vuole utilizzare. In particolare, è possibile scegliere se basarsi su USART, USB, SPI, I²C o CAN.

Se vogliamo utilizzare l'interfaccia USART per accedere al bootloader, è necessario connettere i pin Tx ed Rx attraverso un cavo seriale (Fig. 1). È fondamentale notare che devono essere impiegati resistori di pull-up perché tutto funzioni correttamente. In difetto, il livello alto non potrà mai essere raggiunto e, di conseguenza, letto.

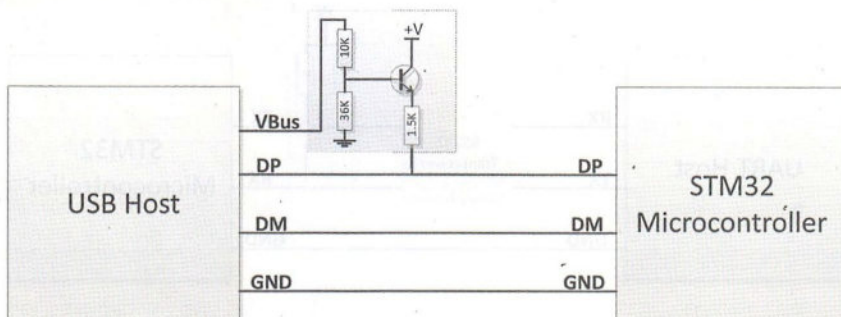
Il corretto dimensionamento del resistore di pull-up è un fattore della fase di progettazione che viene, talvolta, sottovalutato. Fate molta attenzione al livello logico ed al valore di corrente che volete che il resistore garantisca, onde evitare ogni genere di problematica.

Vale la pena di evidenziare che non è possibile dialogare con il micro direttamente, ma è necessario utilizzare un componente intermedio, indicato nella Fig. 1 come RS232 Transceiver. Il suo scopo è quello di fungere da interprete nella comunicazione.

Per tutti coloro che non avessero familiarità con questo tipo di interfaccia, è anche importante notare che i pin Tx ed Rx sono indipendenti, proprio per via del fatto che le linee sono dedicate.

Esiste anche una modalità di connessione che prende nome di DFU, acronimo di Device Firmware Update. In questa modalità, com'è evidente dalla Fig. 2, il microcontrollore viene interfacciato direttamente all'USB Host. Si vede chiaramente che l'alimentazione del bus richiede dei componenti dedicati, che implementano una rete di pull-up. In questo caso la differenza sta nel fatto che la rete è

Fig. 2 - Connessione tramite interfaccia USB.



dedicata all'alimentazione e non alle linee dati. Quando si utilizza l'PC, invece, la configurazione prevede l'utilizzo di due linee dedicate, delle quali una serve per trasmettere i segnali di sincronizzazione (SCL) e l'altra per la trasmissione dei dati (SDA).

Anche in questo caso, come nella prima configurazione, i componenti che implementano le semplici reti di pull-up sono dedicati alle linee dati (Fig. 3). Con l'interfaccia SPI i collegamenti sono, ancora una volta, composti da una linea dedicata alla sincronizzazione, e quindi ai segnali di clock (SCK), e da due linee dedicate alla trasmissione dati. In questo caso, in particolare, l'interfaccia prevede che il dispositivo che funge da Host sia il Master della comunicazione mentre quello cui la trasmissione dati è "dedicata" si comporti da Slave. Viene tipicamente impiegata una semplice resistenza che riferisce a massa il valore del segnale di clock (Fig. 4). Nella Fig. 5, infine, viene mostrata la connessione tramite interfaccia CAN. Qui l'Host comunica con il dispositivo tramite l'utilizzo di linee dedicate che attraversano due transceiver e un cavo seriale. Tipicamente può essere utile l'impiego di un resistore da 120 ohm. Vale la pena di notare che l'esecuzione delle operazioni in questa fase prevede una serie di condizioni e controlli che sono riassunti nella Fig. 6. È proprio nella sequenza di operazioni che tale modalità di funzionamento risulta peculiare.

L'intera sequenza di avvio, infatti, prevede, a valle della scelta dell'interfaccia di comunicazione da utilizzare, una sorta di handshake per la sincronizzazione. Segue la verifica dell'indirizzo, in questo caso l'PC, e l'interfaccia di comunicazione prescelta per la programmazione, per cui, ovviamente, bisognerà controllare la specificità dell'indirizzo.

Non appena viene stabilito che il target è quello corretto, si passa a disabilitare ogni sorgente di interrupt per poter interrompere l'esecuzione di qualsiasi operazione, per poi successivamente abilitare l'interfaccia alla comunicazione. Questa è una fase preliminare che dipende fortemente dalla famiglia di appartenenza del microcontrollore. In particolare, in questo caso, stiamo parlando di STM32F030xC. È molto importante notare che analoghe considerazioni, rispetto a quelle viste finora, possono essere fatte sulla produzione software provvista da diversi produttori, da Texas Instruments ad Analog Device passando per NXP ed Atmel. Dal momento che stiamo analizzando la dotazione software proposta da ST, ci focalizziamo su quanto offerto da questa compagnia. In particolare, esiste una soluzione software che consente di effettuare il cosiddetto flash loader dell'applicazione tramite interfaccia grafica, su Windows. Naturalmente è possibile anche utilizzarlo da riga di comando. Si tratta di uno script Python che potete trovare facilmente disponibile presso questo link: <https://github.com/jfsnyder/stm32loader>. Non vi

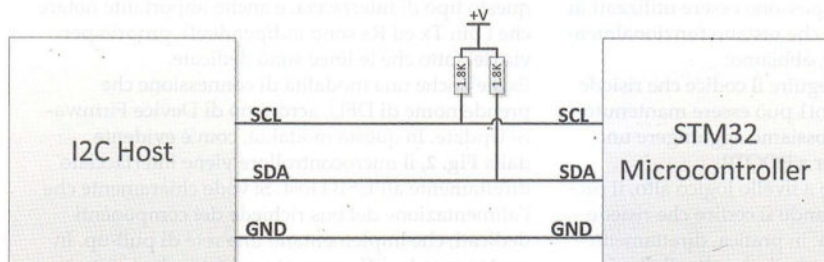


Fig. 3 - Connessione tramite interfaccia PC.

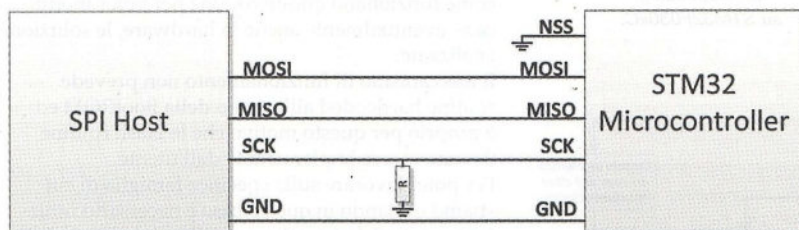


Fig. 4 - Connessione tramite interfaccia SPI.

sarà sfuggito che si tratta di un repository su GitHub, una piattaforma che ormai viene utilizzata in maniera sempre più pervasiva dalla maggior parte degli sviluppatori, soprattutto perché consente la redazione collaborativa del codice. Leggendo la documentazione potrete facilmente rendervi conto che, al fine di utilizzare questa routine, è necessario che abbiate installato Python. Per farlo, vi consigliamo di utilizzare come riferimento il sito ufficiale: <https://www.python.it/download/>. Naturalmente potete eseguire uno script Python in qualunque ambiente, ovvero sistema operativo. Il prototipo della chiamata è il seguente:

```
stm32loader.py [-hqVewvr] [-l length] [-p port]
[-b baud] [-a addr] [file.bin]
```

e nella **Tabella 1**, tutti i parametri sono presentati nel dettaglio e spiegati. Come tutte le routine, anche questa ha delle impostazioni predefinite, in particolare quelle riguardanti la porta seriale; a riguardo è stata fatta la scelta di fissare il valore del baud-rate a 115.200, il che velocizza di molto il trasferimento dati rispetto ad un valore di 9.600, per esempio. Tuttavia è sempre possibile ridefinire sia l'indirizzo della porta, sia il numero di simboli al secondo inviati tramite interfaccia seriale. Ciò dipende dal fatto che non sempre e non tutti i dispositivi sono necessariamente in grado di sostenere tali caratteri-

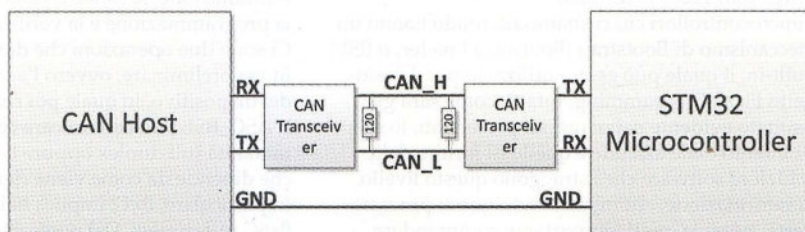
Tabella 1 - Parametri della chiamata.

-h	Help
-q	Quiet
-V	Verbose
-e	Erase
-w	Write
-v	Verify
-r	Read
-l	Length
-p	Serial Port
-b	Baud rate (speed, default: 115200)
-a	Target Address

stiche logiche. Per questo motivo è stata resa possibile la ridefinizione della porta, attraverso l'utilizzo del parametro "p", e del baud-rate, tramite "b". L'output della console suggerisce anche che, qualunque sia il numero e il tipo di parametri che volete definire, l'ultimo campo da riempire sarà il nome del file binario che volete caricare sul vostro microcontrollore.

Una volta che sarete entrati nella modalità adatta per effettuare il caricamento, il chip resterà in attesa dei dati provenienti dal computer. Dal momento che il baud-rate è un parametro fondamentale in questo tipo di comunicazioni, sappiate che se non lo specificherete verrà impostato automaticamente. Questo tipo di funzionalità è ottima perché le comunicazioni sono autoconsistenti ed ottimizzate, soprattutto nei casi in cui i programmatori siano

Fig. 5 - Connessione tramite interfaccia CAN.



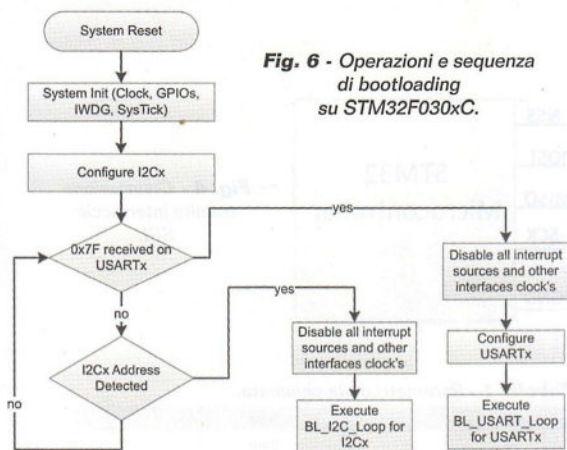


Fig. 6 - Operazioni e sequenza di bootloading su STM32F030xC.

meno esperti nel controllare queste impostazioni. Per utilizzare questa routine Python è importante selezionare la modalità BOOT0 high ed eseguire lo script. La procedura ha una sua durata temporale che, ovviamente, dipende dalla dimensione del firmware ma anche dal numero di simboli trasmessi per unità di tempo. Maggiore sarà il valore del baud rate, minore sarà il tempo impiegato per il trasferimento di dati, ovviamente a parità di dimensioni del firmware.

Una volta completata la procedura, è possibile portare BOOT0 a livello logico basso ed effettuare un reset della scheda. Fatto ciò, come in tutti i microcontrollori, verrà ripresa l'esecuzione periodica e continua della routine caricata all'interno.

Per chiunque voglia approfondire ulteriormente, sappiate che la documentazione ufficiale del bootloader built-in è disponibile nell'Application Note AN2606 <http://www.st.com/stonline/products/literature/an/13801.pdf>.

UN ESEMPIO CONCRETO: BSL XMC1000

Quando si prende in considerazione la serie XMC1000 di Infineon, le cose cambiano leggermente; per dimostrarvelo focalizzeremo la nostra attenzione sull'applicazione di codici in uso su XMC1100/1200/1300/1400.

I microcontrollori cui ci stiamo riferendo hanno un meccanismo di Bootstrap (Bootstrap Loader, o BSL) built-in, il quale può essere utilizzato per il cosiddetto Flash Programming. Infatti, come sarà già risultato evidente nei paragrafi precedenti, lo scopo e l'intento delle aziende è quello di fornire delle soluzioni software che astraggono questo livello di informazione, automatizzando questi processi. Resta, tuttavia, molto importante comprendere

come funzionano queste routine per poter modificare, eventualmente anche in hardware, le soluzioni analizzate.

Il meccanismo di funzionamento non prevede routine hardcoded all'interno della BootROM ed è proprio per questo motivo che le Flash routine devono essere implementate dall'utente.

Per poter lavorare sulla specifica famiglia di cui stiamo parlando in questo caso è necessario utilizzare una toolchain sviluppata direttamente da Infineon, DAVE (Development Platform), attualmente giunta alla sua versione 4.

La famiglia XMC1000 supporta efficacemente sia la modalità BSL Asynchronous Serial Interface (ASCI) sia la Synchronous Serial Interface (SSC).

In generale, quando il dispositivo sarà connesso al computer tramite interfaccia ASCI, saranno utilizzate due sezioni in particolare:

- il Flash Loader program, che viene inviato al dispositivo utilizzando il meccanismo di Bootstrap loading *built.in*. Una volta che il programma viene inviato ed eseguito, il Flash Loader Program stabilisce la comunicazione per ricevere i comandi dall'Host Program in esecuzione sul Personal Computer;
- l'Host Program, invece, in esecuzione sul PC, utilizza il protocollo di comunicazione definito per inviare i comandi e gli specifici codici da programmare. Va da sé che questo significa che si tratta di un programma scritto per la specifica applicazione e, pertanto, richiederà una riscrittura ed, eventualmente, una reinterpretazione qualora si cambiasse l'interfaccia oppure il microcontrollore di riferimento.

Ed ora, entriamo nel dettaglio e del bootstrap.

Nella modalità ASCI_BSL, l'utente può scaricare il suo programma di monitoraggio all'interno della memoria SRAM dell'XMC1000 a partire da uno specificato indirizzo di memoria, in particolare 0x20000200. Una volta ricevuto e memorizzato, il software verrà eseguito in maniera tale da abilitare le comunicazioni con il PC che funge da HOST per l'installazione, la cancellazione della memoria Flash, la programmazione e la verifica dei dati.

Ci sono due operazioni che devono essere eseguite in via preliminare, ovvero l'accensione e il reset del dispositivo, in quale poi rimarrà in attesa.

L'ASCI_BSL può comunicare con il PC Host in modalità full duplex oppure half duplex, una scelta che dipende da come viene riempito l'Header Byte. In particolare, 0x12 implica half-duplex mentre 0x6C full-duplex. Dal nome risulta abbastanza

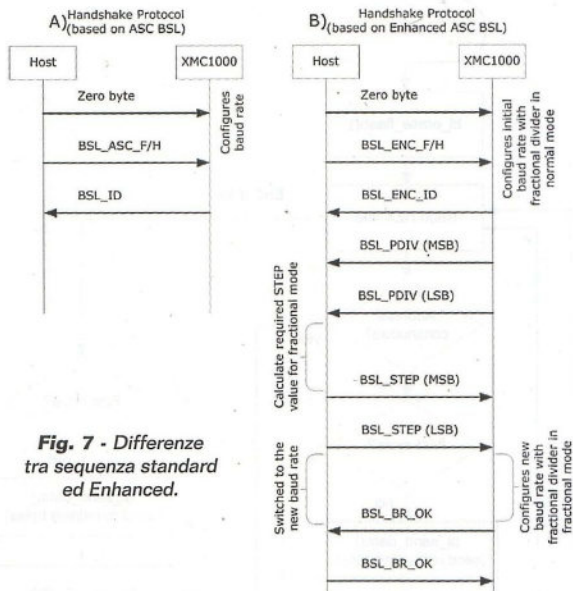


Fig. 7 - Differenze tra sequenza standard ed Enhanced.

immediato comprendere quali siano le differenze tra modalità asincrona e sincrona, viste le differenze delle necessità e dei requisiti delle comunicazioni asincrone rispetto a quelle sincrone.

L'ASC loader supporta efficacemente entrambe le modalità attraverso il protocollo UART.

Ci sono in realtà due modalità in cui funziona l'ASC Bootstrap Loader, ovvero: standard ed Enhanced; la differenza principale tra le due è che nel caso Enhanced, sono supportati baud-rate decisamente maggiori.

A prescindere da quale modalità si decida di utilizzare, in entrambi è necessario:

- rilevare il baud-rate ed eseguire la sequenza di mode selection;
- stabilire la sequenza di download.

Per quanto concerne la prima, l'interazione tra

il bootloader e l'Host avviene tramite protocollo, quindi vi è un handshake, seguito da request/Acknowledgment/data.

Nella Fig. 7 si vede chiaramente che c'è una grande differenza tra la modalità standard, molto più semplice, e quella Enhanced, nella quale viene anche negoziato un eventuale nuovo valore per il baud-rate.

Dopo che sono stati stabiliti il valore del baud rate nonché channel/mode, l'ASC Bootstrap resta in attesa di 4 byte che comunicano la lunghezza dell'applicazione, ovvero la sua dimensione. Segue, quindi, la fase due, ovvero la sequenza di download, come viene mostrato nella Fig. 8.

Il Flash Loader previsto per la famiglia XMC1000 implementa delle routine che servono per stabilire la comunicazione tra il PC ed il target device, ed implementa le seguenti funzionalità:

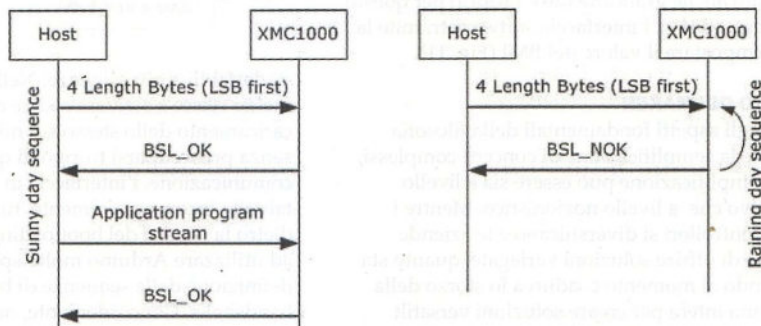
- cancellazione delle aree di memoria;
- cancellazione, programmazione e verifica delle pagine di memoria programmate;
- cambio della modalità Boot Mode Index (BMI) sul dispositivo.

Vediamo, in particolare, alcune delle procedure richiamate: nel caso della Flash Erase (Fig. 9), ciascun settore della memoria viene controllato al fine di verificare il fatto che sia pieno oppure no. Nel caso in cui non lo sia, viene impostata la modalità opportuna e si passa poi al settore successivo. La procedura viene iterata fino alla completa dimensione del file. Dopo aver eseguito l'operazione di Flash Erase, viene letto il file per poter verificare innanzitutto le aree di memoria, successivamente la loro contiguità ma soprattutto se sia il primo tentativo di scrittura oppure no.

Naturalmente, questo controllo serve per verificare se sono presenti eventuali settori danneggiati.

Se tutti i controlli hanno dato esito negativo, le aree

Fig. 8
Sequenza di download nei due casi.



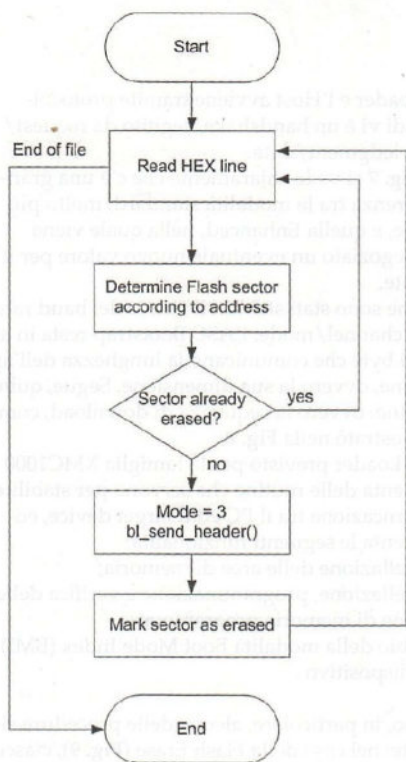


Fig. 9 - Procedura di Flash Erase.

di memoria verranno scritte opportunamente fino alla dimensione desiderata (Fig. 10). L'operazione viene iterata fino al termine della dimensione del file da scrivere. Queste operazioni, che sono sempre indispensabili e comuni a tutti i microcontrollori, vengono eseguite in maniera integrata ed è proprio qui che, almeno su questa famiglia di microcontrollori, risiede la differenza principale. Tutto quello che abbiamo descritto può essere mostrato anche graficamente e proprio per questo vediamo, adesso, l'interfaccia software tramite la quale impostare il valore del BMI (Fig. 11).

IL CASO DEI MAKER

Uno degli aspetti fondamentali della filosofia maker è la semplificazione di concetti complessi; tale esemplificazione può essere sia a livello operativo che a livello nozionistico. Mentre i microcontrollori si diversificano e le aziende cercano di offrire soluzioni variegate, quanto sta capitando al momento coadiuva lo sforzo della comunità intera per creare soluzioni versatili

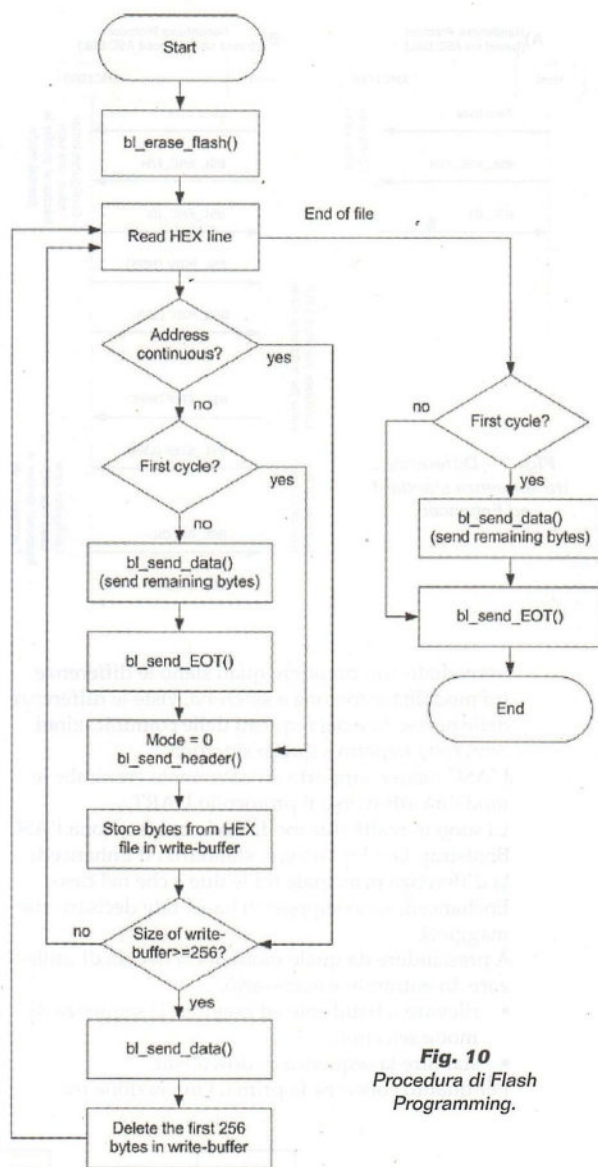


Fig. 10
Procedura di Flash Programming.

e adattabili a più esigenze. Nello specifico, un maker riesce a scrivere codice e ad effettuare il caricamento dello stesso sul microcontrollore anche senza preoccuparsi troppo di quale sia la tecnica di comunicazione, l'interfaccia in uso, ed ignorando, talvolta inconsapevolmente, tutto quello che sta dietro la tecnica del bootloading. Chi comincia ad utilizzare Arduino molto spesso ignora la definizione delle sequenze di boot, o cosa sia un handshake. Concordemente, non è a conoscenza di

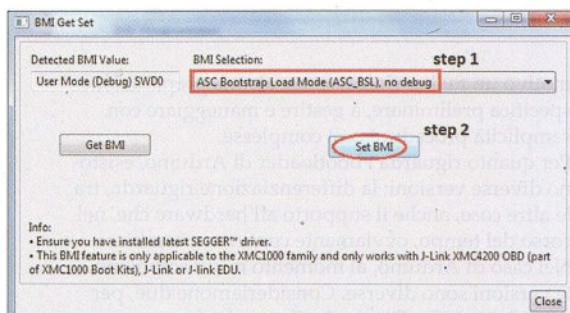


Fig. 11 - Utilizzo del tool dedicato per impostare il valore del BMI su dispositivi XMC1x00.

molte altre grandezze variabili, funzionalità di cui finora abbiamo parlato nel dettaglio. Ciò non vuol dire che vi sia realmente la possibilità di ignorare tali entità. Esse restano comunque indispensabili.

Come fa Arduino a garantire un approccio "maker" al problema?

La risposta la possiamo dare studiando nel dettaglio come fare il condizionamento di un microcontrollore a bordo di un Arduino UNO, ovvero l'ATMega328. Vediamo, dunque, come un maker si appropria a questo tipo di problematiche, risolvendo il problema del caricamento del firmware sul micro stesso.

La Fig. 12 riporta lo schema per la configurazione essenziale del microcontrollore, che possiamo direttamente confrontare con le indicazioni per il caricamento del bootloader visibili nell'immagine in Fig. 13. Ma a cosa servono questi schemi? E come si usano? Ebbene, come abbiamo già accennato, i

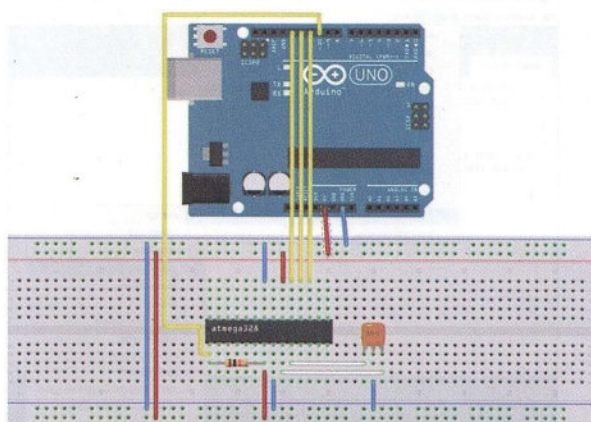


Fig. 12 - Schema di principio per il collegamento per la programmazione di un microcontrollore.

microcontrollori vengono, di solito, programmati attraverso un programmatore dedicato, a meno che non ci sia un firmware all'interno del microcontrollore che consenta l'installazione di nuove funzionalità o la riscrittura del firmware presente attraverso un componente esterno. Quest'ultimo è proprio il bootloader.

Se non si vuole utilizzare lo spazio in memoria Flash, e si vuole anche evitare di dover gestire problematiche di ritardo dovute proprio al bootloader, è possibile utilizzare un programmatore esterno. Per poter effettuare il flashing del bootloader, è necessario dotarsi di un AVR-ISP (quest'ultimo è una sigla, un acronimo che sta per In-System Programmer), USBtinyISP, oppure creare un cosiddetto "Parallel Programmer". Qualunque soluzione tra

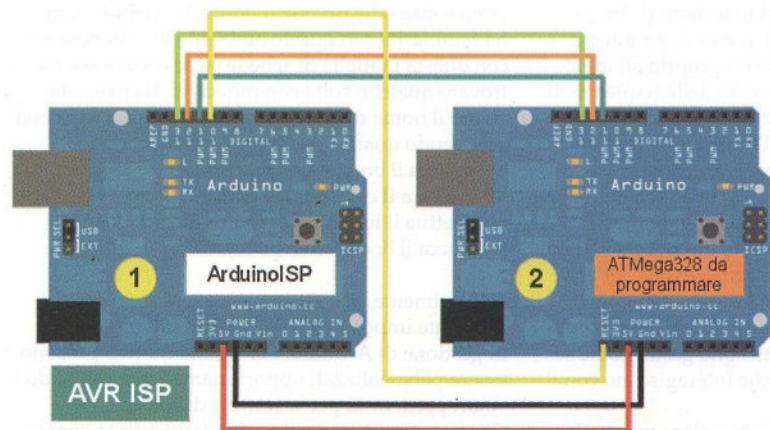


Fig. 13 - Schema di collegamento per la programmazione di un microcontrollore utilizzando Arduino UNO come programmatore.

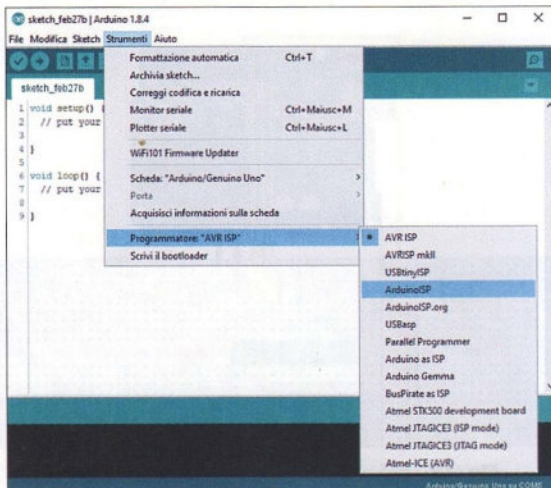


Fig. 14 - Selezione di Arduino ISP dall'IDE di Arduino, utilizzato come ambiente di programmazione per un microcontrollore.

queste scegliate, sappiate che dovrete utilizzare il connettore ICSP a 6 pin (disposti in due file da tre). Utilizzare questo connettore richiederà prima di tutto che stiate molto attenti all'orientamento perché ciascuno dei pin disponibili ha una sua specifica funzione; e comunque, tenete presente che avrete bisogno di una alimentazione esterna per la scheda. Una volta fatto ciò, dovrete semplicemente utilizzare l'IDE di Arduino, selezionando la scheda corretta dalla sezione "Strumenti", per poi scegliere l'opzione "Burn Bootloader".

Perché Arduino rende possibile questa semplificazione?

Perché l'IDE di Arduino non integra soltanto un interprete dei comandi o alcune delle funzionalità più utili per i programmatori in termini di leggibilità del codice. Esso, infatti, possiede ed integra una serie di funzionalità inerenti proprio all'interpretazione del codice ed all'avvio delle sequenze di caricamento dei file compilati sulla scheda. Senza contare, peraltro, che l'interazione tra l'utente e la scheda Arduino viene gestita come se fosse più ad alto livello, dal momento che lo scambio di messaggi avviene tramite un chip integrato che gestisce le comunicazioni tramite interfaccia RS232. Per quest'ultimo, inoltre, va segnalato che, all'atto dell'installazione dell'IDE di Arduino, vengono installati driver dedicati e, pertanto, processi complessi come quelli già descritti sono gestiti come se fossero semplici periferiche che interagiscono con il sistema operativo. Questo grado di semplificazione spiega per quale

motivo un maker riesca, senza alcuna preparazione specifica preliminare, a gestire e maneggiare con semplicità procedure così complesse.

Per quanto riguarda i bootloader di Arduino, esistono diverse versioni: la differenziazione riguarda, tra le altre cose, anche il supporto all'hardware che, nel corso del tempo, ovviamente continua a cambiare. Nel caso di Arduino, al momento in cui scriviamo, le versioni sono diverse. Consideriamone due, per semplicità: NG e Diecimila. Entrambe funzionano con un baud rate pari a 19200 ed occupano circa 2 kB di memoria Flash su ATmega168 (NG). La differenza fondamentale sta nel tempo di attesa della nuova versione del firmware e nel numero di volte che viene acceso il LED connesso al pin 13 della scheda. Dal momento che nella versione Diecimila (e successive), viene implementato il reset automatico, il suo bootloader deve aspettare un tempo veramente breve, inferiore ad un secondo, per il salvataggio. Concordemente, accende il LED connesso al pin 13 soltanto una volta. Nel caso dell'NG, occorrono dai sei agli otto secondi ed il LED viene acceso tre volte.

Versioni più vecchie, sulle prime distribuzioni delle schede della famiglia Arduino, avevano un baud-rate decisamente più basso, impostato al classico valore di 9.600. Naturalmente ci riferiamo a quelle ufficiali. È sempre possibile, ovviamente, utilizzare i bootloader esterni, prodotti da terze parti o da altri utenti della comunità. Naturalmente bisognerà tenere conto delle differenze e studiarlo opportunamente prima dell'impiego in maniera tale da verificare le impostazioni di frasi e la loro modificabilità.

In ambiente Arduino, al comando "Scrivi il bootloader" viene associata all'esecuzione di un tool open source che prende il nome di avrdude. Chi ha familiarità con questo ambiente di sviluppo o con questa famiglia di schede sicuramente si sarà trovato qualche volta con messaggi di errore che citano il nome di questa routine. Avrdude funziona eseguendo quattro passaggi fondamentali:

- sblocca il bootloader sul chip;
- imposta il ciclo per la scrittura;
- effettua il loading del codice del bootloader;
- blocca il bootloader.

Naturalmente, le quattro operazioni sono gestite mediante impostazione fornite all'interno dei file di gestione di Arduino. Per questo motivo possono essere personalizzati opportunamente, cambiando i valori predefiniti per le routine di sblocco. Concretamente, per poter effettuare questa opera-

zione, è necessario utilizzare la routine ArduinoISP (Fig. 14). Il nome di questa modalità, se vogliamo, suggerisce che sia possibile utilizzare una generica board Arduino come programmatore per altri microcontrollori, sempre di tipo AVR.

La routine può essere impiegata con successo con alcuni, piccoli e semplici accorgimenti. In particolare, come dicevamo in precedenza, nelle versioni più recenti del bootloader è stata implementata una funzione di autoreset che serve proprio per garantire l'interruzione dell'esecuzione di ogni operazione in corso da parte del microcontrollore, l'halt e l'abilitazione alla scrittura.

Per gestire opportunamente l'eventuale conflitto con questa funzione automatica, è possibile utilizzare un circuito molto semplice con una rete resistiva composta da una resistenza di valore pari a 120 Ω e da un condensatore da 10 μF connessi direttamente al pin di reset (per intenderci, il primo della sezione di alimentazione sulla versione UNO R3). Naturalmente, come per tutti i valori nominali dei componenti discreti, entrano in gioco anche una serie di considerazioni circa le tolleranze dei valori sia della resistenza sia del condensatore. Fondamentalmente, però, bastano resistenze di valore abbastanza contenuto, per intenderci al di sotto dei 200 Ω , per risolvere il problema.

La soluzione più elegante e completa al problema dell'utilizzo di una board Arduino come programmatore per altri microcontrollori è costituita dall'utilizzo della scheda nella configurazione mostrata in Fig. 13. Una delle due schede, che funzionerà da programmatore, viene predisposta con a bordo il firmware ArduinoISP. L'altra, invece, diventerà il target su cui caricare il nuovo bootloader, oppure aggiornare quello precedentemente presente.

Vale la pena di farvi notare che la versione UNO R3 consente la sostituzione del micro, dal momento che non si tratta di un componente di tipo SMD, ovvero a montaggio superficiale, ma di un integrato distribuito in package PDIP connesso tramite l'utilizzo di apposito socket; questo facilita l'utilizzo da programmatore. Ovviamente, la scheda che fungerà da "sorgente" va collegata al PC ed è alimentata tramite USB. La medesima alimentazione viene poi anche provvista alla scheda di destinazione. Fatti i collegamenti fisici, molto semplici in verità, si passa all'utilizzo di Arduino IDE, all'interno di questo ambiente è indispensabile verificare la versione della scheda connessa e l'indicazione della porta seriale tramite il menu "strumenti". Una volta selezionato ArduinoISP, alla fine della procedura la nostra scheda sarà effettivamente diventata

la "sorgente" e conterrà il firmware. Non servirà più riprogrammarla, ammesso che non dobbiate utilizzarla per altri scopi.

Ora è possibile programmare il microcontrollore AVR Atmel alloggiandolo correttamente nel socket della scheda di destinazione, collegare la scheda in questione ed osservare il trasferimento dei dati attraverso interfaccia seriale grazie ai LED Tx ed Rx che servono proprio a dare un feedback visivo delle trasmissioni in corso.

Da questa breve descrizione, risulta evidente, come il settore "maker" si approcci ad un problema estremamente complesso godendo della possibilità di utilizzare strumenti molto semplici, modulari che sfruttano con facilità tutte le potenzialità dei microcontrollori. Il fatto che si possa utilizzare una coppia di integrati dello stesso tipo per due funzioni diverse comunica dei messaggi importanti: sul fatto che la programmazione faccia la differenza, sulla riutilizzabilità dei componenti e sull'importanza e l'efficacia della prototipazione rapida

CONCLUSIONI

Con questo, siamo giunti al termine di questo excursus nel mondo della programmazione embedded. Abbiamo trattato il bootloader, le operazioni e le sequenze di avvio per i microcontrollori da più punti di vista, analizzando la programmazione, l'implementazione ed il condizionamento con diverse soluzioni.

Il vantaggio risulta evidente soprattutto per via del fatto che diversi esemplari della stessa famiglia, diverse famiglie di componenti ma anche diverse generazioni di microcontrollori prevedono metodi e tecniche differenti.

Per l'ingegnere che si affaccia al mondo della programmazione embedded è davvero molto importante conoscere i principi basilari dell'interazione per poi poterli declinare. Infatti, è proprio dalla conoscenza della tecnica, in generale, che deriva una competenza variegata sui microcontrollori, spendibile anche se non si conosce uno specifico micro quando vi ci si appropria per la prima volta.

Inoltre, in questo articolo abbiamo anche visto come e quanto l'approccio e le soluzioni che vengono adottate in ambito "maker" possano avere un impatto significativo in termini di semplificazione e di immediatezza.

In definitiva, possiamo affermare che comprendere il bootloader, nonché la sua realizzazione, è una condizione indispensabile per il corretto utilizzo di un microcontrollore, qualunque esso sia.



Sfoggia on-line il catalogo generale 2018

**FUTURA
ELETTRONICA®**



O DIRETTAMENTE
DAL TUO TABLET
O SMARTPHONE

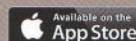


**FUTURA
ELETTRONICA®**

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

Maggiori dettagli e acquisti on-line su
www.futurashop.it

APP SCARICABILE
GRATUITAMENTE DA:



THE TACTIGON
WEARABLE ARDUINO WIRELESS WITH MOTION SENSOR



VLC & THE TACTIGON

Utilizziamo il dispositivo di prototipazione di applicazioni indossabili e rilevamento di gesture tridimensionali, per controllare la riproduzione di una playlist VLC con dei movimenti delle mani.

a cura della Redazione

Abbiamo visto, negli scorsi numeri, come il controller intelligente The Tactigon si interfacci facilmente con diversi tipi di dispositivi, che vanno da uno smartphone Android ad un single board computer come Raspberry Pi 3, dotato di un transceiver Bluetooth Low Energy. In questo articolo utilizziamo The Tactigon in un'applicazione che permette l'interazione con il noto riproduttore multimediale universale VLC Media Player, allo scopo di gestire una playlist. Prima di entrare nel vivo dell'applicazione ricordiamo che The Tactigon è una periferica intelligente (e

per di più Arduino-compatibile) che permette di realizzare controller capaci di rilevare gesture tridimensionali e parametri ambientali come temperatura e pressione, idonei a sviluppare interfacce utente per giochi elettronici, gestione di robot industriali e di servizio, macchinari, droni e radiomodelli avanzati. La board è ideale in generale per la cattura dei movimenti e anche per applicazioni nell'ambito della realtà aumentata. Questo piccolo gioiello tecnologico è governato da un microcontrollore a



32 bit della ST Microelectronics, operante con un clock di 32 MHz, dotato di 80 kB di RAM, 512 kB di memoria di programma e 16 kB di EEPROM. The Tactigon (<https://thetactigon.com/>) è inoltre dotato di connessione wireless Bluetooth Low Energy ed ha a bordo una IMU a 10 gradi di libertà con accelerometro, giroscopio e magnetometro, sensore di temperatura e pressione barometrica.

LA NOSTRA APPLICAZIONE

In questo articolo utilizzeremo The Tactigon per interagire con un'applicazione da computer, invece che per comandare dell'hardware come abbiamo fatto, ad esempio, nel fascicolo di aprile scorso, in cui abbiamo realizzato un controllo di illuminazione a 4 canali via gesture. Più esattamente, The Tactigon funzionerà da controller a gesture del programma VLC per Linux installato in una board Raspberry Pi 3, interfacciato tramite Bluetooth. Abbiamo utilizzato Python 2.7.13 per gestire la ricezione dei comandi Bluetooth Low Energy ed il loro reinoltro a VLC. La libreria utilizzata è *bluepy*, che sfrutta il modulo *BlueZ*.

Nell'articolo pubblicato nel fascicolo di aprile scorso abbiamo spiegato come installare e configurare correttamente questi due software su Raspbian Stretch; senza ripeterci rimandiamo ad esso per gli approfondimenti del caso, ricordandovi comunque di abilitare le funzioni sperimentali per poter sfruttare il Bluetooth Low Energy.

La libreria per il binding delle API VLC, *vlc.py*, è disponibile su GitHub al seguente link: <https://github.com/oaubert/python-vlc>. Questa libreria ci mette a disposizione un'interfaccia con l'applicazione VLC, possiamo utilizzarla tramite script python i comandi che siamo soliti dare tramite interfaccia grafica. È possibile procedere all'installazione tramite PiPy con il comando:

```
pip install python-vlc
```

Avendo mantenuto lo stesso firmware per The Tactigon, abbiamo cambiato solamente il comportamento del codice python, il quale interpreta i comandi seguendo questo protocollo:

- 0x5A5A: **Play/Pause**; inizia la riproduzione della playlist, oppure mette in pausa se già in riproduzione;
- 0x0088: **Stop**; interrompe la riproduzione;
- 0x00BB: **Previous**, ossia il comando passa al brano/video precedente;
- 0x1515: **Next**, ossia il comando che passa al brano/video successivo.

**Guarda The Tactigon
in azione su YouTube!**



Puoi vedere come The Tactigon controlla VLC Media Player alla pagina:

<https://youtu.be/9D142JQp9tY>

Dal nostro canale YouTube

www.youtube.com/user/ElettronicaIN potrai anche vedere i video di moltissimi altri nostri progetti proposti in passato.



È possibile estendere questo protocollo ed avere un controllo maggiore, ma lasciamo a voi l'esercizio di implementare queste funzioni. Ad esempio, utilizzando una diversa gesture è possibile avanzare di un secondo nel brano, impostare lo schermo intero oppure la finestra, e così via.

THE TACTIGON - ARDUINO IDE

Il firmware, ossia lo sketch di The Tactigon, resta sostanzialmente invariato rispetto a quello dell'articolo proposto in aprile scorso e si occupa di rilevare la posizione assunta dalla board desumendola dalle letture dell'accelerometro. Il codice che realizza questa funzione è riportato qui di seguito.

```
if (!lastSupX && accData.x > threshold) {
    lastSupX = true;
    lastInFX = false;
    lastInFY = false;
    lastSupY = false;
    buffData[0] = 0x00;
    buffData[1] = 0xBB;
    flash();
    bleChar.update(buffData);
    justSent = true;
}
```

Questo codice, che viene ripetuto per i due assi x ed y utilizzati, permette di rilevare la posizione esatta della scheda. La variabile booleana *last-*

SupX viene utilizzata, appunto, per evitare che la stessa configurazione venga inviata ripetutamente e quindi alleggerire il carico alla Raspberry Pi 3; questo dettaglio, sebbene poco rilevante nell'applicazione qui descritta, diventa importante quando The Tactigon deve cimentarsi in applicazioni decisamente più impegnative dal punto di vista del calcolo.

Una volta superato questo controllo, viene valutata la disuguaglianza:

```
accData.x > threshold.
```

In essa il parametro *threshold* è una soglia di accelerazione, espressa in m/s^2 , che ci permette di stabilire la sensibilità di The Tactigon alla "gesture" che andiamo ad eseguire; in altre parole, la soglia consente di rilevare come gesture il movimento sopra una certa velocità ed ignorare i movimenti lenti delle mani, lasciandoci liberi di muoverle senza per questo impartire comandi errati.

Invece *accData.x* è la componente dell'accelerazione registrata da The Tactigon sull'asse X.

Abbiamo la possibilità di sfruttare l'IMU a 9 libertà di movimento per catturare le gesture di cui abbiamo bisogno. Utilizzando la libreria fornita nel pacchetto di installazione di The Tactigon è facilmente possibile ottenere le informazioni su inclinazioni ed accelerazioni, per poter così pro-

grammare la scheda e personalizzarla al massimo. Il modulo Bluetooth Low Energy integrato penserà alla comunicazione di queste gesture sotto forma di comandi come abbiamo pensato nel nostro firmware.

RASPBERRY PI 3 - PYTHON 2.7.13

La Raspberry Pi esegue uno script python che:

- esegue l'inizializzazione del player VLC, il caricamento della playlist;
- connessione alla scheda The Tactigon, cercando il mac address e gli UUID del servizio e della caratteristica;
- imposta la notification sulla caratteristica Bluetooth Low Energy, per poter utilizzare una callback al momento della ricezione di nuovi dati;
- la callback, quando attivata, esegue il parse dei comandi ricevuti tramite Bluetooth Low Energy e, grazie alla libreria di binding *vlc.py*, controlla le azioni del player VLC.

Queste due righe di codice permettono di initialize il nostro player VLC:

```
mediaListPath = "file:///home/pi/vlc-tactigon/media/playlist.xspf"
instance = vlc.Instance(["--sub-source=marq"] + [mediaListPath])
```

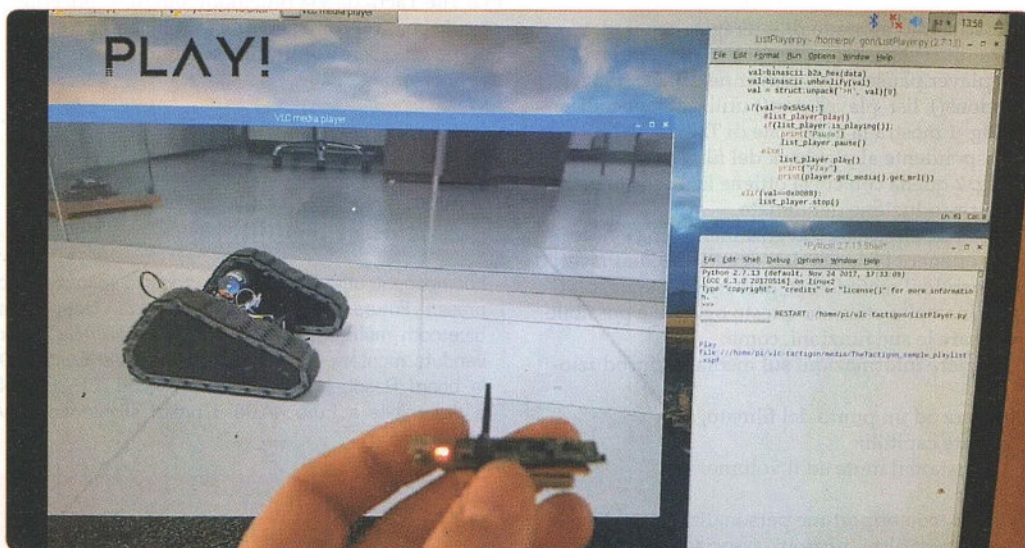


Fig. 1 - Gesture corrispondente al PLAY..

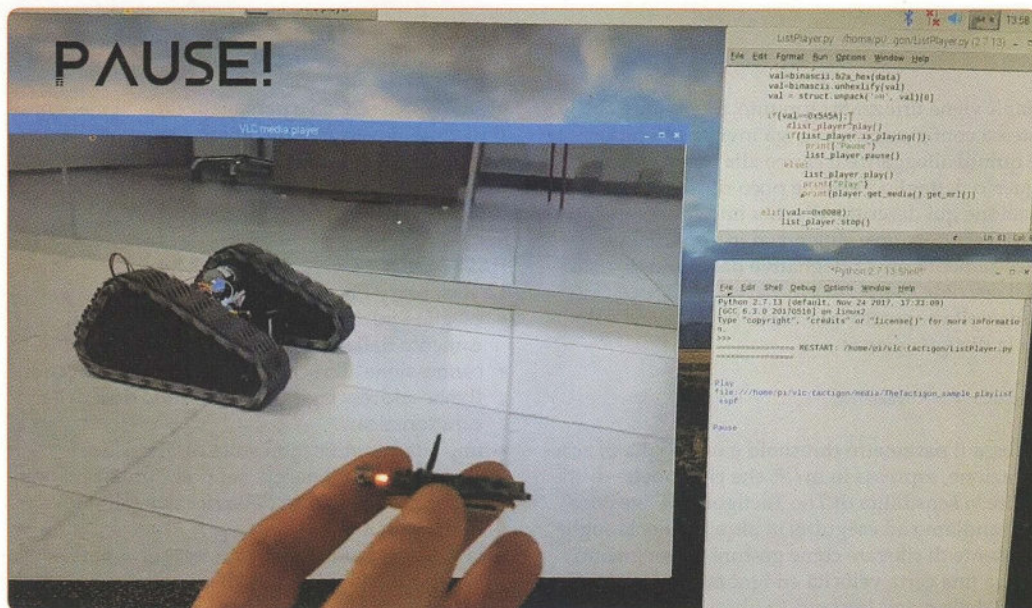


Fig. 2 - Gesture che mette in pausa il filmato.

Per poterlo utilizzare, però, abbiamo bisogno di ottenere una nuova istanza di `media_list_player`, ed associare una playlist, ovvero la nostra `Media_list`:

```
list_player = vlc.Instance.media_list_player_
new(instance)
list_player.set_media_list(Media_list)
```

Adesso siamo in grado di interagire con il player tramite i semplici comandi `list_player.play()`, `list_player.pause()`, `list_player.next()`, `list_player.previous()`, `list_player.stop()` utilizzati nel codice. La Fig. 1 mostra la posizione di The Tactigon corrispondente al playback del filmato in VLC e la Fig. 2 quella con cui si ottiene la messa in pausa della riproduzione del filmato.

È possibile effettuare l'avanzamento, ma è necessario ottenere l'istanza dell'oggetto `MediaPlayer`, chiamando il metodo `list_player.get_media_player()`. Una volta ottenuto questo oggetto è possibile utilizzare le sue funzioni, come:

- ottenere informazioni sul media in riproduzione;
- saltare ad un punto del filmato;
- saltare capitoli;
- impostare il mute ed il volume.

Volendo, con opportune personalizzazioni è possibile agire su altre funzioni, associandole a gesture complesse, oppure semplicemente più rapide di

quelle previste (giocando sulla soglia di sensibilità di The Tactigon).

CONCLUSIONI

Bene, ora che abbiamo ottenuto il controllo della nostra playlist tramite gesture, possiamo pensare ad ulteriori sviluppi dell'interazione tra Raspberry Pi e The Tactigon via Bluetooth; infatti il concetto su cui si basa il controllo di VLC non è detto che non possa essere applicato e adattato a tante altre applicazioni.



per il MATERIALE

I componenti utilizzati in questo progetto sono disponibili presso Futura Elettronica. La scheda `Tactigonbasic` bluetooth multisensore (cod. `TACTIGONBASIC`) viene venduta montata e collaudata al prezzo di Euro 89,00, la board Raspberry Pi 3 Tipo B+ (cod. `RPI3BPLUS`) è disponibile a Euro 44,00. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 • Fax: 0331-792287 - www.futurashop.it

LA TENSIONE DI RETE OVUNQUE TU SIA!

Nuovi inverter economici e potenti

Inverter da 12 Vdc a 220 Vac con forma d'onda d'uscita sinusoidale modificata dotati di protezione da sovraccarico, sovratensione d'ingresso, cortocircuito e allarme batteria scarica. Dispongono di uscita USB supplementare (5 V - 500 mA), presa di tipo Schuko e cavi per il collegamento alla batteria.

Ideali su camper, camion, roulotte, per alimentare TV, VCR, PC, impianti HI-FI e molto altro!

€ 39,⁰⁰

Cod. FR670M300



600 W



€ 54,⁰⁰

Cod. FR670M600

€ 148,⁰⁰

Cod. FR670M1500



Prezzi IVA inclusa.

...e per le tue batterie scegli FIAMM!

Batteria ricaricabile ermetica
12 V - 1,2 Ah



€ 18,⁰⁰

Cod. NP12-12

Batteria ricaricabile ermetica 12 V - 2 Ah



€ 19,⁵⁰

Cod. NP21-12

Batteria ricaricabile ermetica
12 V - 7,2 Ah



€ 32,⁰⁰

Cod. NP72-12

**FUTURA
ELETTRONICA®**

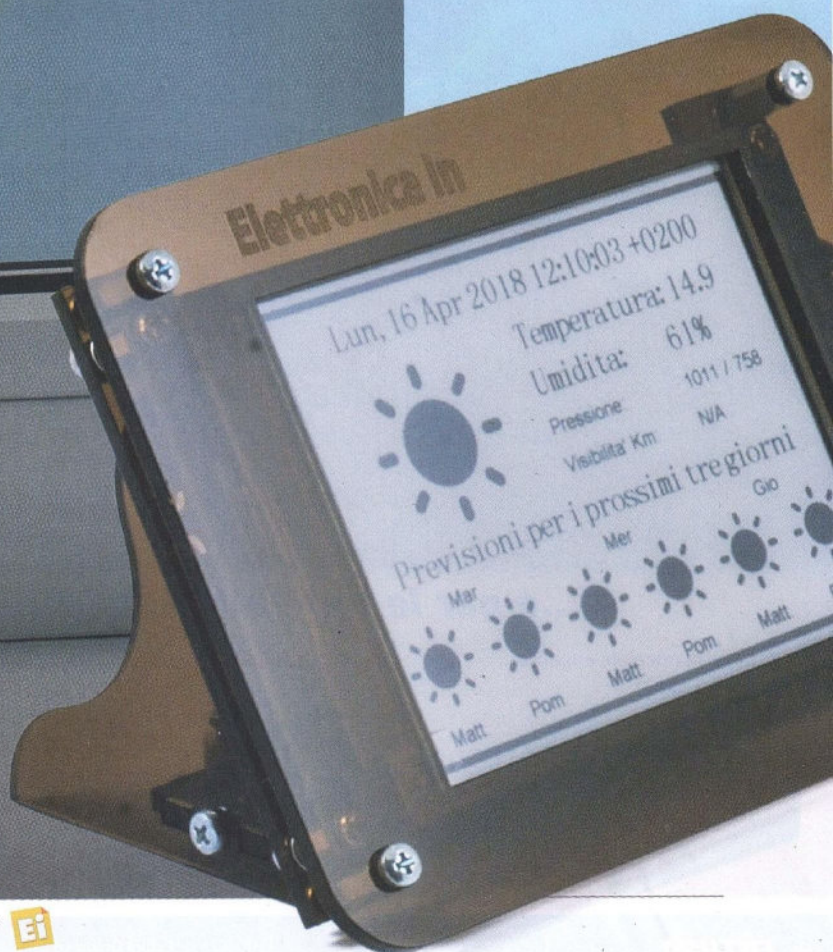
www.futurashop.it

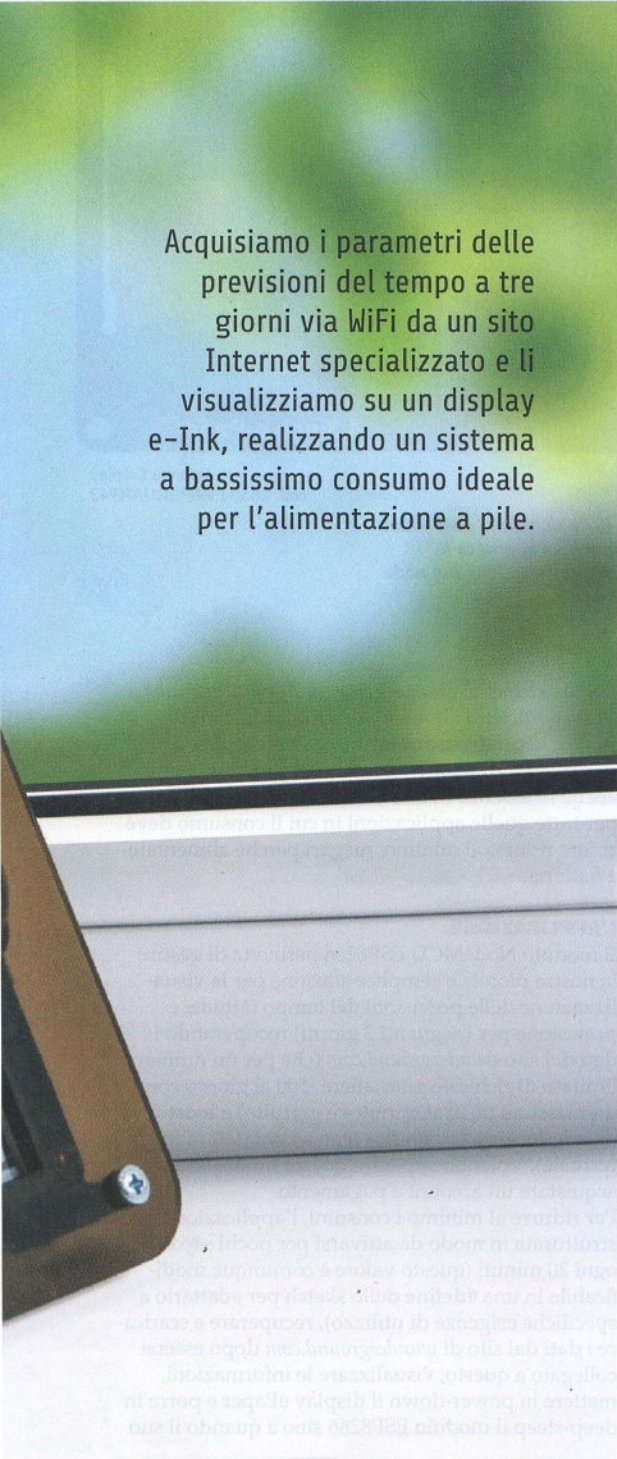
Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche di questi prodotti
e acquisti on-line su www.futurashop.it

STAZIONE METEO CON ePAPER

di GUGLIELMO BRAGUGLIA





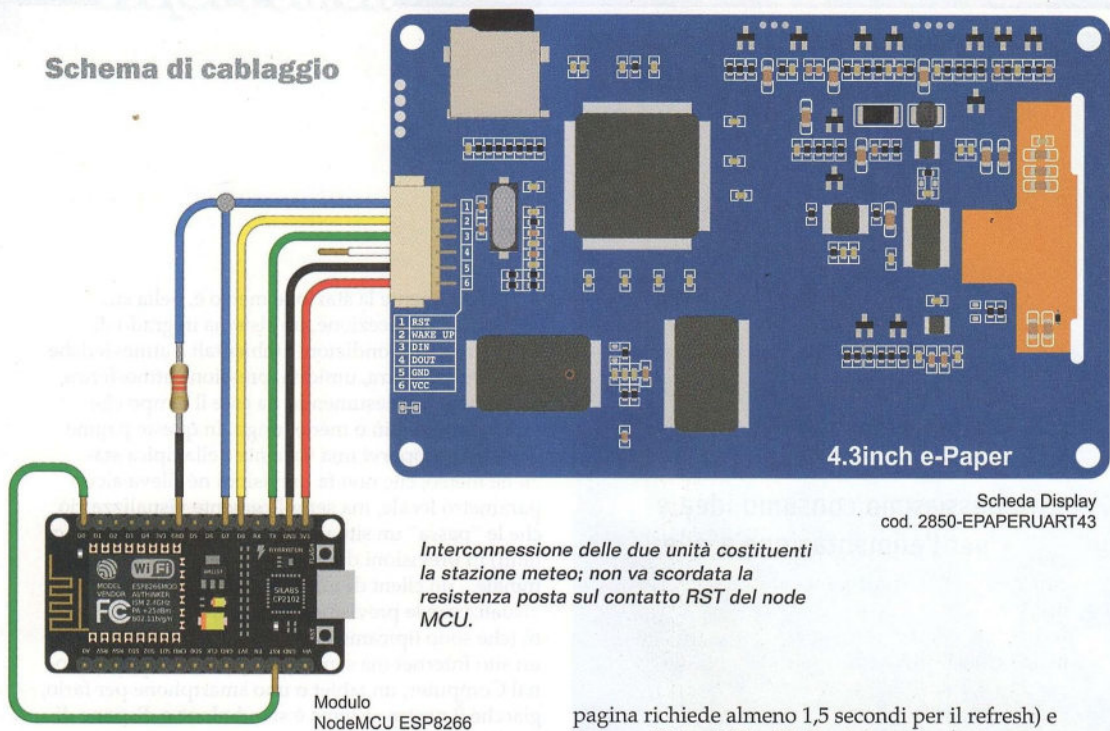
Acquisiamo i parametri delle previsioni del tempo a tre giorni via WiFi da un sito Internet specializzato e li visualizziamo su un display e-Ink, realizzando un sistema a bassissimo consumo ideale per l'alimentazione a pile.

Generalmente la stazione meteo è, nella sua canonica accezione, un sistema in grado di acquisire delle condizioni ambientali e atmosferiche come temperatura, umidità, pressione atmosferica, precipitazioni, desumendo da esse il tempo che farà a termine più o meno lungo. In queste pagine vogliamo proporvi una variante della tipica stazione meteo, che non fa previsioni né rileva alcun parametro locale, ma semplicemente visualizza ciò che le "passa" un sito web specializzato (ce ne sono tanti) in previsioni del tempo; in pratica è un terminale... un client di un server web che permette di visualizzare le previsioni, nel nostro caso a tre giorni (che sono tipicamente le più affidabili), fornite da un sito Internet ma senza dover utilizzare un Personal Computer, un tablet o uno smartphone per farlo, giacché il nostro sistema è stand-alone e dispone di un display da 4,3" di diagonale, in formato 4:3. Il tutto è molto compatto e prende posto in un frame a cornice che farà bella mostra di sé su qualsiasi mobile di casa.

COME FUNZIONA

Per svolgere il proprio compito, la nostra speciale centralina meteo si avvale di una board detta NodeMCU, ossia un sistema a microcontrollore dotato di connettività wireless, nato e predisposto per funzionare da nodo di una rete wireless, che nel nostro caso è la comunissima WiFi; allo scopo integra un modulo basato sul popolare integrato ESP8266 (cuore di una gran quantità di moduli WiFi low-cost per il mondo Arduino e non solo), che si fa carico dello stack TCP/IP e si occupa della connessione WiFi. Tale modulo è interfacciato al microcontrollore della board NodeMCU che governa la stazione meteo e lo abbiamo programmato affinché provveda a realizzare la funzione richiesta; nello specifico, il micro del NodeMCU si occupa dell'interfacciamento all'ESP8266 (ma non per la gestione della connessione WiFi, che è demandata ad esso), del puntamento al sito di riferimento e all'ottenimento dei dati da esso. Inoltre provvede alla gestione della visualizzazione sul display ePaper. Per ottenere informazioni meteo attinenti, il No-

Schema di cablaggio



deMCU legge dal firmware la località di cui si richiedono le previsioni (poi spiegheremo come si specifica la località...), quindi la comunica al sito di riferimento (abbiamo scelto www.wunderground.com) per avere il meteo della corrispondente area geografica.

Il display ePaper che abbiamo utilizzato nel progetto è un 4,3" di diagonale prodotto dalla Waveshare e commercializzato dalla Futura Elettronica con il codice 2850-EPAPERUART43. L'ampia dimensione dello schermo e la risoluzione che ne consegue (800x600 punti/pixel), la possibilità di scegliere tra quattro livelli di grigio, l'interfaccia seriale (di cui è strettamente necessario solo il pin RX se si ignorano le risposte ai comandi che il display invia sul pin TX), l'alloggiamento per schede mini SD per la memorizzazione di immagini e la buona velocità di aggiornamento della schermata (il refresh completo della pagina richiede solo 1,5 secondi) pongono questo visualizzatore al primo posto tra quelli prodotti da Waveshare e lo rendono adatto a tutti quei progetti dove sono richiesti ampio spazio grafico, facilità d'uso, bassissimo consumo e dove non sia necessario un continuo aggiornamento dei dati visualizzati (ricordiamo che ogni modifica della

pagina richiede almeno 1,5 secondi per il refresh) e siano sufficienti i livelli di grigio supportati. Inoltre la caratteristica intrinseca dei display di tipo ePaper di mantenere memorizzata l'immagine anche in assenza di alimentazione, lo rende ideale per tutte quelle applicazioni in cui il consumo deve essere ridotto al minimo, magari perché alimentate a batteria.

L'APPLICAZIONE

Il modulo NodeMCU ESP8266 permette di gestire la nostra piccola e semplice stazione per la visualizzazione delle previsioni del tempo (attuale e previsione per i seguenti 3 giorni) recuperando i dati dal sito wunderground.com (che per un numero limitato di richieste giornaliere -500 al giorno con un massimo di 10 al minuto- è gratuito) e mostrandole in forma iconografica (con alcuni dati in forma testuale). Volendo superare questo limite, occorre acquistare un account a pagamento.

Per ridurre al minimo i consumi, l'applicazione è strutturata in modo da attivarsi per pochi secondi ogni 20 minuti (questo valore è comunque modificabile in una `#define` dello sketch per adattarlo a specifiche esigenze di utilizzo), recuperare e scaricare i dati dal sito di wunderground.com dopo essersi collegato a questo, visualizzare le informazioni, mettere in power-down il display ePaper e porre in deep-sleep il modulo ESP8266 sino a quando il suo

timer interno non avrà contato i suddetti 20 minuti (pari a 1.200 secondi o, come richiesto per l'ESP8266, 1.200.000.000 microsecondi); poi viene generato un segnale su GPIO16/D0 che, collegato con un ponticello al pin RST, provocherà il reset della scheda, facendo ripartire il ciclo dall'inizio.

Questo sistema consente di ottenere il completo spegnimento della scheda senza il bisogno di far girare una routine software nel microcontrollore; molto semplicemente, il firmware non prevede un loop ma viene eseguito in sequenza dall'inizio (conseguente a ciascun reset) fino alla messa in deep-sleep del NodeMCU, cui segue l'impulso su GPIO16/D0 che esegue il reset e fa ripartire l'esecuzione del firmware da capo (vedere il Listato 1).

IL RECUPERO DELLE PREVISIONI

Torniamo adesso sul modo in cui la nostra stazione meteo preleva i dati sulle previsioni meteo: per recuperare le previsioni del tempo da visualizzare sul display utilizzeremo il sito *wunderground.com*. Per recuperare i dati occorre una API Key da inserire nelle richieste inviate automaticamente al sito dai nostri sistema; la API key viene assegnata ad ogni utente registrato, quindi dovete:

1. registrarvi (gratuitamente) sul sito di Wunderground: *www.wunderground.com*;
2. dal menu in alto alla home page, selezionare "More" che aprirà un menu a tendina; qui, selezionare "Weather API for Developers";
3. generare una propria APIkey di tipo "STRATUS PLAN", FREE, ovvero gratuita.

Si otterrà un qualche cosa che somiglia a "bf11dccc6988fc42" che è la APIkey che vi occorre. Le istruzioni per l'utilizzo e l'elenco delle "parole chiavi" utilizzabili nelle richieste (che si ricevono nelle risposte dal sito), sono recuperabili dalla seguente pagina web: *www.wunderground.com/weather/api/d/docs?MR=1* (è molto utile il menu a sinistra della pagina).

Ottenuta la API Key la utilizzeremo nel nostro programma per creare le richieste HTTP indirizzate al sito *wunderground.com*; ciascuna richiesta, per lo scopo che ci siamo prefissati assumerà la forma seguente: *http://api.wunderground.com/api/_api_key_assegnata/_conditions/forecast/q/nazione/città.json*.

In essa, al posto di nazione va scritto lo Stato (con l'iniziale maiuscola, quindi il nostro Paese è Italy) e in luogo di città, appunto la città della quale si desiderano le previsioni meteo (ad esempio Milano). A ogni richiesta, il sito ci risponderà con una paginata in formato json contenente tutte le informa-



Un nuovo e-Ink A5 per Sony

Il colosso giapponese rinnova la propria offerta di tablet Digital Paper con il nuovissimo DPT-CP1 equipaggiato con un display E-ink Mobius in formato A5 con cui è possibile interagire tramite un pennino, che permette di leggere, creare e modificare contenuti come se si lavorasse su un block-notes. Il nuovo display è da 10,3" e presenta l'invidiabile risoluzione di 1.404x1.872 pixel (con una densità di pixel di 224 ppi). Particolarmente interessante l'autonomia, resa chiaramente possibile dal consumo quasi nullo del pur ampio display E-ink: sino a 3 settimane di utilizzo con una sola ricarica (leggendo 30 pagine al giorno per 1 ora con WiFi e Bluetooth spenti). La memoria disponibile per i documenti è poco più di 10 GB.

Secondo l'annuncio fatto dalla Sony, il nuovo tablet eInk sarà in vendita in Italia da giugno.

zioni disponibili; siccome la pagina json può essere molto lunga (anche oltre i 10 kbyte), per risparmiare SRAM il nostro firmware la analizzerà in "tempo reale" durante la ricezione dei dati. Per fare ciò

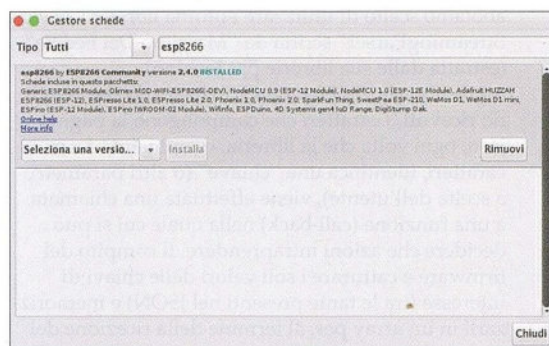


Fig. 1 - Configurazione del NodeMCU.

Listato 1

```
void setup() {
  char  inChar;
  //
  delay(1000);
  //
  Serial.begin(115200);
  //
  pinMode(LED_PIN, OUTPUT);
  digitalWrite(LED_PIN, LOW);
  myEPD_Init();
  //
  strcat(httpGet, MY_APIKEY);
  strcat(httpGet, "/conditions/forecast/q/");
  strcat(httpGet, MY_LOC);
  strcat(httpGet, ".json");
  //
  #ifdef __DEBUG
  Serial.println();
  Serial.println();
  Serial.print("HTTP GET: ");
  Serial.println(httpGet);
  Serial.println("Attende 5 secondi per stabilizzare ESP8266");
  #endif
  delay(5000);
  // Si connette all'access point
  WiFiMulti.addAP(MY_SSID, MY_PASS);
  //
  #ifdef __DEBUG
  Serial.print("In attesa della connessione ...");
  #endif
  while (WiFiMulti.run() != WL_CONNECTED) {
    #ifdef __DEBUG
    Serial.print(".");
    #endif
    delay(500);
  }
  #ifdef __DEBUG
  Serial.println(" connesso.");
  #endif
  // Tutto pronto, inizializza ora il parser JSON
  parser.setCallback(jsonCallback, NULL);
  // Effettua la richiesta HTTP al server ...
  HTTPClient http;
  http.begin(httpGet);
  int httpCode = http.GET();
  //
  if (httpCode > 0) {
    // HTTP GET riuscita ...
    if (httpCode == HTTP_CODE_OK) {
      // get lenght of document (is -1 when Server sends no Content-Length header)
      #ifdef __DEBUG
```

abbiamo scelto di utilizzare l'ottima libreria "JSON-StreamingParser" scritta da "Massimo Del Fedele" (estratta dalle sue librerie per Fishino).

A tale libreria vanno passati, man-mano che vengono ricevuti, i caratteri che compongono la pagina json; ogni volta che la libreria, esaminando i vari caratteri, identifica una "chiave" (o altri parametri a scelta dell'utente), viene effettuata una chiamata a una funzione (call-back) nella quale cui si può decidere che azioni intraprendere. Il compito del firmware è catturare i soli valori delle chiavi di interesse (tra le tante presenti nel JSON) e memorizzarli in un array per, al termine della ricezione del json, averli a disposizione e creare la videata con le previsioni del tempo aggiornate.

Come mostrato nel Listato 2, il nostro sketch estrae dal json i dati relativi all'orario locale ("local_time_rfc822") temperatura e umidità ("temp_c" e "relative_humidity") pressione atmosferica ("pressure_mb") visibilità ("visibility_km") icone del tempo ("icon", ossia sole o nuvole) periodo di validità delle previsioni ("period" con valore 3) e relative icone.

Anche se la risposta json entra molto nel dettaglio, abbiamo semplificato la sua interpretazione, raggruppando una serie di risposte e creando una serie di icone (di dimensioni diverse per la situazione attuale, che è 200x200) e per le previsioni dei giorni successivi (100x100) in B/N (formato bitmap di Windows). Tale icone debbono essere copiate nella


```

Serial.println("Richiesta al server fatta con successo ...");
#endif
int lung = http.getSize();
// Get the tcp stream
WiFiClient * stream = http.getStreamPtr();
// read all data from server
#ifdef __DEBUG
Serial.println("... stampa dei dati ricevuti ...");
#endif
//
fStartJSON = false;
JSONelement = 0;
idxArray = 0;
parser.reset();
clearArray();
//
while ( http.connected() && (lung > 0) ) {
  size_t stSize = stream->available();
  if (stSize) {
    stream->readBytes(&inChar, 1);
    if (inChar == '{') fStartJSON = true;
    #ifdef __DEBUG
    Serial.print(inChar);
    #endif
    if (fStartJSON) parser.feed(inChar);
    lung--;
  }
}
#ifdef __DEBUG
Serial.println();
#endif
showResult();
}
else {
  // HTTP GET fallita ...
  #ifdef __DEBUG
  Serial.println("Errore HTTP GET, richiesta al server fallita.");
  #endif
}
http.end();
// Da il tempo al display ePaper di terminare tutte le operazioni
delay(10000);
epd_enter_stopmode();
delay(2500);
digitalWrite(LED_PIN, HIGH);
// Entra in "deep-sleep" per un periodo di POSTINGINTERVAL microsecondi (usec) ...
ESP.deepSleep(POSTINGINTERVAL);
// ... ed, al risveglio, effettua un RESET della scheda ripartendo da capo con il setup().
}

```

directory radice di una scheda mini SD che dovrà essere inserita nell'apposito lettore presente sulla scheda controller del display ePaper; così saranno richiamabili direttamente dal programma applicativo attraverso il loro nome.

Ma attenzione: la lunghezza del nome delle icone (e dei file leggibili dal display dalla SD) è limitata al formato 6,3 e il nome deve essere tutto in maiuscolo.

CONFIGURAZIONE DEL NODEMCU

Per programmare questa scheda sarà necessario aprire l'IDE Arduino e scrivere http://arduino.esp8266.com/stable/package_esp8266com_index.json nella casella di testo inferiore della finestra "URL aggiuntive per il gestore di schede", quindi fare

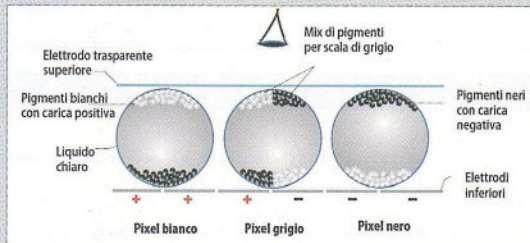
clic su OK per continuare. Dopo aver fatto questo primo passaggio possiamo installare la scheda, cliccando dalla schermata principale dell'IDE "Strumenti>Scheda>Gestore schede..." si aprirà la finestra "Gestore schede" (Fig. 1).

A questo punto, nella barra "Filtrate la ricerca" scrivete "esp8266" e se avete eseguito i passaggi precedenti correttamente, troverete l'opzione "Installa" disponibile (raccomandiamo di installare l'ultima versione disponibile).

Dopo aver installato la scheda (lo spazio richiesto ammonta a circa 150 MB) potrete passare alla configurazione: nella schermata principale dell'IDE cliccate *Strumenti>Scheda>NodeMCU 1.0* (ESP-12E Module) e poi cliccate nuovamente su

Il display e-Ink

Nato con gli e-book reader, è considerato un visualizzatore "a inchiostro elettronico" perché riflette la luce come un comune foglio di carta bianca o la assorbe come l'inchiostro nero. Il display è composto da due lastre (una delle quali è trasparente), chiuse lateralmente, e tra le quali si trova una sostanza liquida contenente piccolissime sfere caricate elettricamente; in ciascuna delle microsfeere, metà è positiva e colorata di nero, mentre l'altra metà è caricata negativamente ed è di colore bianco. Applicando un campo elettrico a speciali elettrodi sulla superficie delle lastre, si possono orientare le sfere in modo che appaiano bianche o nere; in altre parole, applicando la polarità positiva sulla lastra esterna (quella da cui guarda l'osservatore) le sfere si orientano con la metà negativa da quella parte e, siccome le semisfere negative sono bianche, nei punti polarizzati positivamente il display è bianco. Se invece si applica la polarità positiva sulla lastra interna e negativa su quella esterna, dall'esterno appare la metà nera (positiva). La tecnologia E-Ink permette di realizzare supporti sottili e flessibili, dato che la struttura a sfere non viene alterata dalla torsione o flessione. Ma ciò che rende davvero unica la tecnologia E-Ink sono due caratteristiche: la prima è che tramite l'utilizzo delle sfere consente di rappresentare perfettamente un foglio stampato, dato che l'immagine (foto o testo) appare formata da punti, esattamente come avviene con una stampa a getto d'inchiostro, a laser o in tipografia; nei display E-Ink, analogamente a quanto si fa nelle macchine tipografiche, una zona è tanto più scura quanto più fitti sono i punti neri e viceversa. Non meno importante è il fatto che l'E-Ink è l'unico display praticamente zero-power: essendo basato su microsfeere che una volta orientate restano ferme, richiede elettricità solo quando bisogna polarizzare le lastre per orientare le sfere stesse e quindi soltanto per cambiare il contenuto dello schermo. Il tipico contrasto di un display E-Ink è di 7:1, mentre l'angolo di visione è di 180 gradi; il tempo di risposta (ossia di rotazione della sfera dalla



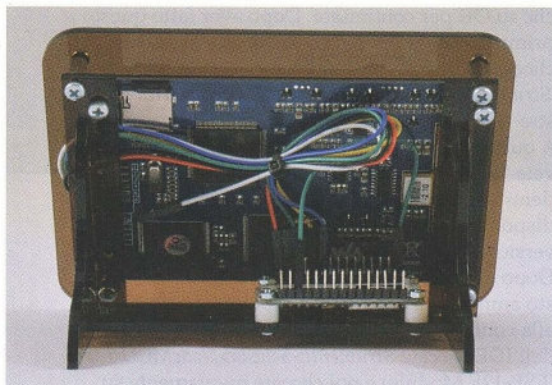
Schematizzazione del funzionamento dell'e-ink.

parte nera a quella bianca) è intorno ai 700 ms (nella modalità a scala di grigio) e circa 250 ms in bianco e nero. Dato che nascono per funzionare senza retroilluminazione i display E-Ink non sono destinati a TV e monitor, ma a leggere libri, giornali ed altri documenti alla luce del giorno; anzi, proprio l'assenza di luce propria li assimila alla carta stampata e consente di guardarli per ore senza affaticare la vista.

La tecnologia E-Ink è stata la base della realizzazione da parte di LG-Philips di display a colori. Il metodo di eccitazione rimane invariato rispetto a quello dei display monocromatici, rispetto ai quali però, quando un punto deve restare spento la sfera viene polarizzata in modo da girarsi mostrando il lato nero. Il fondo del display è una lamina di metallo che permette di far riacquistare all'insieme la forma originale anche se viene piegato più volte. Le sfere sono gestite da un sistema tipo il TFT degli LCD e ogni punto è diviso in tre sub-pixel, ognuno dei quali contiene una microsfera ed ha davanti un filtro colorato (rosso, verde e blu); la visualizzazione avviene come nella stampa a colori, però si usano solamente i colori della proiezione, anche se si lavora a riflessione di luce: quando una sfera è ruotata dalla metà nera, il colore non partecipa, mentre quando la stessa è dalla metà bianca, la luce si riflette su di essa e torna all'esterno. Ogni pixel si presenta di un colore formato dalla somma della luce riflessa dalle sfere girate dal lato bianco.

Strumenti>Upload Speed>115200.

Una volta fatto ciò, avrete terminato la configurazione e l'installazione della scheda e sarete pronti a programmare il NodeMCU!



LA COSTRUZIONE

Bene, spiegato a sommi capi come funziona la nostra stazione meteo, vediamo come realizzarla; la prima cosa è provvedere all'interconnessione tra il NodeMCU e il pannello ePaper, il cui cablaggio è descritto dall'apposito schema di collegamento visibile nella pagina accanto. Potete consultare la Tabella 1, che riporta i collegamenti tra il display e il NodeMCU; in particolare, sul modulo ESP8266 bisogna unire il GPIO16/D0 con il contatto RST, al fine di assicurare il reset automatico. Inoltre è necessario collegare una resistenza da 2,2 kΩ 1/4W tra il contatto di reset del display ePaper e il GND, così da evitare che, quando il NodeMCU viene portato in deep-sleep e i pin di I/O vanno in condizione

Il prototipo della stazione meteo visto da dietro, con in evidenza il cablaggio tra le due unità, avvitate al telaio.

three-state, il pin (divenuto fluttuante) possa causare un livello logico alto e forzare il reset del display, risvegliandolo prima del tempo.

Per alimentare il tutto si può utilizzare o un alimentatore con uscita 5V su micro USB da collegare direttamente al NodeMCU o una scheda per la gestione e ricarica di un pannello solare con una batteria LiPo 3,7V/2.800 mAh, alimentata da un alimentatore AC/DC da 5V connessa via microUSB al NodeMCU.

L'insieme dell'elettronica va inserito in un contenitore di dimensioni adeguate, che frontalmente abbia una cornice che consente di vedere il display ePaper e che sul retro riporti il connettore di alimentazione ed abbia un foro per far passare un cavetto USB/microUSB con cui collegare il NodeMCU al computer per caricare il firmware.

Il nostro prototipo è stato assemblato su una cornice in plexiglass brunito composta da un telaio a V con piedistallo e da una cornice frontale che circonda l'ePaper, uniti tramite colonnine distanziali. Sulla parte anteriore del telaio abbiamo fissato il display, mentre all'interno del piedistallo, tramite distanziali cilindrici in plastica abbiamo montato la scheda NodeMCU, rendendone accessibile il connettore microUSB. Le foto del prototipo che trovate in queste pagine sono un utile spunto per la realizzazione.

IL FIRMWARE

Il programma per il nostro sistema è stato sviluppato in ambiente Arduino IDE v1.8.5 con l'utilizzo del "core" ESP8266 v2.4.0 ed è ampiamente commentato. Caratteristica peculiare è che, contrariamente alla norma, il programma risiede interamente nella funzione `setup()` (oltre ad una serie di funzioni accessorie da essa richiamate), mentre la funzione `loop()` è completamente vuota.

Questo perché, come già descritto, il programma si avvia, effettua le richieste, visualizza i dati e si pone in "deep-sleep", condizione dalla quale esce con un reset che riavvia il processo da capo. In tali condizioni è evidente che non c'è nulla di ripetitivo e continuo da inserire nel `loop()`, ma solo una serie di istruzioni che, nel `setup()`, vengono ogni volta eseguite sempre come fosse la prima volta.

Oltre al "core" per ESP8266, occorre installare la libreria "JSONStreamingParser" (fa parte delle librerie standard per Fishino) e la libreria "epd" che si trova nell'archivio "4.3inch-e-Paper-Code.7z" scaricabile dal sito di Waveshare alla pagina <https://www.waveshare.com/wiki/File:4.3inch-e-Paper-Code.7z>; la libreria, una volta scompattato il file scaricato dal sito della Waveshare, va installata nella vostra

Tabella 1 - I collegamenti da realizzare.

Display ePaper	NodeMCU
GND (cavo nero)	GND
Vcc (cavo rosso)	+3,3V
RX (cavo verde)	GPIO1 (TXD0)
TX (cavo bianco)	N.C.
RESET (cavo blu)	GPIO13 (D7 - modificabile da #define)
WakeUp (cavo giallo)	GPIO15 (D8 - modificabile da #define)

cartella delle librerie, tipicamente dentro *sketchbook\libraries*.

Una volta configurato l'IDE è necessario aprire il programma e personalizzarlo modificando alcune #define che si trovano ad inizio programma subito dopo le #include, quindi salvarlo:

```
#define MY_SSID "ssid_della_rete_wifi"
#define MY_PASS "password_della_rete_wifi"
#define MY_APIKEY "api_key_assegnata"
#define MY_LOC "località_di_interesse"
```

Le prime due riguardano l'SSID e la password della rete WiFi cui la stazione si conatterà per raggiungere Internet e da essa il sito *www.wunderground.com*; le desumete dalla rete wireless, ovvero dalla pagina di impostazione del router WiFi. La terza riguarda la APIKEY assegnata dal sito, secondo quanto già spiegato.

Relativamente alla #define MY_LOC "località di interesse" occorre rispettare la sintassi ricavabile dal sito *www.wunderground.com*, ovvero la località va espressa con il formato *nome_nazione/nome_città*. Nel caso ci interessino le previsioni della zona di Milano dovremo mettere tra gli apici della #define il seguente testo: "Italy/Milano" così da avere: #define MY_LOC "Italy/Milano"

Se volete simulare da browser la richiesta e vedere il codice json che viene ritornato (così da verificare anche il corretto formato della località), basta usare il seguente URL: http://api.wunderground.com/api/_api_key_assegnata/conditions/forecast/q/Italy/Milano.json, mettendo al posto di *_api_key_assegnata* la API KEY precedentemente ottenuta.

Una caratteristica del codice che riteniamo interessante far notare è che le varie indicazioni di "icona" ritornate nel json possono essere:

```
"chanceflurries", "chancerain", "chancesleet",
"chancesnow", "chancetstorms", "clear", "cloudy",
"flurries", "fog", "hazy", "mostlycloudy", "mostlysunny", "partlycloudy", "partlysunny", "sleet",
"rain", "snow", "sunny", "tstorms", "unknown"
```


Listato 2

```
void jsonCallback(uint8_t filter, uint8_t level, const char *name, const char *value, void *cbObj) {
    char sIdx[4];
    //
    switch (JSONElement) {
        case 0:
            // JSONElement = 0, attende di trovare "local_time_rfc822"
            if (strcmp(name, "local_time_rfc822") == 0) {
                JSONElement++;
                strncpy(conditionArray[idxArray], value, MAXSIZE);
                conditionArray[idxArray++][MAXSIZE] = 0x00;
            }
            break;
        case 1:
            // JSONElement = 1, attende di trovare "temp_c"
            if (strcmp(name, "temp_c") == 0) {
                JSONElement++;
                strncpy(conditionArray[idxArray], value, MAXSIZE);
                conditionArray[idxArray++][MAXSIZE] = 0x00;
            }
            break;
        case 2:
            // JSONElement = 2, attende di trovare "relative_humidity"
            if (strcmp(name, "relative_humidity") == 0) {
                JSONElement++;
                strncpy(conditionArray[idxArray], value, MAXSIZE);
                conditionArray[idxArray++][MAXSIZE] = 0x00;
            }
            break;
        case 3:
            // JSONElement = 3, attende di trovare "pressure_mb"
            if (strcmp(name, "pressure_mb") == 0) {
                JSONElement++;
                strncpy(conditionArray[idxArray], value, MAXSIZE);
                conditionArray[idxArray++][MAXSIZE] = 0x00;
            }
            break;
        case 4:
            // JSONElement = 4, attende di trovare "visibility_km"
            if (strcmp(name, "visibility_km") == 0) {
                JSONElement++;
                strncpy(conditionArray[idxArray], value, MAXSIZE);
                conditionArray[idxArray++][MAXSIZE] = 0x00;
            }
            break;
        case 5:
            // JSONElement = 5, attende di trovare "icon"
            if (strcmp(name, "icon") == 0) {
                JSONElement++;
                strncpy(conditionArray[idxArray], value, MAXSIZE);
                conditionArray[idxArray++][MAXSIZE] = 0x00;
            }
            break;
        case 6:
        case 8:
        case 10:
        case 12:
        case 14:
        case 16:
            // JSONElement = (6, 8, 10, 12, 14, 16) attende di trovare "period" con valore (2, 3, 4, 5, 6, 7)
            // ovvero di trovare "period" con la stringa equivalente a ((JSONElement / 2) - 1)
            itoa((JSONElement / 2) - 1, sIdx, 10);
            if ( (strcmp(name, "period") == 0) && (strcmp(value, sIdx) == 0) ) JSONElement++;
            break;
        case 7:
        case 9:
        case 11:
        case 13:
        case 15:
        case 17:
            // JSONElement = (7, 9, 11, 13, 15, 17) attende di trovare "icon"
            if (strcmp(name, "icon") == 0) {
                JSONElement++;
                strncpy(conditionArray[idxArray], value, MAXSIZE);
                conditionArray[idxArray++][MAXSIZE] = 0x00;
            }
            break;
        default:
            break;
    }
}
```


Il prototipo con collegato il cavo USB al NodeMCU, che permette anche l'alimentazione della stazione.

e vengono associate ad un numero più ridotto di immagini presenti sulla SD e rappresentate da un numero identificativo che può essere:

"04", "09", "04", "04", "13", "11", "06", "04", "15", "15", "07", "07", "07", "07", "04", "10", "03", "11", "13", "16"

Come si vede, alcune indicazioni di "icona" ricevute nel json vengono associate alla stessa icona grafica che viene presentata sul display. Esempio: "mostlycloudy", "mostlysunny", "partlycloudy", "partlysunny" vengono tutti associati all'icona con identificativo "07".

Altra cosa da notare è che i nomi sia dei mesi che dei giorni vengono ricevuti nel json in inglese e tradotti dal programma in italiano, tramite l'utilizzo di due array dedicati. I messaggi testuali che vengono quindi inviati al display ePaper sono memorizzati in stringhe classiche del "C" dichiarate come costanti (per ovvie ragioni di ottimizzazione). Altro dettaglio interessante è che alcuni metodi della libreria "esp" non sono stati utilizzati (perché richiedono l'utilizzo dei pin prefissati in libreria) ma abbiamo preferito loro delle versioni personalizzate; per l'esattezza, nel firmware sono state create le seguenti funzioni che li rimpiazzano:

- void myEPD_Init(void);
- void myEPD_Reset(void);
- void myEPD_WakeUp(void);

Inoltre abbiamo creato alcune funzioni per "pulire" dai caratteri non necessari alcuni valori che si ricevono dal file json, eliminando, ad esempio, i doppi apici (") o i prefissi ("_nt") al fine di snellire e semplificare il trattamento dei valori stessi. Una volta compilato il programma e caricato sul NodeMCU, si può connettere quest'ultima scheda all'alimentazione che si è deciso di utilizzare; una volta alimentato l'insieme, si vedrà il LED blu presente sul NodeMCU accendersi e in concomitanza con questo evento assisteremo all'accensione del display ePaper (inizialmente apparirà bianco). Collegandosi alla rete e da essa a Internet, verranno recuperate le informazioni e una volta visualizzate tali informazioni, atteso il tempo necessario che la visualizzazione sia stabile, si spegnerà sia il display che il LED blu, presente sulla NodeMCU, ad indicare l'entrata in deep-sleep del modulo. Il ciclo si ripeterà aggiornando la situazione ogni 20 minuti (3 volte l'ora).



Ad ogni aggiornamento delle informazioni vedrete il display diventare bianco e poco dopo mostrare i nuovi dati meteo con le relative nuove icone e i parametri aggiornati.

CONCLUSIONI

Bene, con questo abbiamo spiegato tutto quanto riguarda la nostra stazione meteo; sebbene sia in realtà un client appoggiato a un server web di previsioni meteorologiche, il progetto presenta spunti interessanti e svolge egregiamente il compito che ci si aspetta.

Il bassissimo consumo ottenuto con la modalità deep-sleep del NodeMCU e l'assorbimento nullo dell'ePaper nei periodi tra un refresh e l'altro consentono il funzionamento a batterie, volendo, ricaricate da un piccolo pannello solare; infatti l'unico momento in cui la stazione assorbe una corrente significativa è quando il modulo ESP2866 trasmette in WiFi.



per il MATERIALE

I componenti utilizzati nel progetto elencati di seguito sono disponibili presso Futura Elettronica. La board transceiver Wi-Fi con ESP8266 (cod. NODEMCUE-SP) è in vendita a Euro 9,50, il display ePaper da 4,3 pollici (cod. EPAPERUART43) costa Euro 74,00 e il supporto con pannello in plexiglass tagliato al laser (cod. CASEMETEOSTATIO) è disponibile a Euro 14,00. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 • Fax: 0331-792287 - www.futurashop.it

Meteo sempre sotto controllo!

Sveglia con proiezione dell'ora e del meteo

Tutto sotto controllo grazie alla nuova sveglia con stazione meteo integrata che proietta l'ora e le previsioni meteorologiche sul soffitto o sulla parete. Il display retroilluminato della sveglia visualizza inoltre l'ora (nel formato a 12 o 24 ore), il mese, la data, il giorno della settimana, le previsioni meteo, il grafico della temperatura nell'arco delle 12 ore, la temperatura e l'umidità interna.



€ 36,00

Cod. TP0033

- ✓ Orologio al quarzo.
- ✓ Calendario perpetuo.
- ✓ Visualizzazione dell'ora 12 e 24 ore.
- ✓ Allarme con snooze.
- ✓ Temperatura interna in °C / °F.
- ✓ Registrazione temperatura e umidità minima e massima.
- ✓ Andamento della temperatura.

- ✓ Proiezione dell'ora e delle previsioni meteo con 4 icone (soleggiato, poco nuvoloso, nuvoloso, piovoso).
- ✓ Retroilluminazione a LED.
- ✓ Alimentazione: 2 batterie da 1,5 Volt tipo AAA (non comprese) oppure tramite adattatore di rete (incluso).
- ✓ Dimensioni: 142x116x65 mm.



€ 49,00

Cod. WH1170



PORTATA
SENSORE
50
METRI

Stazione meteo con orologio DCF

Stazione meteorologica composta da un'unità base e da un sensore esterno wireless per la rilevazione della temperatura e dell'umidità. Ora e data radiocontrollati, registrazione della temperatura e dell'umidità minima e massima, previsioni e tendenza meteo, modalità di visualizzazione dell'ora 12h/24h, sveglia con snooze, calendario perpetuo e indicatore di batteria scarica.



Prezzi IVA inclusa.

Disponibili anche:

Sveglia digitale con campanelli meccanici e doppio allarme



Consente di impostare differenti orari di allarme. Display retroilluminato, funzione snooze.

€ 11,80

Cod. WT705G

Sveglia radiocontrollata tramite segnale DCF



Display retroilluminato. Visualizza la temperatura interna, data e giorno della settimana. Formato ora: 12/24.

€ 13,00

Cod. TIP020

Termometro e igrometro con data e ora

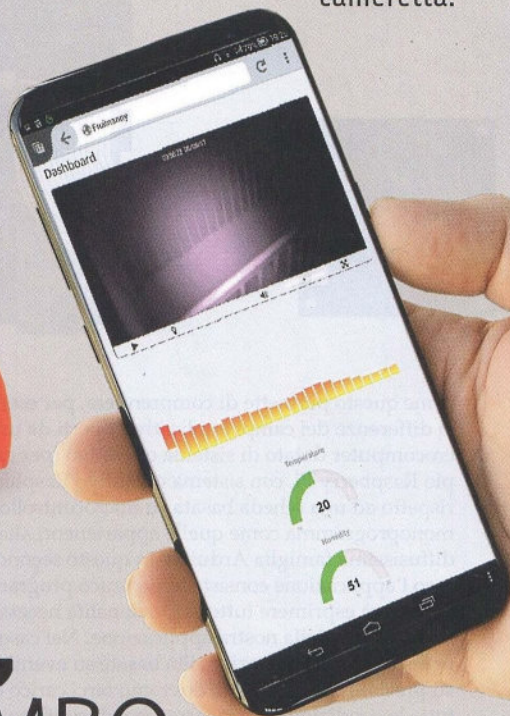


Con ampio display LCD visualizza la temperatura interna da -10 °C a +50 °C, l'umidità 10% - 99% RH, la data e l'ora.

€ 5,00

Cod. TERMOCLOCK

Il custode elettronico che veglia sul riposo del nostro bimbo e sulle condizioni ambientali della sua cameretta.

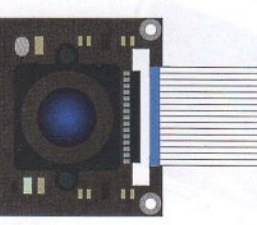


FRUITNANNY, OVVERO IL BIMBO A PORTATA DI RETE

di MARCO MAGAGNIN

A volte, "sfogliando" il web alla ricerca di soluzioni a problemi comuni, capita di imbattersi in progetti decisamente significativi dal punto di vista dell'impiego di risorse finalizzate a risolvere un problema che ci si è posto. Nel nostro caso si tratta di assicurarci che il nostro bimbo, durante i periodi di riposo, goda di un sonno tranquillo e nelle migliori condizioni ambientali possibili. Il progetto che ci ha incuriosito è stato sviluppato da Dmitry Ivanov, un ingegnere nel settore delle costruzioni, specializzato, guarda caso, nell'auto-

mazione di edifici. Nel tempo libero, tra le altre cose, si diletta nello sviluppare applicazioni informatiche, come quella di nome FruitNanny, che abbiamo notato visitando il suo sito e che vi presentiamo in questo articolo. Ci ha molto incuriosito la sua realizzazione in quanto è finalizzata a risolvere un problema molto comune, quale la supervisione del riposo di un bimbo, ideando una soluzione che supera i limiti di soluzioni simili, anche commerciali e, per di più, offre un'ottima opportunità, dal punto di vista didattico. La replica di un progetto



crocomputer dotato di sistema operativo Raspberry Pi, con sistema rispetto ad una scheda basata su monoprogramma come quella della diffusissima famiglia Arduino. In questo caso l'applicazione consiste nel far sì che dovrà esprimere tutte le

tri scambiando dati e messaggi. I moduli che devono funzionare separatamente, queste architetture sono da microcomputer dotati di software e da sistemi di software di gestione delle risorse disponibili per diverse applicazioni secondo

tettura che ha la finalità di

durante il riposo. L'ambiente stanza buia, o quantomeno del progetto è sorvegliare mera che ne catturi eventuali per esempio un risveglio a bimbo si mette a piangere, questa condizione per mezz

Per quanto riguarda in particolare l'acquisizione di immagini video, trattandosi di una stanza in penombra, dobbiamo per forza orientarci verso una telecamera a infrarossi. Per una maggior visibilità è anche opportuno dotare il nostro sistema di un'illuminazione tramite LED che emettono nel campo degli infrarossi, in modo da "illuminare" la stanza senza "svegliare" il bimbo con luce visibile. Siccome non staremo sempre incollati all'immagine della stanza, vogliamo poter accendere e spegnere l'illuminazione a nostro piacimento.

Infine, per semplificare l'interfaccia "utente" si vogliono raggruppare tutte le informazioni, immagini, suoni e grandezze misurate, in un'unica pagina web, fruibile da qualsiasi dispositivo dotato di browser come PC, tablet o smartphone. Come potete intuire, quello che vi presentiamo è un tipico progetto di "integrazione" di dispositivi e tecnologie. Le abilità richieste per portare a termine questi progetti non sono focalizzate unicamente sulla realizzazione di programmi ma sulla capacità di scegliere, installare e configurare applicazioni diverse, in modo che collaborino insieme. Una buona conoscenza dell'ambiente GNU/Linux, e Raspbian in particolare, è di sicuro aiuto per risolvere eventuali difficoltà che si possono presentare.

Per quanto riguarda un'introduzione di base a Raspberry Pi e all'ambiente Raspbian potete far riferimento al libro "Entra nel mondo di Raspberry Pi 3" ed ai numerosi articoli già apparsi sulla rivista. Infine nella rete web si trovano moltissimi aiuti e suggerimenti.

SCHEMA DEI COLLEGAMENTI

Nella Fig. 1 potete vedere lo schema della soluzione proposta nel progetto, modificata, rispetto al progetto originale, in base ai componenti che abbiamo potuto reperire con maggior facilità. In Fig. 2 sono visibili i riferimenti dei collegamenti al GPIO di Raspberry Pi. La configurazione "hardware" di "FruitNanny" è piuttosto semplice. L'elemento base è il microcomputer Raspberry Pi, nella versione 3. Se possedete già un Raspberry Pi 2 nel "cassetto" e volete destinarla a questo progetto, va benissimo, anche perché vi consigliamo di collegare Raspberry Pi alla rete locale di casa tramite cavo Ethernet, rendendo superflua la funzionalità WiFi. Se per diversi motivi preferite collegare Raspberry Pi via WiFi allora è preferibile utilizzare un Raspberry Pi 3 che dispone nativamente di connettività WiFi. Abbiamo raggruppato in questa descrizione sia l'elenco dei dispositivi utilizzati che la realizzazione dei collegamenti elettrici. La raccomandazione è di

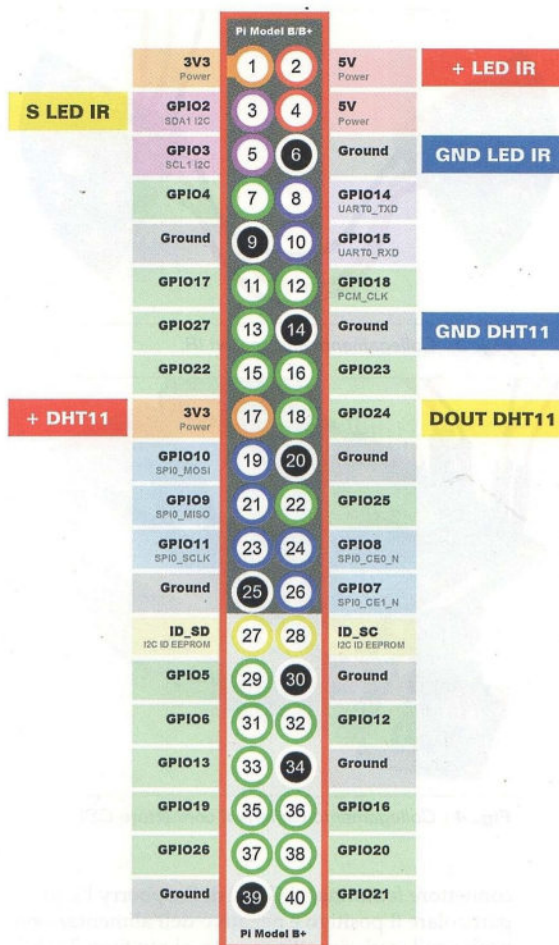


Fig. 2 - Collegamenti al GPIO di Raspberry Pi.

assicurarvi che Raspberry Pi sia scollegata dall'alimentazione prima di eseguire i collegamenti. Come vedete in Fig. 1 al microcontrollore sono collegati la microcamera a infrarossi, nel nostro caso il modello 2846-RASPBERRYCAMIR disponibile da Futura Elettronica. In alternativa è possibile utilizzare la camera a infrarossi ufficiale Pi Camera NoIR. Per quanto riguarda gli illuminatori agli infrarossi abbiamo utilizzato due kit FT1336K, collegati assieme a formare un illuminatore con otto LED IR. L'accensione e lo spegnimento avvengono portando in stato di conduzione o di interdizione i Mosfet BS170 presenti sul circuito, rispettivamente portando al livello alto (3,3V) o basso (GND) il Gate dei Mosfet stessi. Gli illuminatori IR vanno collegati in parallelo come visibile in Fig. 3 e collegati con cavetti, con

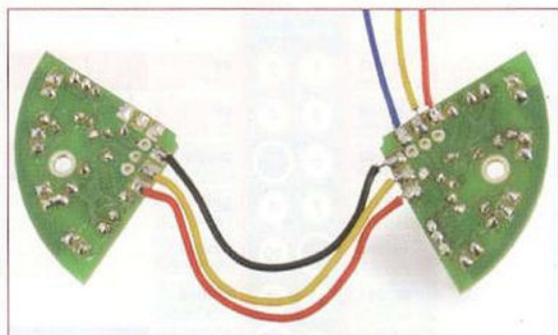


Fig. 3 - Collegamento illuminatori IR.



Fig. 4 - Collegamento camera al connettore CSI.

connettore femmina, al GPIO di Raspberry Pi. In particolare il positivo e negativo dell'alimentazione vanno collegati, rispettivamente, ai pin fisici 2 (+5V) e 6 (GND) del connettore GPIO mentre il cavetto di segnale (Gate dei Mosfet BS170) al pin fisico 3 (GPIO2).

Il cavo piatto della microcamera va inserito nello slot CSI (Camera Serial Interface) di Raspberry Pi situato tra il connettore HDMI ed il jack di uscita audio/video. Per inserire il connettore sollevate la cornicetta di blocco, bianca o nera a seconda dei

modelli, del connettore CSI, tirando dolcemente verso l'alto le alette laterali. Inserite con attenzione il cavo piatto della telecamera nel connettore CSI curando di tenere la parte blu rivolta verso il jack audio/video e i connettori argentati verso il connettore HDMI, come visibile in Fig. 4. Verificate che il cavo piatto sia totalmente inserito e ben allineato, poi chiudete la cornicetta di blocco.

Per il sensore di temperatura abbiamo utilizzato la scheda 2850-TEMPHUM con a bordo il sensore DHT11 che permette di misurare temperature comprese tra 0°C e +50°C e umidità relativa dal 20% al 90%. Comunica con Raspberry Pi mediante un'interfaccia seriale a singolo filo, nel progetto corrispondente al pin fisico 18 che corrisponde al pin BCM GPIO24. La comunicazione utilizza un protocollo basato su una serie di impulsi di handshaking iniziale seguiti da una sequenza di impulsi di trasmissione dei valori letti dai sensori, dalla durata dei quali si evince se si tratta di "zeri" o di "uno". La tensione positiva e negativa viene prelevata rispettivamente dai pin fisici 17 (+3,3V) e 14 (GND).

Il microfono USB, del tipo 2846-USBMICMINI, è semplicemente inserito in un connettore USB.

REALIZZAZIONE PRATICA

L'attività di realizzazione pratica del progetto consiste delle seguenti fasi principali:

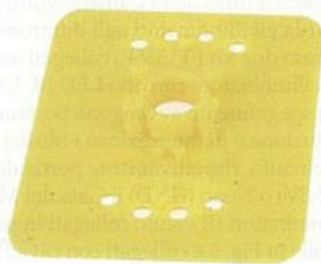


Fig. 5 - Contenitore stampato in 3D

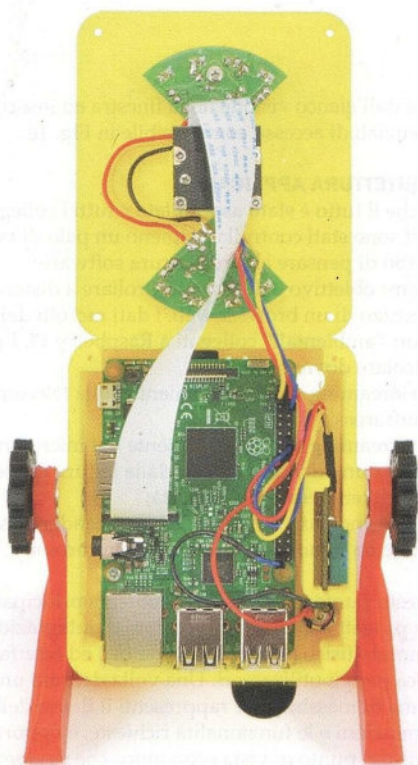


Fig. 6 - Disposizione hardware

1. Assemblaggio dei due kit FT1336K che richiedono alcune saldature. Aspettate a saldare i LED in quanto è bene "aggiustarli" prima all'interno del contenitore. Le istruzioni per il montaggio sono incluse nei kit stessi, mentre potete trovare la descrizione completa nel numero 218 della rivista come parte dell'articolo ArduTrain;
2. Realizzazione del cablaggio dei vari componenti tra loro e verso Raspberry Pi come visibile in Fig. 2;
3. Preparazione del contenitore, visibile nella Fig. 5. Nel nostro caso abbiamo realizzato il contenitore con un processo di stampa 3D;
4. Installazione e configurazione del software che costituisce l'applicazione. Vi consigliamo di installare e configurare il software prima di richiudere i componenti nella scatola, in modo da poter collegare monitor, tastiera e mouse. Tutto il processo di installazione e configurazione del software è descritto nel capitolo "Installazione Software";
5. Posizionamento dei componenti all'interno della scatola e chiusura della scatola stessa. La dispo-



Fig. 7 - Vista posteriore contenitore



Fig. 8 - Vista anteriore contenitore

sizione dei componenti è visibile in Fig. 6. Ricordate di saldare i cavetti al plug di alimentazione, posizionarlo nel suo alloggiamento sul retro della scatola e poi, dopo aver inserito Raspberry Pi, collegare positivo e negativo ai piedini 2 (+5V) e 39 (GND) del connettore GPIO. La scatola assemblata e chiusa è visibile nelle Fig. 7 e Fig. 8. Nel posizionamento dei componenti all'interno della scatola fate attenzione all'allineamento dei LED. Vi avevamo consigliato di aspettare a saldarli

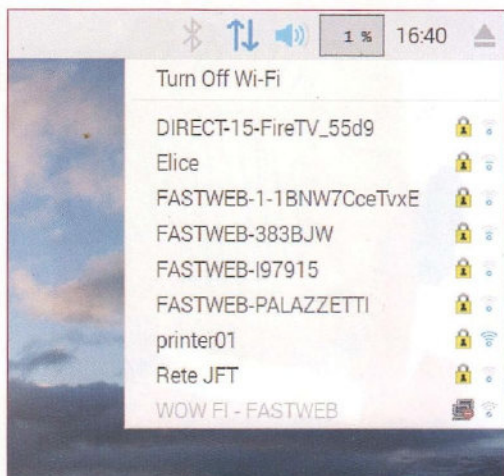


Fig. 9 - Configurazione WiFi - scelta rete.

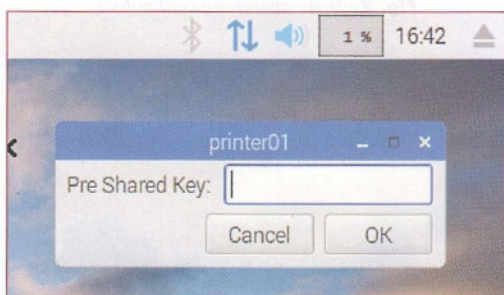


Fig. 10 - Configurazione WiFi - inserimento credenziali.

dopo aver posizionato gli illuminatori nel loro alloggiamento. Ricordate? Fate anche attenzione ad inserire Raspberry Pi nella scatola. Non forzatela. Il rischio è di incrinare la micro SD Card, rendendola inutilizzabile. Per inserire la telecamera togliete il "tappo" dell'obiettivo.

Una volta chiusa la scatola, rimane accessibile un connettore USB, al quale andrà collegato il microfono. Il connettore Ethernet può essere utilizzato per collegare FruitNanny alla rete locale, in alternativa al collegamento WiFi. Se volete collegare Raspberry Pi alla vostra rete locale in modalità WiFi dovete configurare la connessione di rete. Anche questa operazione è opportuno eseguirla prima di inserire Raspberry Pi nella scatola. Il metodo più semplice per eseguire la configurazione, è di utilizzare l'interfaccia grafica disponibile nel desktop. Questo richiede di utilizzare Raspberry Pi con monitor, tastiera e mouse. Aprite il configuratore di rete come visibile in Fig. 9. Selezionate la vostra rete

locale dall'elenco visibile nella finestra ed inserite le credenziali di accesso come visibile in Fig. 10.

ARCHITETTURA APPLICATIVA

Ora che il tutto è stato assemblato e tutti i collegamenti sono stati controllati almeno un paio di volte è tempo di pensare all'architettura software.

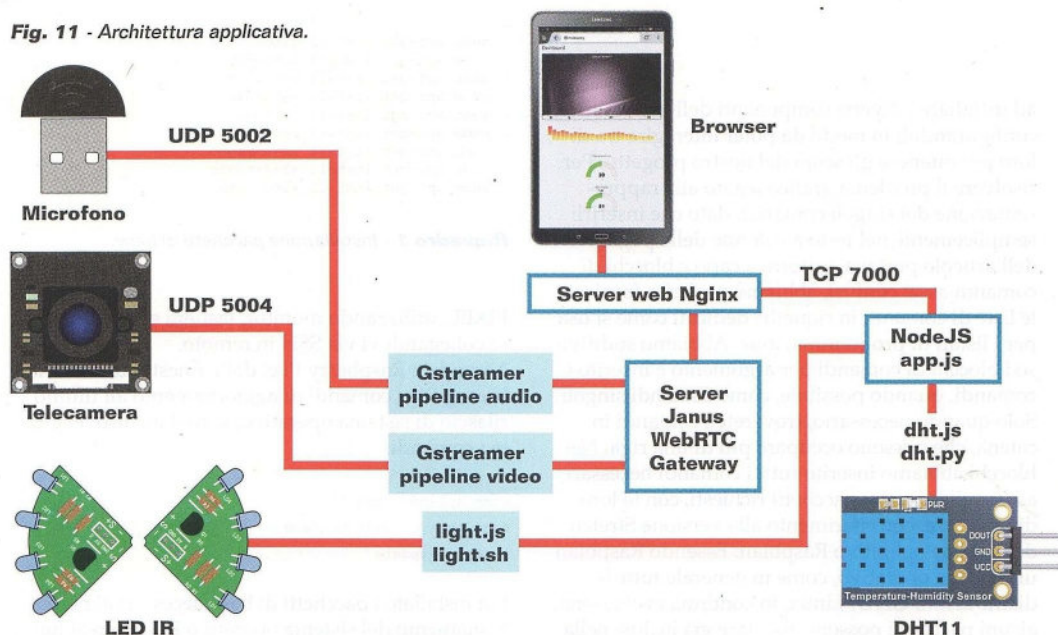
Il nostro obiettivo è di poter controllare a distanza, per mezzo di un browser web, i dati raccolti dai sensori "ambientali" collegati a Raspberry Pi. In particolare dovremo gestire:

- Lo streaming video proveniente dalla telecamera a infrarossi;
- Lo streaming audio proveniente dal microfono;
- I dati numerici provenienti dalla lettura dei sensori di temperatura e umidità;
- Infine vogliamo la possibilità di accendere e spegnere da remoto gli illuminatori infrarossi.

A questo punto si presenta il problema principale nella progettazione di applicazioni di integrazione tra mondo fisico (sensori ed attuatori) ed interfaccia grafica per gli utilizzatori. Una volta definito uno schema di massima che rappresenti il flusso delle informazioni e le funzionalità richieste, è opportuno, sia dal punto di vista economico che dei tempi di realizzazione, cercare di ricoprire la maggior parte delle funzionalità con pacchetti esistenti e disponibili nei repository della distribuzione di sistema operativo, nel nostro caso Raspbian, oppure sui vari siti web di produttori di sensori e o realizzatori di applicazioni open source e di libero utilizzo (le due condizioni non sempre si verificano congiuntamente). Senza dilungarci eccessivamente, in quanto quello che vi presentiamo è un articolo e non un tomo voluminoso sulla progettazione di applicazioni complesse nel mondo GNU/Linux, vi elenchiamo le scelte fatte nel progetto, che potete vedere in forma grafica nella Fig. 11.

- Per quanto riguarda il sistema operativo, la scelta non poteva che cadere su GNU/Linux ed in particolare sulla distribuzione Raspbian nella versione Stretch, l'ultima resa disponibile dalla Fondazione Raspberry Pi;
- Per la gestione dell'applicativo web, è necessario utilizzare un server web, cioè un'applicazione, in esecuzione permanente, in grado di acquisire le richieste dai browser che si collegano, di pagine e dati, e di rispondere utilizzando pagine già preconfezionate oppure richiamare le applicazioni in grado di produrre i dati all'occorrenza, nel nostro caso gli streaming audio e video ed i valori di

Fig. 11 - Architettura applicativa.



temperatura ed umidità provenienti dai sensori. La soluzione adottata in questo progetto è un misto tra le due tecnologie. Per questa funzione ci si avvale del pacchetto "Nginx" un server web molto leggero e configurabile che svolge ottimamente anche le funzioni di proxy, che sfrutteremo per la trasmissione dei flussi video e audio all'interno della pagina "contenitore";

- Gli streaming provenienti dalla camera a infrarossi e dal microfono sono, per loro natura, degli streaming multimediali ovvero flussi continui di pacchetti che devono essere trasmessi e ricevuti in sincronia, in modo da evitare di vedere immagini ... rallentate e flussi di suoni a ... singhiozzo. Per questa funzionalità si è utilizzato il pacchetto "GStreamer" una libreria specializzata nella costruzione di "pipeline" (flussi di trasporto) per la trasmissione di dati multimediali. In particolare il flusso video viene trasmesso utilizzando lo standard H264 senza modifiche, mentre il flusso audio viene compresso utilizzando il codec "Opus". Entrambi i flussi di streaming vengono trasmessi via socket TCP/IP (pacchetti UDP);
- Per collegare i flussi di streaming audio e video al browser, sul dispositivo utente, si utilizza il pacchetto "Janus WebRTC gateway". Come dice il nome questo pacchetto è un gateway ovvero un dispositivo di passaggio tra un'applicazione ed un'altra, in particolare tra i flussi di streaming ed il protocollo HTTP. La sigla RTC sta per Real

Time Communication;

- Per soddisfare le richieste dei dati di temperatura ed umidità e per eseguire i comandi di accensione e spegnimento degli illuminatori infrarossi è necessario utilizzare un server applicativo, un'applicazione in esecuzione permanente che recepisca le richieste dal browser ed esegua le letture ed i comandi richiesti. Questa è la parte realizzata personalmente da Dmitry Ivanov nell'ambito del progetto FruitNanny e disponibile sul suo repository GitHub. L'applicazione è realizzata in NodeJs, un ambiente di sviluppo ad eventi, programmabile in javascript. Sempre nel suo repository sono reperibili gli script in python che eseguono fisicamente le letture dei sensori e gli script bash per l'accensione e spegnimento degli illuminatori. Gentilmente, sono stati forniti anche tutti i file di configurazione dei vari pacchetti;
- Per semplificare la lettura dei dati dei sensori di temperatura e umidità, utilizzando l'interfaccia seriale monofilo, si utilizzerà la libreria "Adafruit_Python_DHT" che fornisce il software di base per interfacciare i sensori ed il wrapper per poter utilizzare le funzionalità della libreria nei programmi python.

INSTALLAZIONE SOFTWARE

Con l'hardware assemblato ed uno schema generale dell'architettura da realizzare si può iniziare

ad installare i diversi componenti dell'architettura, configurandoli in modo da poter interagire tra di loro per ottenere gli scopi del nostro progetto. Per risolvere il problema grafico legato alla rappresentazione dei singoli comandi, dato che inserirli semplicemente nel testo a colonne delle pagine dell'articolo portava a ritorni a capo e blocchi di comandi assai confusi, abbiamo preferito fornirvi le liste di comandi in riquadri dedicati come si usa per i listati di programmazione. Abbiamo suddiviso i blocchi di comandi per argomento e inserito i comandi, quando possibile, come comandi singoli. Solo quando necessario, troverete i comandi in catena, che possono occupare più di una riga. Nei blocchi abbiamo inserito tutti i comandi necessari ad installare tutti i pacchetti richiesti, con le loro dipendenze, con riferimento alla versione Stretch del sistema operativo Raspbian. Essendo Raspbian un sistema operativo, come in generale tutte le distribuzioni GNU/Linux, in continua evoluzione, alcuni pacchetti possono risultare già inclusi nella configurazione predefinita. In altri casi, soprattutto nel caso della catena audio, a causa degli aggiornamenti successivi, può essere necessario intervenire sulla configurazione di qualche elemento. In questi casi il web, e l'esperienza, sono di sicuro aiuto. In fondo il motivo per il quale vi presentiamo questo progetto è principalmente didattico. Collegare Raspberry Pi alla rete. Decidete se utilizzare una finestra di terminale dall'interfaccia grafica

```
sudo apt-get install git
sudo apt-get install libraspberrypi-dev
sudo apt-get install autoconf
sudo apt-get install automake
sudo apt-get install libtool
sudo apt-get install pkg-config
sudo apt-get install alsa-base
sudo apt-get install alsa-tools
sudo apt-get install alsa-utils
```

Riquadro 1 - Installazione pacchetti di base.

PIXEL, utilizzando monitor, tastiera e mouse, oppure collegandovi via SSH in remoto.

Accendete Raspberry Pi e, dalla finestra di terminale eseguite i comandi di aggiornamento all'ultimo rilascio di sistema operativo, kernel incluso. Digitate i comandi:

```
sudo apt-get update
sudo apt-get dist-upgrade
sudo rpi-update
```

Poi installate i pacchetti di base necessari al perfezionamento del sistema operativo Raspbian ai fini del nostro progetto. In pratica eseguite in sequenza i comandi elencati nel **Riquadro 1**. Noterete che per alcuni pacchetti il gestore "apt" troverà che sono già installati ed aggiornati. Nessun problema, proseguite con il comando successivo. Man mano che vengono rilasciate nuove release di Raspbian vengono inclusi in modo predefinito molti pacchetti. Il nostro elenco contiene comunque tutti quelli necessari.

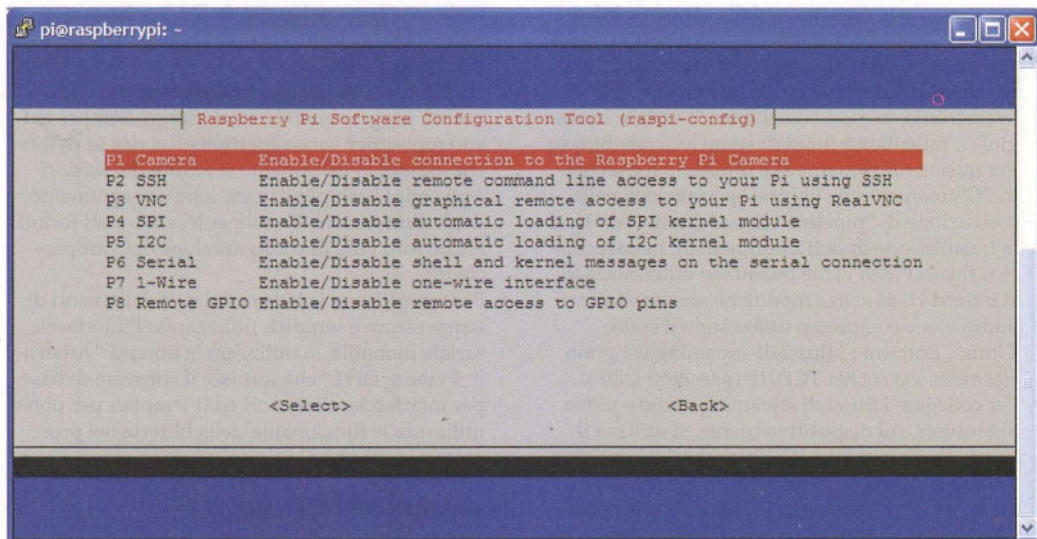


Fig. 12 - Abilitazione Camera Pi.

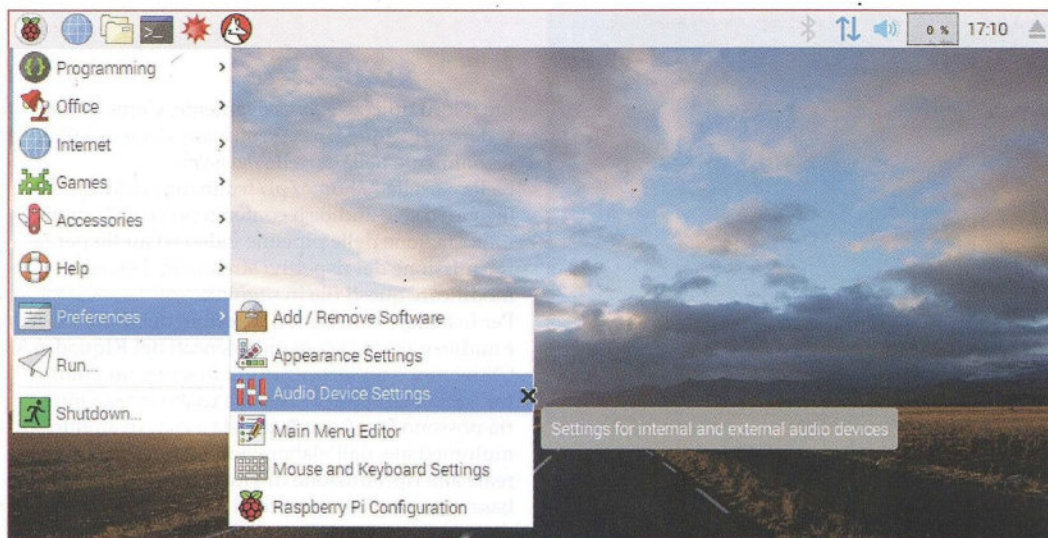


Fig. 13 - Configurazione microfono USB - Menu.

Ora è tempo di abilitare la telecamera. Per questa operazione utilizzeremo le funzionalità rese disponibili dall'applicazione di configurazione di Raspbian. Per eseguirlo digitate il comando:

```
sudo raspi-config
```

Dal menu selezionate "Interfacing Options" e nella pagina successiva la voce "P1 Camera", come visibile in Fig. 12.

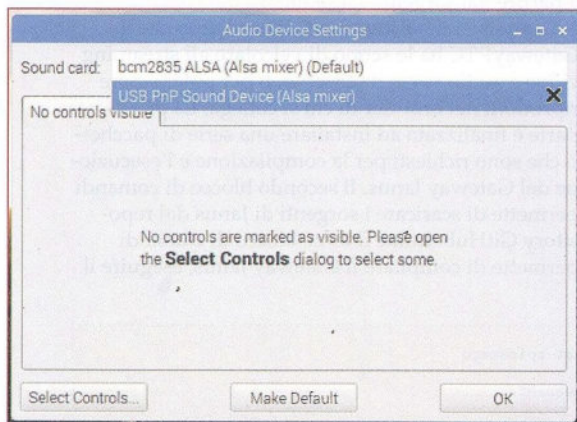
Selezionate "Yes" nella pagina di configurazione e poi "OK" nella pagina di richiesta di conferma. La prossima operazione che andiamo a descrivere

richiede necessariamente il desktop grafico, che, a sua volta, richiede l'utilizzo di monitor, tastiera e mouse, oppure con collegamento in desktop remoto. La configurazione serve a far riconoscere al sistema operativo, il microfono USB, come periferica predefinita di acquisizione suono. Dal menu aprite la funzionalità di configurazione "Audio Device Settings" come visibile in Fig. 13.

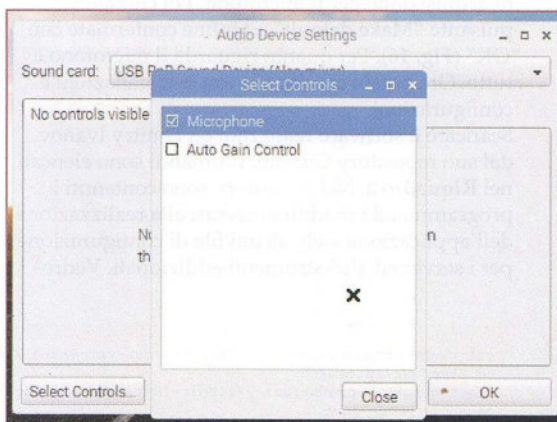
Dalla tendina in alto nella pagina selezionate il microfono USB (Fig. 14).

Nella sottopagina spuntate la voce "Microphone" (Fig. 15).

Nella nuova pagina, impostate il livello del volume



**Fig. 14 - Configurazione microfono USB
Scelta dispositivo.**



**Fig. 15 - Configurazione microfono USB
Scelta controllo.**

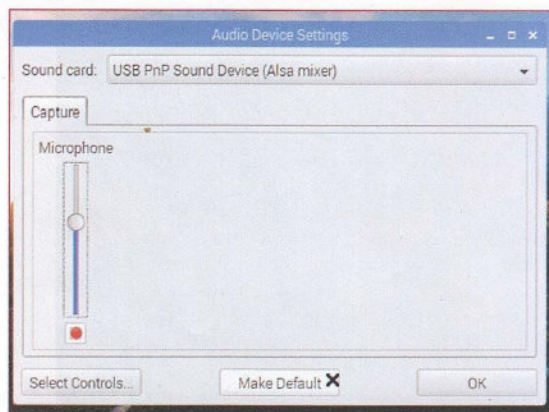


Fig. 16 - Configurazione microfono USB come predefinito.

```
cd /home/pi
sudo mkdir fruitnanny
sudo chown pi:pi fruitnanny
git clone https://github.com/ivadim/fruitnanny
chmod -R 755 /home/pi/fruitnanny
```

Riquadro 2 - Installazione pacchetto FruitNanny da GitHub.

```
sudo apt-get install gstreamer1.0-tools
sudo apt-get install gstreamer1.0-plugins-good
sudo apt-get install gstreamer1.0-plugins-bad
sudo apt-get install gstreamer1.0-plugins-ugly
sudo apt-get install libgstreamer1.0-dev
sudo apt-get install libgstreamer-plugins-base1.0-dev
sudo apt-get install gstreamer1.0-alsa
```

Riquadro 3 - Installazione Gstreamer.

di acquisizione per il microfono. Poi cliccate sul pulsante "Make default" ed infine confermate con "OK" (Fig. 16). Per quanto riguarda il microfono è tutto. Ora potete continuare con le installazioni e configurazioni.

Scaricate il software realizzato da Dmitry Ivanov dal suo repository GitHub. I comandi sono elencati nel Riquadro 2. Nel repository sono contenuti i programmi ed i moduli necessari alla realizzazione dell'applicazione web, alcuni file di configurazione per i server ed altri strumenti aggiuntivi. Vedre-

```
git clone https://github.com/thaytan/gst-rpicamsrc /tmp/gst-rpicamsrc
cd /tmp/gst-rpicamsrc
./autogen.sh --prefix=/usr --libdir=/usr/lib/arm-linux-gnueabi/hf/
make
sudo make install
```

Riquadro 4 - Installazione plugin Gstreamer per Camera Pi.

mo il loro utilizzo successivamente. Come potete vedere nel Riquadro 2, il software viene installato inizialmente nella cartella `/home/pi/`.

Nella cartella `/home/pi/fruitnanny/bin/` i file `video.sh` e `audio.sh` contengono i comandi di impostazione delle pipeline video ed audio per la trasmissione dei rispettivi streaming. Dovremo modificare questi file in seguito.

Per installare GStreamer, il gestore dei flussi video e audio, eseguite i comandi elencati nel Riquadro 3. GStreamer è un'infrastruttura di streaming multimediale. Le applicazioni che usano questa libreria possono fare (quasi) qualsiasi cosa in ambito multimediale, dall'elaborazione del suono in tempo reale alla riproduzione di video. La sua architettura, basata su plugin, consente di aggiungere nuovi tipi di dati o capacità di riproduzione semplicemente installando i plugin interessati, nel nostro caso quelli per la gestione di video ed audio.

Tanto per cambiare, il repository ufficiale di Gstreamer non prevede i plugin necessari a riconoscere le camere CSI per Raspberry Pi. Fortunatamente il Sig. Jan Schmidt, australiano, ha realizzato e reso disponibile nel suo repository GitHub i sorgenti del pacchetto "gst-rpicamsrc", un wrapper costruito "attorno" alle funzionalità "raspivid" e "raspistill" in grado di trasformare i flussi provenienti dalla Camera Pi nel formato atteso da GStreamer. Essendo il pacchetto in formato sorgente, dopo averlo scaricato è necessario compilarlo ed installarlo. Fortunatamente uno script bash fornito da Jan Schmidt facilita di molto il processo. I comandi li trovate nel Riquadro 4.

Anche il Gateway RTC Janus deve essere installato a partire dai sorgenti. La sequenza dei comandi è disponibile nel Riquadro 5. Ricordiamo che il GatewayRTC ha lo scopo di veicolare gli streaming video e audio verso il server web per poter essere riprodotti nel browser di chi si collega. La prima parte è finalizzata ad installare una serie di pacchetti che sono richiesti per la compilazione e l'esecuzione del Gateway Janus. Il secondo blocco di comandi permette di scaricare i sorgenti di Janus dal repository GitHub. Infine il terzo blocco di comandi permette di compilare il Gateway Janus, eseguire il

build ed installare gli eseguibili ed i file di compilazione nelle cartelle di destinazione.

I file di configurazione di Janus adatti al nostro progetto sono disponibili nella cartella:

/home/pi/fruitnanny/configuration/janus/:

- janus.cfg – File di configurazione generale per Janus. Rispetto al file di configurazione originale vengono disabilitati alcuni plugin non necessari nel nostro progetto;
- janus.transport.http.cfg – Abilita l'accesso a Janus da protocollo HTTP;
- janus.plugin.streaming.cfg – Configura gli streaming audio e video in modo da presentarli rispettivamente sulla porta TCP 5002 per lo streaming

audio e 5004 per lo streaming video.

Per configurare correttamente il Gateway RTC Janus è sufficiente copiare i file di configurazione elencati sopra nella cartella di configurazione di Janus. I comandi sono riportati nel **Riquadro 6**. Il passo successivo consiste nell'installazione di NodeJs e di alcuni pacchetti aggiuntivi, per permettere la comunicazione tra i programmi che eseguono le letture dei sensori ed il server web, in modo da esporre sulla pagina web la rappresentazione grafica di temperatura ed umidità. Il tutto elencato nel **Riquadro 7**.

Di questi tempi la privacy non è mai troppa, soprattutto quando si tratta di minori, quindi è bene crittografare la comunicazione degli streaming

1- Installazione pacchetti prerequisiti

```
sudo apt-get install libmicrohttpd-dev
sudo apt-get install libjansson-dev
sudo apt-get install libnice-dev
sudo apt-get install libssl-dev
sudo apt-get install libsrtp-dev
sudo apt-get install libsofia-sip-ua-dev
sudo apt-get install libglib2.0-dev
sudo apt-get install libopus-dev
sudo apt-get install libogg-dev
sudo apt-get install pkg-config
sudo apt-get install gengetopt
sudo apt-get install libsrtp2-dev
```

2- Scarico sorgenti Janus

```
git clone https://github.com/meetecho/janus-gateway /tmp/janus-gateway
cd /tmp/janus-gateway
git checkout v0.2.5
```

3- Compilazione ed installazione

```
sh autogen.sh
./configure --disable-websockets --disable-data-channels --disable-rabbitmq --disable-mqtt
make
sudo make install
```

Riquadro 5 - Installazione GatewayRTC Janus.

```
sudo cp /home/pi/fruitnanny/configuration/janus/janus.cfg /usr/local/etc/janus
sudo cp /home/pi/fruitnanny/configuration/janus/janus.plugin.streaming.cfg /usr/local/etc/janus
sudo cp /home/pi/fruitnanny/configuration/janus/janus.transport.http.cfg /usr/local/etc/janus
```

Riquadro 6 - Copia file di configurazione GatewayRTC Janus.

```
cd /home/pi/fruitnanny
curl -sL https://deb.nodesource.com/setup_6.x | sudo -E bash -
sudo apt install -y nodejs

stando in /home/pi/
npm install express
npm install ejs
```

Riquadro 7 - Installazione nodejs.


```
cd /usr/local/share/janus/certs
sudo openssl req -x509 -sha256 -nodes -days 365 -newkey rsa:2048 \
-keyout mycert.key -out mycert.pem
```

Riquadro 8 - Generazione certificato SSL.

```
git clone https://github.com/adafruit/Adafruit_Python_DHT /home/pi/Adafruit_Python_DHT
cd /home/pi/Adafruit_Python_DHT
sudo apt-get install build-essential python-dev python-pip
sudo python setup.py install
```

Riquadro 9 - Installazione libreria Adafruit_Python_DHT.

audio e video utilizzando il protocollo HTTPS. Per fare questo è necessario generare un certificato SSL (Secure Socket Layer) che contenga le chiavi di crittografia e consenta di certificare la comunicazione. Per questo dobbiamo utilizzare i comandi indicati nel Riquadro 8. Lo stesso certificato sarà utilizzato per accedere, dal browser, al server web Nginx mediante il protocollo HTTPS. Durante la generazione del certificato vi verranno chieste alcune informazioni che permettono di personalizzare e rendere univoco il certificato stesso.

Listato 1 - audio.service

```
[Unit]
Description=FruitNanny audio service
After=network.target

[Service]
User=pi
Type=simple
ExecStart=/home/pi/fruitnanny/bin/audio.sh
LimitNOFILE=65536
Restart=always
KillSignal=SIGTERM
KillMode=process

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

Listato 2 - video.service

```
[Unit]
Description=FruitNanny video service
After=network.target

[Service]
User=pi
Type=simple
ExecStart=/home/pi/fruitnanny/bin/video.sh
LimitNOFILE=65536
Restart=always
KillSignal=SIGTERM
KillMode=process

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

Rispondere alle domande man mano che vi vengono poste fino al completamento della procedura. Per poter accedere al GPIO per la lettura dei sensori e per l'accensione e spegnimento degli illuminatori infrarossi, senza dover ricorrere all'utente "root", è necessario far partecipare l'utente con il quale ci colleghiamo a Raspberry Pi (in genere "pi") al gruppo "gpio". Lo si ottiene con il comando:

```
sudo adduser $USER gpio
```

Abbiamo anticipato che il sensore di temperatura ed umidità DHT11 comunica mediante un protocollo seriale monofilo personalizzato. Adafruit ha realizzato una libreria che rende trasparente il protocollo seriale e permette di acquisire i dati dei sensori direttamente dai programmi python. Non resta che installare la libreria, prelevandola dal repository GitHub di Adafruit.

I comandi sono nel Riquadro 9.

Una delle funzionalità necessarie in un'applicazione "autonoma" o, come si dice "stand-alone" è che tutto parta automaticamente all'accensione del dispositivo che la ospita. Per far questo vi sono diverse possibilità. La più professionale è di affidare questo compito al processo di inizializzazione di tutti i processi di sistema, al termine della fase di boot. Il processo è "systemd" un notevole miglioramento, inserito nelle ultime versioni della distribuzione Raspbian, che ha sostituito il "vecchio" e più lento "initd".

Nel nostro caso gli script di auto start sono già disponibili nel repository. Prima però dobbiamo modificarne un paio per rendere coerenti i riferimenti interni. Cominciamo dallo script di auto start dello streaming audio "/home/pi/fruitnanny/configuration/systemd/audio.service". Lo dovete modificare come visibile nel Listato 1.

Ora modificate lo script di autostart dello streaming video "/home/pi/fruitnanny/configuration/systemd/video.service". Lo dovete modificare come visibile nel Listato 2. Dopo aver salvato le modifi-

che copiate gli script di auto start nella cartella "system" di "systemd" e abilitate l'autostart. I comandi sono visibili nel **Riquadro 10**.

A questo punto gli streaming audio, video e Janus vengono eseguiti come servizi, e possono essere gestiti con i comandi "systemctl". Per esempio per fermare un servizio digitate il comando:

```
sudo systemctl stop SERVICE_NAME.
```

Oltre a "stop" sono disponibili i comandi "start", "restart" e "status". Ancora un po' di pazienza, siamo verso la fine.

Dobbiamo installare il server web Nginx, che avete già avuto modo di conoscere nell'articolo a lui dedicato nel numero 224 della rivista. In questo progetto abbiamo il vantaggio di avere già disponibili i file

di configurazione. Seguite la sequenza di comandi elencati nel **Riquadro 11**. Prima però è doverosa un'avvertenza. A seguito dell'ultimo comando vi verrà richiesto di inserire una password per l'utente "fruitnanny" e di confermarla inserendola una seconda volta.

Se volete, potete cambiare l'utente "fruitnanny" con uno a vostro piacere.

Ricordatevi di prendere nota sia del nome utente che della password. Sono le uniche credenziali che vi permetteranno di accedere al mini-sito associato al nostro "guardiano". Lo scopo di nginx è di far raggiungere gli streaming provenienti da Janus e gestire le richieste verso nodejs da un'unica pagina e da un unico indirizzo URL (http://<indirizzo_IP_di_RaspberryPi>). Il proxy Nginx è impostato per instradare le richieste verso nodejs all'indirizzo

1- Copia degli script di autostart

```
sudo cp /home/pi/fruitnanny/configuration/systemd/audio.service /etc/systemd/system/
sudo cp /home/pi/fruitnanny/configuration/systemd/video.service /etc/systemd/system/
sudo cp /home/pi/fruitnanny/configuration/systemd/janus.service /etc/systemd/system/
```

2 - Abilitazione e all'autostart e start dei processi:

```
sudo systemctl enable audio
sudo systemctl start audio
sudo systemctl enable video
sudo systemctl start video
sudo systemctl enable janus
sudo systemctl start janus
```

Riquadro 10 - Autostart audio, video e Janus.

1- Installazione Nginx

```
sudo apt-get install nginx
```

2- Rimozione dei file di configurazione predefiniti

```
sudo rm -f /etc/nginx/sites-enabled/default
```

3 - Modifica file configurazione Nginx

Andate alla cartella /home/pi/fruitnanny/configuration/nginx/

Per entrambi i file fruitnanny_http e fruitnanny_https modificare la configurazione

```
root /opt/fruitnanny;
```

in

```
root /home/pi/fruitnanny;
```

4- Copia dei file di configurazione per il progetto

```
sudo cp /home/pi/fruitnanny/configuration/nginx/fruitnanny_http /etc/nginx/sites-available/fruitnanny_http
sudo cp /home/pi/fruitnanny/configuration/nginx/fruitnanny_https /etc/nginx/sites-available/fruitnanny_https
```

5- Abilitazione dei nuovi file di configurazione

```
sudo ln -s /etc/nginx/sites-available/fruitnanny_http /etc/nginx/sites-enabled/
sudo ln -s /etc/nginx/sites-available/fruitnanny_https /etc/nginx/sites-enabled/
```

6- Impostazione di un utente e password per l'accesso da web

```
sudo sh -c "echo -n 'fruitnanny:' >> /etc/nginx/.htpasswd"
sudo sh -c "openssl passwd -apr1 >> /etc/nginx/.htpasswd"
```

Riquadro 11 - Installazione e configurazione server web nginx.


```
@reboot nohup bash /home/pi/fruitnanny/node_start.sh /dev/null >2 /dev/null &
```

Riquadro 12 - Configurazione schedatore Cron.

Listato 3 - app.js

```
"use strict";
const dht_1 = require("./routes/dht");
const express = require("express");
const light_1 = require("./routes/light");
let app = express();
let config = {
  baby_birthday: "2016-03-15",
};
app.set("view engine", "ejs");
app.set("views", "server/views");
app.use("/public", express.static("public"));
app.get("/", (req, res, next) => {
  res.render("index", { config });
});
app.use("/api/light", light_1.default);
app.use("/api/dht", dht_1.default);
app.listen(7000, () => {
  console.log("Example app listening on port 7000!");
});
```

Listato 4 - routes/dht.js

```
"use strict";
const express = require("express");
let exec = require('child_process').exec;
let router = express.Router();
router.get("/current", (req, res, next) => {
  exec("python bin/dht22.py", function(err, stdout, stderr){
    let values = stdout.split(" ");
    let t = values[0];
    let h = values[1];
    let result = { humidity: h, temperature: t };
    res.json(result);
  });
});
Object.defineProperty(exports, "__esModule", { value: true });
exports.default = router;
```

Listato 5 - /bin/dht22.py

```
#!/usr/bin/python
import sys
import Adafruit_DHT

pin = 24
humidity, temperature = Adafruit_DHT.read_retry(Adafruit_DHT.DHT11, pin)
if humidity is not None and temperature is not None:
    print('{0:0.1f} {1:0.1f}'.format(temperature, humidity))
else:
    print('Failed to get reading. Try again!')
    sys.exit(1)
```

(<http://127.0.0.1:7000>) e verso il server Janus all'indirizzo (<https://127.0.0.1:8089>).

Per eseguire il punto 3 del Riquadro 12 utilizzate la funzionalità WinSCP. Aprite i file indicati, eseguite le modifiche e salvate. Ora qualche aggiustamento per fare in modo che tutto funzioni a dovere. Modificate il file `/home/pi/fruitnanny/server/app.js`

come indicato nel Listato 3. Modificate il file `/home/pi/fruitnanny/server/routes/dht.js` come visibile nel Listato 4. Modificate il file `/home/pi/fruitnanny/bin/dht22.py` come visibile nel Listato 5. Intanto che avete lo strumento WinSCP aperto, copiate la cartella `/home/pi/fruitnanny/public` nella cartella `/home/pi/fruitnanny/views` (trascinatela



Fig. 17 - Pagina web di controllo del progetto FruitNanny.

tenendo premuto CTRL). Questo accorgimento risolve alcuni problemi di indirizzamento all'interno della pagina web. Ora è necessario lanciare l'applicazione "app.js" e fare in modo che venga eseguita automaticamente al momento del boot. Per questo create nella cartella /home/pi/fruitnanny un file di nome "node_start.sh" con il contenuto visibile nel Listato 6. Per fare in modo che questo file venga eseguito al boot utilizziamo lo schedulatore cron con l'opzione "@reboot". Aprite lo schedulatore cron con il comando:

```
crontab -e
```

Inserite in fondo al file di configurazione il comando che trovate nel **Riquadro 12**. Salvate con CTRL+X e confermate. Eseguite un reboot. Ora tutto dovrebbe funzionare. Da un browser di un qualunque dispositivo collegato alla rete locale digitate l'URL

Listato 6 - node_start.sh

```
#!/bin/bash
cd /home/pi/fruitnanny
nohup node server/app.js > /dev/null & 2> /dev/null &
```

http://<indirizzo_IP_di_Raspberry_Pi>

Vi verranno chiesti utente e password per l'accesso. Ricordate? Una volta inserite le credenziali apparirà la pagina visibile in Fig. 17. Il significato dei pulsanti dovrebbe essere intuitivo. I colori degli indicatori di temperatura rosso, giallo o verde danno l'indicazione di quanto sia confortevole o meno l'ambiente di riposo per il nostro bimbo. A questo punto, se avete avuto la pazienza e la costanza di arrivare a concludere il progetto, non vi dovrebbero essere più dubbi sulle differenze di funzionalità e di destinazione tra un dispositivo dotato di microcontrollore ed un computer dotato di sistema operativo. Quello che vi abbiamo appena presentato è comunque un progetto di integrazione tra microcontrollori e microcomputer. Sia la Camera Pi che il sensore DHT sono dotati di microcontrollori che acquisiscono immagini e dati e comunicano con il microcomputer utilizzando protocolli digitali. Per chi volesse approfondire le funzionalità dei pacchetti utilizzati in questo progetto la migliore fonte di informazioni è sicuramente la rete web, oltre che qualche buon libro. Non possiamo che augurare al nostro bimbo sogni d'oro e buon lavoro a chi vorrà cimentarsi nella realizzazione di questo progetto.



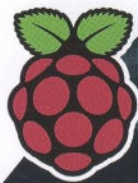
per il MATERIALE

Il materiale per realizzare il progetto è disponibile presso Futura Elettronica. La board Raspberry Pi 3 Tipo B+ (cod. RPI3BPLUS) è in vendita a Euro 44,00, la telecamera IR per Raspberry Pi (cod. RASPBERRYCAMIR) è disponibile a Euro 35,00, il modulo a LED IR (cod. FT1336K) costa Euro 4,80, la scheda con sensore temperatura e umidità DHT11 (cod. TEMPHUM) è in vendita a Euro 5,40, il mini microfono USB (cod. USBMICMINI) è in vendita a Euro 2,90 e il case per FruitNanny in stampa 3D (cod. CASEFRUITNANNY) costa Euro 75. I prezzi si intendono IVA compresa.

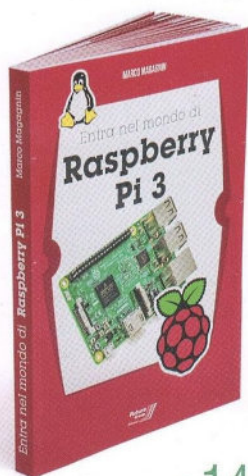
Il materiale va richiesto a:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - Fax: 0331-792287 - www.futurashop.it

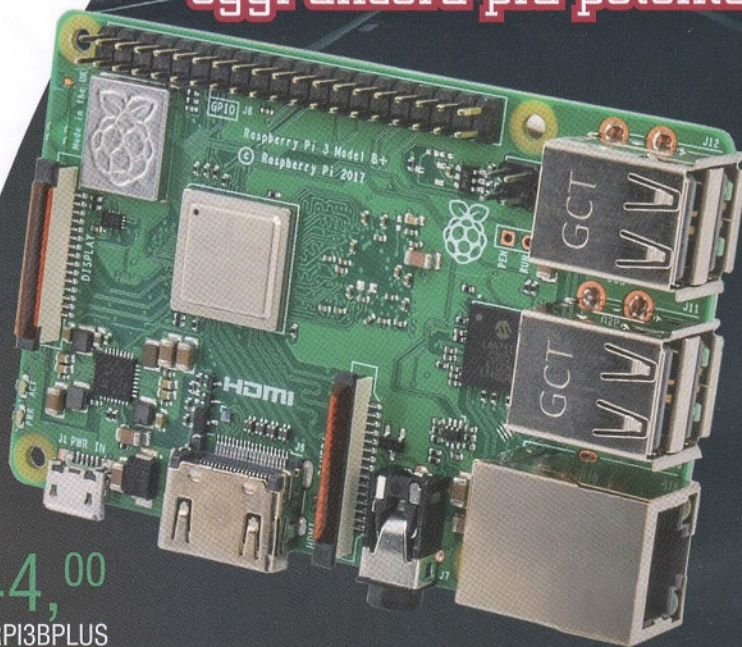
Entra nel mondo di Raspberry Pi 3



Scopri la nuovissima Raspberry Pi 3 B+ oggi ancora più potente.



€ 14,00
cod. RASPB00KPI3



€ 44,00
cod. RPI3BPLUS

Questo libro è rivolto a chi vuole avvicinarsi al mondo dei microcomputer dotati di sistemi GNU/Linux focalizzando l'attenzione sul recente modello Raspberry Pi 3, il più potente e contemporaneamente il più semplice da usare della serie. Una guida completa con spiegazioni, progetti pratici e schede per introdurre il lettore all'utilizzo iniziale, senza trascurare l'approfondimento di alcune funzioni avanzate disponibili nell'affascinante mondo della progettazione con Raspberry Pi.

SoC	Broadcom BCM2837B0
CPU	quad core 1,4 GHz 64bit ARM CortexA53
GPU	4 (su hub USB 5 porte)
Memory	1 GB (condivisa con la CPU)
USB	4 porte versione 2.0 su hub 5 porte
Ingressi video	connettore 15pin MIPI, connettore CSI (camera interface) per Raspberry Pi camera o Raspberry Pi NoIR camera
Uscite video	HDMI, videocomposito (jack 3,5 mm), MIPI display interface (DSI) per pannelli LCD
Uscite audio	analogica (jack 3,5 mm) e digitale da HDMI e I ² S
Onboard storage	slot microSDHC, USB (Boot Mode)
Rete	Gigabit Ethernet a 300 Mbps, wireless 802.11ac, Bluetooth 4.2, BLE Power over Ethernet (PoE) con HAT
I/O	Header con 40 GPIO
Consumo	mediamente 300 mA (1,5 W) a riposo e 1,34 A (6,7W) al massimo (con monitor, tastiera, mouse e WiFi utilizzati)
Alimentazione	5,1V da microUSB, GPIO oppure da shield PoE

Prezzi IVA inclusa.

Aeroporti: maggiore sicurezza grazie al Politecnico di Milano

Giacomo Boracchi e Cesare Alippi del Dipartimento di Elettronica, Informazione e Bioingegneria del Politecnico di Milano hanno sviluppato una ricerca che ha portato, in collaborazione con Gilardoni Raggi X di Mandello del Lario (LC), ad un algoritmo capace di individuare automaticamente sostanze esplosive nascoste nei bagagli.

L'algoritmo ha permesso un sostanziale miglioramento rispetto alle soluzioni precedenti. Il sistema potrà essere impiegato in macchinari a raggi X "a doppia vista" che permettono una rapida scansione dei bagagli e sono ampiamente utilizzati per i controlli in aeroporto e in altre infrastrutture critiche ed affollate come tribunali e dogane.

Per ottenere l'algoritmo sono state utilizzate tecniche avanzate di elaborazione dell'immagine e di intelligenza artificiale, in particolare di machine learning, che hanno permesso di migliorare la qualità delle immagini e allo stesso tempo di raggiungere un'ottima accuratezza nel riconoscimento delle

sostanze esplosive. La macchina evidenzia le sostanze ritenute pericolose con un riquadro e invita l'operatore ad ulteriori accertamenti.

La progettazione dell'algoritmo e la sua ingegnerizzazione è avvenuta in stretta collaborazione con Gilardoni Raggi X. Grazie a queste ricerche, la macchina per i controlli di sicurezza aeroportuale prodotta dall'azienda è tra le prime al mondo ad aver raggiunto l'importante certificazione C1 degli standard europei ECAC.

Come le radiografie, le macchine per i controlli di sicurezza aeroportuale permettono di visualizzare il contenuto dei bagagli. I sistemi "a doppia vista" sono dotati di due sorgenti radiogene che acquisiscono due immagini da prospettive diverse, in particolare "dall'alto" e "da lato". La colorazione delle immagini indica il materiale contenuto nei bagagli (ad esempio in marrone le so-

stanze organiche, in verde l'alluminio, il vetro e le plastiche pesanti, in blu gli altri metalli).

L'algoritmo sviluppato dal Politecnico di Milano è al cuore dell'Explosive Detection System for Cabin Baggage (EDS-CB), un software in grado di rivelare automaticamente le sostanze esplosive presenti nei bagagli a mano. Queste sostanze vengono visualizzate sullo schermo presieduto dall'operatore alla sicurezza con un contorno rosso, invitando l'addetto a fare ulteriori accertamenti.

www.polimi.it



Primo volo supersonico per Virgin Galactic

Dopo il tragico incidente del 2014, il velivolo SpaceShipTwo VSS Unity, ha completato con successo il primo volo supersonico; lo ha annunciato lo stesso Richard Branson, rilanciando il progetto di voli suborbitali a scopo turistico. La Vss Unity ha sorvolato il deserto del Mojave, in California a bordo del velivolo WhiteKnightTwo VMS Eve raggiungendo circa 14.000 metri di quota.

Dopo pochi secondi Unity ha acceso i motori stac-

candosi dalla navetta madre accelerando fino a raggiungere la soglia di Mach 1.87 in circa 30 secondi di propulsione del razzo. In questa fase, il motore ibrido costruito dalla Spaceship Company ha permesso a Unity di effettuare per la prima volta il volo supersonico.

Durante lo spegnimento del razzo, Unity ha continuato la sua corsa verso l'alto fino a oltre 25.000 metri di altezza prima di iniziare le manovre di discesa. Successivamente i piloti hanno aperto il sofisticato sistema alare che, grazie al suo design unico, permette il rientro del velivolo in modo affidabile, come

un vero e proprio aereo. Il successo del volo, non può che sostenere gli ambiziosi programmi di Branson per il futuro, tra cui figura la sperimentazione di voli suborbitali in accordo con Altec e Thales Alenia Space che prevede la costruzione di uno spaziorpporto sul suolo italiano. E non è tutto: il magnate inglese ha anche pianificato l'inserimento della sua Virgin Orbit nel mercato riguardante la fornitura di servizi di lancio per piccoli satelliti.

www.virgingalactic.com



Superchip con struttura ottica integrata

Un nuovo modo di integrare strutture ottiche sui tradizionali processori potrebbe rendere la comunicazione tra i vari chip molto più veloce aumentando la potenza di elaborazione e riducendo le perdite energetiche. *"Quello di cui stiamo parlando è l'integrazione di circuiti ottici con il materiale semiconduttore con cui è realizzato attualmente il chip"*, afferma Milos Popovic, professore di ingegneria elettrica e informatica presso la Boston University.

"Il metodo prevede di aggiungere pochi passaggi all'attuale processo standard con cui viene trattato il silicio e non dovrebbe comportare un significativo aumento di costi o tempi al processo di produzione", aggiunge Popovic.

Lui, insieme ai colleghi del Massachusetts Institute of Technology, dell'Università della California (Berkeley), dell'Università del Colorado (Boulder) e del SUNY Polytechnic Institute di Albany, NY, ha descritto il metodo in un recente articolo su Nature.

Il loro approccio prevede l'aggiunta di un sottile strato di silicio policristallino sulla struttura già esistente del chip, materiale simile a quello del dielettrico di

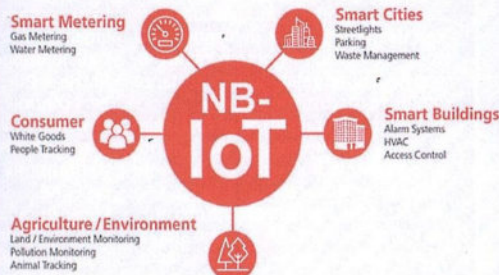
gate, ma in una forma che assorbe troppa luce per essere utile come una guida d'onda.

Per rendere questo materiale più adatto ai requisiti della fotonica, i ricercatori hanno ottimizzato il processo di deposizione, alterando fattori come la temperatura, per ottenere una diversa struttura cristallina. Hanno anche utilizzato degli strati isolanti con biostrati di silicio, già utilizzato per isolare elettricamente i transistor l'uno dall'altro, ma con tracce più profonde, per impedire alla luce di passare dal silicio policristallino al substrato di silicio.

Con questo approccio i ricercatori hanno costruito chip con tutti i componenti fotonici necessari: guide d'onda, risonatori ad anello, accoppiatori verticali, modulatori ad alta velocità e fotodetettori a valanga, oltre a transistor con tecnologia 65 nm. Il chip, che necessita di una sorgente laser esterna, è realizzato con la tecnologia a 65 nm perché è quella disponibile presso i laboratori di SUNY Albany, ma secondo Popovic lo stesso processo può essere applicato anche ai transistor realizzati con dimensioni inferiori.

www.bu.edu

Vodafone: a settembre copertura NB-IoT completata



Vodafone Italia accelera sull'implementazione della tecnologia Narrowband IoT (Narrowband Internet of Things) e annuncia la copertura del 100% dei propri siti 4G entro settembre 2018, per creare una rete in grado di connettere qualsiasi tipologia di oggetto, dalle biciclette ai container, fino ai contatori dell'acqua. L'Internet delle cose rappresenta il prossimo capitolo della trasformazione del paese, di cui Vodafone è protagonista, anche attraverso la sperimentazione 5G che sta guidando a Milano. La copertura nazionale attraverso la tecnologia Narrowband IoT renderà possibili nuove applicazioni nei campi del monitoraggio ambientale e strutturale, dell'agricoltura di precisione, delle smart cities e delle smart utilities. Ad esempio sarà possibile connettere e monitorare contatori per l'efficienza dei sistemi idrici, cassonetti dell'immondizia intelligenti e parcheggi.

Vodafone oggi connette circa 62 milioni di oggetti nel mondo, e 7,5 in Italia, dati in costante aumento grazie alla rapida espansione della tecnologia IoT in tutti i mercati e che incrementeranno ulteriormente con la diffusione del NB-IoT in tutto il territorio. Attraverso la tecnologia NB-IoT, infatti, sarà potenzialmente possibile connettere più di 3 miliardi e mezzo di

dispositivi. Una cifra che è oltre trenta volte superiore al numero di SIM attive attualmente in Italia. L'NB-IoT è uno standard di trasmissione dati su reti cellulari che integra le caratteristiche di una rete Low Power Wide Area (LPWA), con quelle delle reti cellulari su banda licenziata LTE 2G, 3G e 4G, consentendo: un basso consumo energetico con durata delle batterie di oltre 10 anni; la copertura di aree remote, rurali e sotterranee attraverso l'adozione di tecniche che migliorano la propagazione del segnale radioelettrico, fino a +20dBm. Il minor consumo energetico è reso possibile dai sensori intelligenti NB-IoT che utilizzano la modalità "a riposo", limitando il consumo di energia solo quando ci sono dei pacchetti dati da inviare, e ottimizzando tutti quei processi, pre e post invio dei dati, che richiedono più tempo ed energia. Questo è il motivo per cui le batterie dei sensori durano fino a dieci anni e non è necessaria alcuna fonte di alimentazione esterna.

In questo modo, grazie a una lunga durata della batteria, e a una maggiore capacità di copertura, il Narrowband-IoT permette già oggi di far comunicare tra loro oggetti che prima d'ora non potevano essere connessi, sviluppando così le potenzialità della tecnologia IoT in svariati mercati.

www.vodafone.it

Leonardo, Airbus e Dassault Aviation rinnovano il pieno sostegno al primo programma europeo per un velivolo a pilotaggio remoto di classe MALE

Il primo modello a grandezza naturale dello European Medium-Altitude Long-Endurance Remotely Piloted Aircraft (MALE RPAS - velivolo a pilotaggio remoto per operazioni a quote intermedie e a lungo raggio) è stato presentato all'ILA Air Show 2018 di Berlino, presso l'aeroporto di Schönefeld. L'evento è stato l'occasione per confermare l'impegno dei quattro Stati europei (Germania, Francia, Italia e Spagna) e

partner industriali allo sviluppo congiunto di una soluzione indipendente per la difesa e la sicurezza dell'Europa. Questa nuova fase giunge a distanza di quasi due anni dal lancio dello studio di definizione, nel settembre 2016, da parte dei quattro Paesi partecipanti e segue la Dichiarazione di Intenti sulla collaborazione congiunta ad un sistema aereo senza pilota di classe MALE europeo, firmata dalle stesse nazioni nel maggio 2015.

Il MALE RPAS diverrà parte integrante della capacità dell'Europa di garantire la propria indipendenza in futuro. Il programma è pensato per soddisfare urgenti requisiti in termini di capacità da parte delle Forze Armate europee. Questa partnership innovativa attenua anche gli effetti generati dalle restrizioni di bilancio attraverso la combinazione efficace dei fondi per ricerca e sviluppo. "I sistemi aerei senza pilota e le loro applicazioni rappresentano uno degli elementi tecnologici fondamentali per la futura evoluzione dell'industria europea della difesa", ha sottolineato Lucio Valerio Cioffi, Responsabile della Divisione Velivoli di Leonardo. "Il MALE RPAS europeo punta a promuovere lo sviluppo di alte tecnologie e contribuirà a sostenere competenze chiave e posti di lavoro in Europa, fornendo alle Forze Armate prestazioni avanzate e un sistema operativo indipendente".

www.leonardocompany.com

Il Lofar arriva in Italia

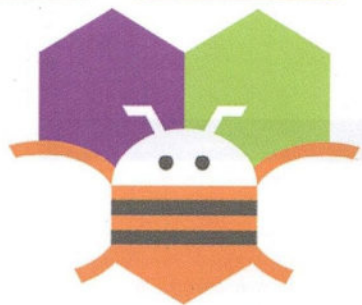
Con oltre 25 mila antenne raggruppate in 51 stazioni distribuite in 7 stati europei, Lofar (Low Frequency Array) è la più estesa rete per osservazioni radioastronomiche in bassa frequenza attualmente operativa. Il radiotelescopio europeo si estende ora anche all'Italia, con il contratto per la realizzazione di una nuova stazione presso Medicina, in provincia di Bologna. Con la firma da parte dell'Istituto nazionale di astrofisica (Inaf) del contratto per la realizzazione della nuova stazione, Lofar si avvia a diventare ancor più esteso e ad accrescere di conseguen-

za le sue capacità osservative; l'Inaf è alla guida per l'Italia di un consorzio che vedrà nel prossimo futuro il coinvolgimento di alcune università del nostro paese. Lofar è una struttura di livello mondiale per la ricerca astronomica progettata e sviluppata da Astron, l'Istituto olandese per la radioastronomia, e ora con l'Italia conta la partecipazione di otto stati in Europa. Grazie alla sua avanzata elettronica, permette di captare onde radio provenienti dalle sorgenti più prossime a noi come il Sole, per spingersi fino ai flebilissimi segnali emessi da galassie e buchi neri

lontanissimi ai confini dell'universo. René Vermeulen, direttore dell'International Lofar Telescope (Ilt), è entusiasta dell'ingresso dell'Italia: "Attraversando le Alpi, Lofar otterrà una rete di antenne meglio distribuita in tutta Europa, a beneficio di tutti gli astronomi che potranno avere a disposizione una migliore della qualità delle immagini raccolte. La nuova partnership con l'Italia rafforza i legami di vecchia data di scambio e collaborazione nell'ambito della radioastronomia in Europa".

www.inaf.it





MIT APP INVENTOR

3

di FRANCESCO FICILI

Proseguiamo lo studio e la conoscenza di App Inventor, il tool di sviluppo per applicazioni Android creato dal MIT e basato su un linguaggio di programmazione grafico. In questa terza puntata analizzeremo due interessanti famiglie di componenti, ossia Storage e Sensors.

Nella scorsa puntata di questo corso ci siamo lasciati dopo aver introdotto il concetto di componente in MIT App Inventor (i componenti sono gli elementi che permettono di aggiungere funzionalità all'app in sviluppo...) ed aver constatato e sperimentato quanto potente sia questo strumento, che ci permette sostanzialmente di creare l'interfaccia grafica della nostra app, di accedere alle varie funzionalità specifiche del nostro smartphone e di utilizzare i vari sensori che si trovano al suo interno e compongono la dotazione hardware. Abbiamo anche avuto modo di conoscere le interfacce di comunicazione presenti all'interno degli smartphone e, a contorno della definizione del

concetto di componenti, con un esempio pratico siamo riusciti a sperimentare l'utilizzo del componente notifier, che permette di notificare all'utente un messaggio al verificarsi di un certo evento.

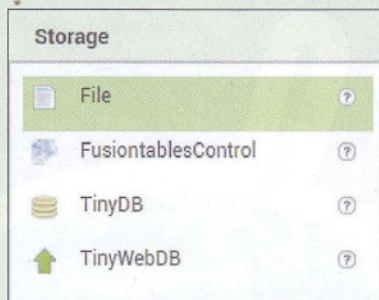


Fig. 1 - Famiglia di componenti storage.

In questa puntata riprendiamo da dove ci siamo fermati e analizziamo in dettaglio due nuove famiglie di componenti di App Inventor: la famiglia di componenti **Storage** e la famiglia di componenti **Sensors**.

Anche in questa puntata, come nelle precedenti, per brevità utilizzeremo spesso l'acronimo AI per dire App Inventor.

I componenti Storage

La famiglia di componenti Storage contiene componenti per la gestione di file e dei data-base; in essa è inoltre presente un componente per la gestione delle Google Fusion Tables, che tuttavia non tratteremo in questo corso. La Fig. 1 riporta una schermata che raggruppa la famiglia di componenti Storage nell'IDE di App Inventor.

Ciò detto, passiamo adesso a un'analisi più di dettaglio dei singoli componenti ed in particolare

di proprietà, eventi e metodi che li caratterizzano. Ricordiamo anche brevemente la classificazione dei colori per i behaviours dei componenti:

- Proprietà: verde;
- Eventi: beige;
- Metodi: viola.

File Storage

Si tratta di un componente non visibile (non-visible component) che permette di effettuare operazio-



Fig. 2 - Layout della app di esempio per l'utilizzo del componente File.

PROPRIETÀ	
SINTASSI	DESCRIZIONE
Nessuna	-
EVENTI	
SINTASSI	DESCRIZIONE
GotText(text text)	Evento che indica che il contenuto di un dato file è stato letto e fornisce il risultato della lettura tramite il parametro 'text'.
METODI	
SINTASSI	DESCRIZIONE
AppendToFile(text text, text fileName)	Appende il parametro 'text' al file puntato dal parametro 'fileName'.
Delete(text fileName)	Cancella il file puntato dal parametro 'fileName'.
ReadFrom(text fileName)	Legge il contenuto del file puntato dal parametro 'fileName'.
SaveFile(text text, text fileName)	Salva il contenuto del parametro 'text' sul file puntato dal parametro 'fileName'.

Tabella 1 - Proprietà, Eventi e Metodi del componente File.

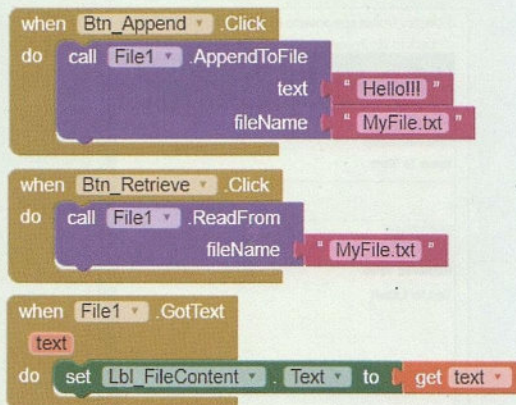


Fig. 3 - Codice grafico per l'app di esempio d'uso del componente File.

ni di salvataggio e lettura su un generico file. Per impostazione predefinita, il componente lavora sulla memoria privata dell'app in esecuzione, ma se il path di salvataggio inizia con il simbolo slash ("/"), allora il file viene creato rispetto alla directory relativa `/sdcard`. Proprietà, eventi e metodi del componente sono riportati sinteticamente in **Tabella 1**. Vediamo un semplice esempio pratico che ci faccia comprendere come funziona il componente File: creiamo un nuovo progetto in AI e aggiungiamo i seguenti componenti:

- 2 componenti Button (chiamati Btn_Append e Btn_Retrieve);
- 1 componente Label (chiamato Lbl_FileContent);
- 1 componente File.

A questo punto il layout della nostra nuova app dovrebbe somigliare a quello riportato nella Fig. 2.

Ciò che ci proponiamo di fare è semplicemente di salvare la stringa "Hello!!!" su file alla pressione del tasto "Append" e di recuperarla e stamparla sulla label alla pressione del tasto "Retrieve".

Passiamo ora alla schermata "Blocks" ed inseriamo il codice grafico riportato in Fig. 3. Come potete vedere sempre dalla Fig. 3, abbiamo utilizzato i due metodi "AppendToFile" e "ReadFrom", collegati agli eventi "Button.Click" dei pulsanti Btn_Append e Btn_Retrieve, rispettivamente per salvare e leggere dal file "MyFile.txt".

Il metodo di lettura è asincrono, nel senso che non restituisce immediatamente il testo letto, ma bisogna utilizzare l'evento asincrono "GotText" per

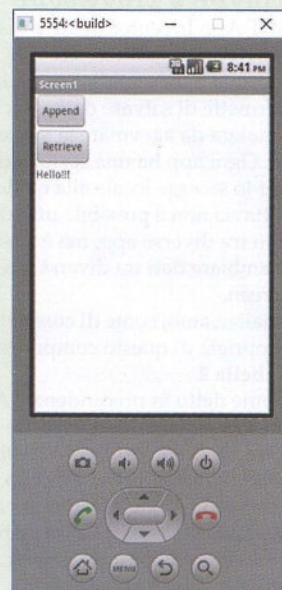


Fig. 4 - Esecuzione dell'app di esempio tramite l'emulatore.

PROPRIETÀ	
SINTASSI	DESCRIZIONE
Nessuna	-
EVENTI	
SINTASSI	DESCRIZIONE
Nessuno	-
METODI	
SINTASSI	DESCRIZIONE
ClearAll()	Cancella tutti i dati contenuti nel DB.
ClearTag(text tag)	Cancella la entry associata al tag passato come parametro.
any GetTags()	Ritorna la lista di tutti i tags contenuti nel DB.
any GetValue(text tag, any valueIfTagIsNotThere)	Ritorna il valore immagazzinato in corrispondenza del tag passato come parametro. Se il tag non è presente ritorna il valore associato al parametro "ValueIfTagIsNotThere".
StoreValue(text tag, any valueToStore)	Salva permanentemente il valore passato tramite il parametro "ValueToStore" associato tag passato tramite il relativo parametro.

Tabella 2 - Proprietà, Eventi e Metodi del componente TinyDB.

accedere ai dati recuperati dal file stesso.

Possiamo testare questa app usando un comune smartphone oppure l'emulatore: una volta avviata, premendo una volta sul tasto "Append" e la successiva sul tasto "Retrieve" si otterrà il risultato illustrato in Fig. 4.

Come avrete potuto notare da questo esempio applicativo, il componente File è molto utile per salvare informazioni in maniera non volatile ed è estremamente semplice da utilizzare, tuttavia può risultare poco efficiente quando con esso si debbano gestire grosse quantità di dati; in tali situazioni sono più efficienti i componenti database.

TinyDB e TinyWebDB

MIT App Inventor mette a disposizione due tipi differenti di gestione dei DB: TinyDB e TinyWebDB. Il primo è un semplice servizio di storage locale che permette di salvare dati associandoli a dei tag, in maniera da agevolare la successiva fase di recupero. Ogni app ha una sezione di memoria riservata per lo storage locale alla quale TinyDB si appoggia, tuttavia non è possibile utilizzarla per scambiare dati tra diverse app, ma è possibile sfruttarla per scambiare dati tra diversi screen in una app multi screen.

Analizziamo, come di consueto, eventi, metodi e proprietà di questo componente, aiutandoci con la Tabella 2.

Come detto in precedenza, AI implementa anche un componente che permette di gestire Database Remoti, chiamato TinyWebDB: si tratta di un set di servizi di comunicazione con un WebService che gestisce un database, molto vantaggioso nel caso in cui ci sia la necessità di scambiare dati tra app diverse o tra dispositivi diversi. Chiaramente occorre

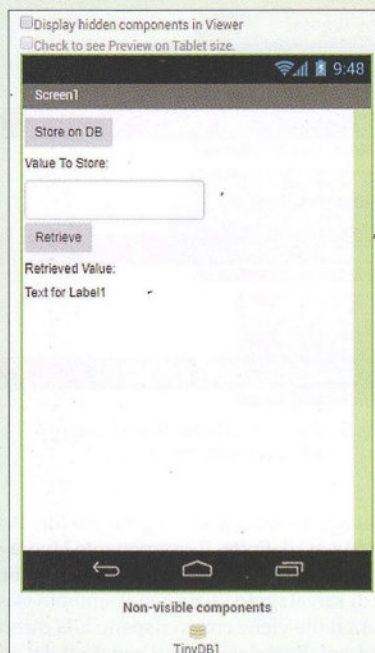


Fig. 5 - Layout dell'app relativo all'esempio di utilizzo del componente TinyDB.

anche impostare la parte lato server. Non ci dilungheremo nella trattazione di questo componente, limitandoci, in questo corso, ad evidenziarne le potenzialità, analizzandone eventi, metodi e proprietà; questi sono meglio descritti nella Tabella 3. Passiamo ora ad un semplicissimo esempio di uti-

PROPRIETÀ	
SINTASSI	DESCRIZIONE
ServiceURL	Questa proprietà serve ad impostare la URL del webservice con il quale il componente comunica per la gestione del DB
EVENTI	
SINTASSI	DESCRIZIONE
GotValue(text tagFromWebDB, any valueFromWebDB)	Questo evento informa che il valore richiesto al WebService è stato recuperato e viene passato insieme al relativo tag.
ValueStored()	Questo evento informa che il WebService ha completato una precedente richiesta di salvataggio.
WebServiceError(text message)	Questo evento indica che il WebService ha segnalato un errore.
METODI	
SINTASSI	DESCRIZIONE
GetValue(text tag)	Questo metodo serve ad inviare una richiesta di lettura del dato marcato dal tag passato come parametro al DB WebService.
StoreValue(text tag, any valueToStore)	Questo metodo serve ad inviare una richiesta di scrittura del dato marcato dal tag passato come parametro al DB WebService.

Tabella 3 - Proprietà, eventi e metodi del componente TinyWebDB.

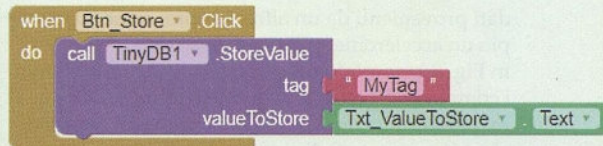
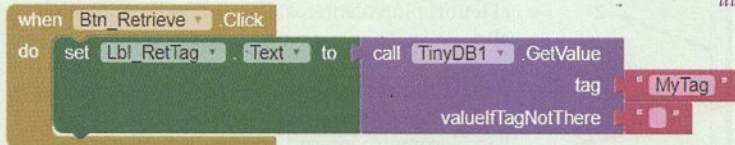


Fig. 6 - Codice grafico relativo all'esempio di utilizzo del componente TinyDB.



lizzo del componente TinyDB. Creiamo dunque una nuova app e aggiungiamo i seguenti componenti:

- 2 componenti Button (chiamati Btn_Store e Btn_Retrieve);
- 3 componenti Label (chiamati Lbl_Label1, Lbl_Label2, Lbl_ReqTag);
- 1 componente TextBox (chiamato Txt_ValueToStore);
- 1 componente TinyDB.

L'esempio è del tutto simile al precedente, ma stavolta salveremo il contenuto della textbox mar-

candolo con un tag sul data-base e lo recupereremo utilizzando il medesimo tag, per poi stamparlo sulla label.

Una volta completate le operazioni descritte in precedenza, il vostro layout dovrebbe essere del tutto simile a quello riportato nella Fig. 5.

A questo punto diamo uno sguardo al codice grafico di App Inventor, riportato nella Fig. 6. Come potete vedere, tale codice è molto semplice: alla pressione del pulsante Btn_Store viene salvato il valore del campo text del textbox, associandolo al tag "MyTag", mentre alla pressione del pulsante Btn_Retrieve viene interrogato il data-base alla ricerca del dato contrassegnato dal tag "MyTag" e il risultato viene scritto sulla label Lbl_reqTag. Il risultato ottenuto con l'emulatore è riportato nella finestra di dialogo illustrata nella Fig. 7.

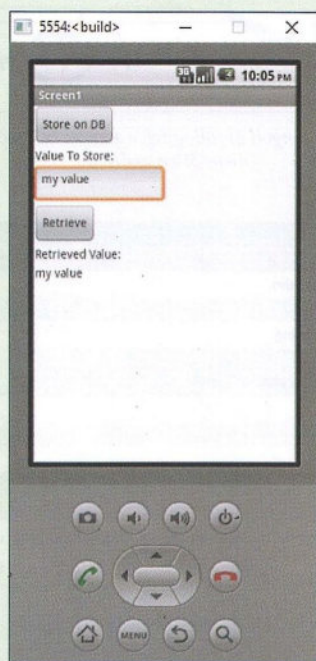


Fig. 7 - Risultato ottenuto tramite l'emulatore.

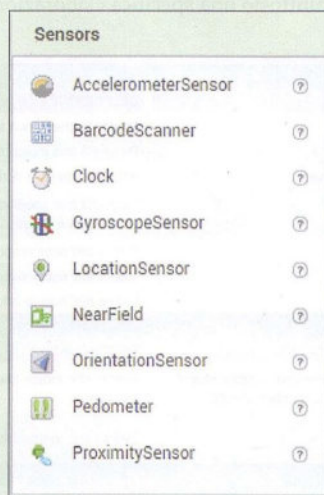


Fig. 8 - Famiglia di componenti Sensors.

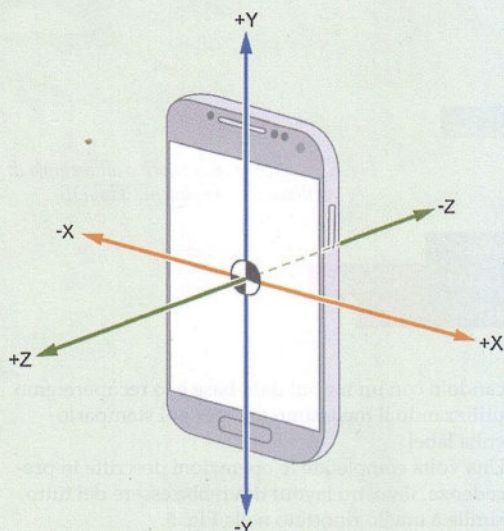


Fig. 9 - Tipica configurazione degli assi di un accelerometro, su uno smartphone Android.

I componenti Sensors

Passiamo ora all'analisi di un'altra famiglia di componenti: la famiglia di componenti Sensors. All'interno di questo gruppo si trovano componenti per la gestione dei vari sensori che tipicamente si trovano all'interno di un comune smartphone, sia fisici (come giroscopi, sensori di prossimità, GPS, ecc.), che virtuali (come ad esempio il pedometer, che in genere non rappresenta un dispositivo fisico reale, ma piuttosto una specifica elaborazione dei

dati provenienti da un altro sensore, come ad esempio un accelerometro).

In Fig. 8 è riportato uno screenshot che rappresenta i componenti presenti in questa specifica famiglia. In questa trattazione noi ci limiteremo all'analisi di alcuni componenti di questa famiglia (i componenti Accelerometer, Orientation e Pedometer), lasciando ai lettori più volenterosi l'approfondimento sugli altri.

Accelerometro

Questo componente è in grado di interfacciare l'accelerometro interno, rilevando accelerazioni in unità del SI (m/s^2) oppure condizioni di "shaking". Le accelerazioni sono rilevate sui tre assi: XAccel, YAccel, ZAccel. La Fig. 9 riporta una tipica configurazione degli assi di misura di un accelerometro in

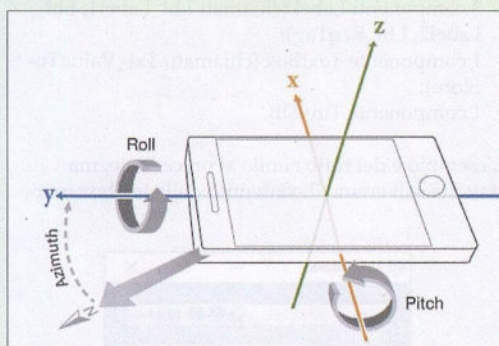


Fig. 10 - Angoli di roll, pitch e azimuth in un classico dispositivo android.

PROPRIETÀ	
SINTASSI	DESCRIZIONE
Available	Proprietà che indica se il sensore è disponibile sul dispositivo.
Enabled	Proprietà che indica se il sensore è abilitato.
MinimumInterval	Intervallo minimo di rilevamento (in ms) tra eventi di shaking.
Sensitivity	Proprietà che specifica quanto è sensibile l'accelerometro. La codifica è su tre valori: 1 = bassa, 2 = moderata, 3 = alta.
XAccel	Valore dell'accelerazione sull'asse X.
YAccel	Valore dell'accelerazione sull'asse Y.
ZAccel	Valore dell'accelerazione sull'asse Z.
EVENTI	
SINTASSI	DESCRIZIONE
AccelerationChanged(number xAccel, number yAccel, number zAccel)	Evento che indica che l'accelerazione è cambiata su uno dei tre assi e contestualmente passa i valori delle accelerazioni.
Shaking()	Evento che indica che si è verificato un evento di shaking.
METODI	
SINTASSI	DESCRIZIONE
Nessuno	

Tabella 4 - Proprietà, eventi e metodi del componente Accelerometer.

PROPRIETÀ	
SINTASSI	DESCRIZIONE
Available	Proprietà che indica se il sensore è disponibile sul dispositivo.
Enabled	Proprietà che indica se il sensore è abilitato.
Azimuth	Ritorna l'angolo di azimut del dispositivo.
Pitch	Ritorna l'angolo di pitch del dispositivo.
Roll	Ritorna l'angolo di roll del dispositivo.
Magnitude	Ritorna un numero da 0 a 1 che indica quanto il dispositivo è inclinato.
Angle	Ritorna un angolo che indica la direzione verso la quale il dispositivo è inclinato.
EVENTI	
SINTASSI	DESCRIZIONE
OrientationChanged (number azimuth, number pitch, number roll)	Evento che indica che è cambiato l'orientamento del dispositivo, e fornisce tra i parametri i tre angoli di roll, pitch ed azimut.
METODI	
SINTASSI	DESCRIZIONE
Nessuno	-

Tabella 5 - Proprietà, eventi e metodi del componente Orientation Sensor.

un dispositivo Android.

In Tabella 4 sono riportati, come di consueto, eventi, metodi e proprietà del componente.

Orientation Sensor

Il sensore di orientamento è un componente non visibile che permette di determinare l'orientamento spaziale del dispositivo. Questo componente fornisce i seguenti tre valori, in gradi:

- Rollio (Roll): valore dell'angolo di rollio espresso in gradi;

- Beccheggio (Pitch): valore dell'angolo di beccheggio in gradi;
- Azimut: valore dell'angolo di azimut in gradi; questo angolo vale 0° in direzione nord, 180° in direzione sud, 90° in posizione est e 270 gradi in posizione ovest; le informazioni su questo angolo possono essere utilizzare per implementare bussole digitali.

In Fig. 10 è riportata una rappresentazione grafica che chiarisce ulteriormente come sono misurati gli

PROPRIETÀ	
SINTASSI	DESCRIZIONE
Distance	Questa proprietà ritorna la distanza approssimativa percorsa in metri.
ElapsedTime	Ritorna il tempo trascorso in millisecondi da quando il pedometro è stato attivato.
SimpleSteps	Ritorna il numero di "Simple Step" eseguiti da quando il pedometro è stato attivato.
StopDetectionTimeout	Questa proprietà serve a settare la durata in millisecondi del timeout di idle, ossia il tempo trascorso il quale senza che vengano rilevati dei passi il componente passa in "stopped" state.
StrideLength	Setta la lunghezza del passo medio in metri.
WalkSteps	Ritorna il numero di "Walk Step" eseguiti da quando il pedometro è stato attivato. Si definisce "Walk Step" il passo tipico eseguito per muoversi in avanti.
EVENTI	
SINTASSI	DESCRIZIONE
SimpleStep(number simpleSteps, number distance)	Evento che indica l'esecuzione di un "Simple Step".
WalkStep(number walkSteps, number distance)	Evento che indica l'esecuzione di un "Walk Step".
METODI	
SINTASSI	DESCRIZIONE
Pause()	Mette il pedometro in pausa.
Reset()	Resetta tutti i contatori.
Resume()	Riavvia il pedometro.
Save()	Salva tutti i contatori del pedometro in memoria non volatile.
Start()	Avvia il pedometro.
Stop()	Arresta il pedometro.

Tabella 6 - Proprietà, eventi e metodi del componente pedometer.

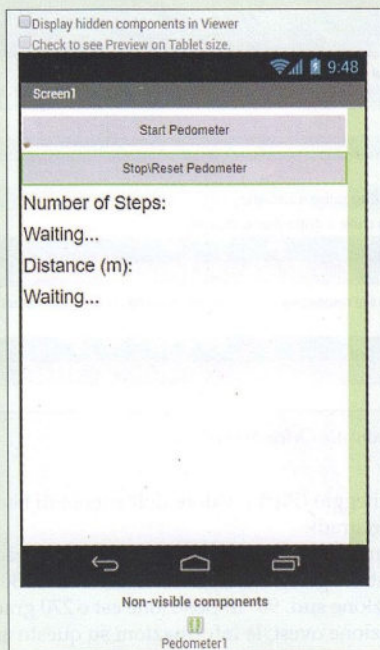
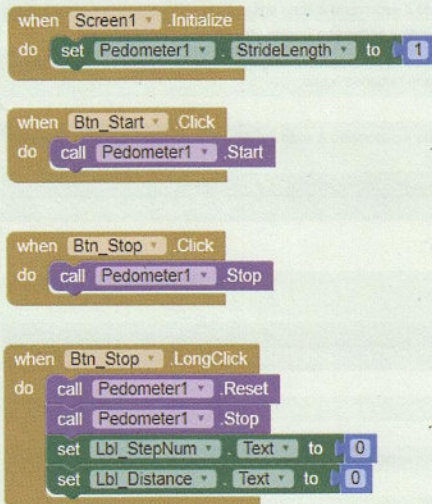


Fig. 11 - Layout dell'applicazione di esempio

angoli di roll, pitch ed azimut. Anche in questo caso esaminiamo eventi, metodi e proprietà, aiutandoci con la Tabella 5.



Pedometer

L'ultimo componente della famiglia sensor che analizziamo in questa puntata del corso è il Pedometer (o pedometro): si tratta di un sensore virtuale, che sfrutta le informazioni provenienti dall'accelerometro interno per determinare se la persona che porta con sé lo smartphone ha compiuto dei passi; ciò viene ottenuto analizzando l'accelerazione in una combinazione di direzioni che fa presupporre che si stia camminando, perché corrisponde al movimento in avanti e dal basso in alto e viceversa, il tutto contemporaneamente.

Inoltre, sempre con il componente pedometer, configurando opportunamente un parametro (`strideLength`) è possibile anche stimare la distanza percorsa.

Esaminiamo anche in questo caso eventi, metodi e proprietà, aiutandoci con la Tabella 6, che li raccoglie e li descrive.

Esempio pratico sull'uso dei componenti Sensors

Passiamo a questo punto al nostro consueto esempio pratico, utile a sperimentare i concetti che abbiamo esposto. Per l'esempio abbiamo deciso di utilizzare l'ultimo componente descritto (ossia il Pedometer) per realizzare una semplice applicazione contapassi, con in più una misura approssimativa della distanza percorsa in metri. Per realizzare questa applicazione posizioniamo i seguenti elementi grafici:

- 2 Buttons (Btn_Start e Btn_Stop);

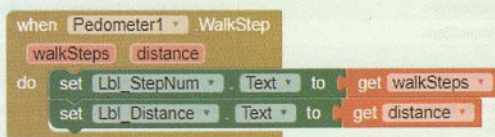


Fig. 12 - Codice grafico dell'applicazione di esempio

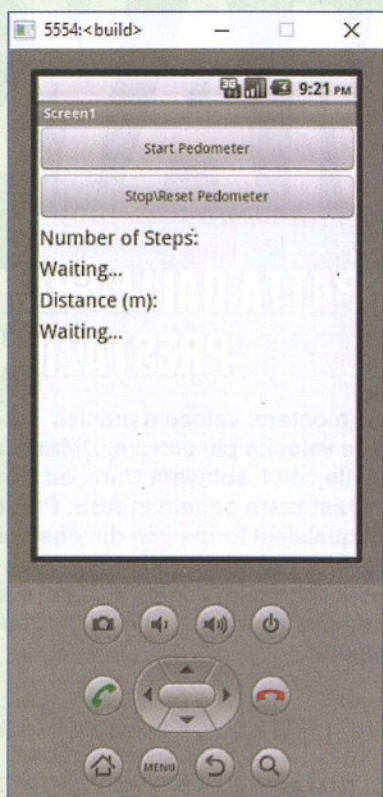


Fig. 13 - Screenshot dell'applicazione di esempio in emulazione.

- 4 Labels (Lbl_Label1, Lbl_Label2, Lbl_StepNum, Lbl_Distance);
- 1 componente pedometer.

A questo punto organizziamo il layout come proposto in Fig. 11 e passiamo sulla Block view.

La realizzazione dell'app, sfruttando le potenzialità del componente pedometer, risulta piuttosto semplice; infatti sfruttiamo gli eventi `button.click` sui pulsanti `Btn_Start` e `Btn_Stop` rispettivamente per avviare e arrestare il contapassi (invochando i metodi `Start()` e `Stop()`) ed inoltre utilizziamo l'evento `Button.LongClick` per resettare i contatori (chiamando in sequenza i metodi `Reset()` e `Stop()` e resettando le label così da riportare il valore 0).

Poi sfruttiamo l'evento `Pedometer.WalkStep` per contare i passi, settando ad ogni occorrenza dell'evento stesso i valori delle due label `Lbl_StepNum` ed

`Lbl_Distance`, pari ai valori dei parametri `WalkSteps` e `Distance` che possiamo recuperare tramite gli appositi metodi di `get`.

Il codice grafico che implementa quanto spiegato in questo esempio pratico è riportato nella Fig. 12. Invece nella Fig. 13 trovate uno screenshot dell'applicazione che gira in emulazione sull'emulatore, mentre nella Fig. 14 riportiamo un ulteriore screenshot ma stavolta catturato da un dispositivo reale, durante il funzionamento con l'app in esecuzione.

Conclusioni

In questa terza puntata abbiamo iniziato ad analizzare più in dettaglio una delle caratteristiche più interessanti di MIT Appinventor, ossia i componenti, concentrandoci in particolare sulle famiglie `Storage` e `Sensors`.

Nelle prossime puntate continueremo su questa linea, studiando nuovi componenti (l'elenco completo dei componenti di App Inventor lo trovate nella seconda puntata del corso) passando anche in rassegna componenti che permettono di interagire con dell'hardware esterno allo smartphone.

Quindi vi diamo appuntamento al mese prossimo! ■

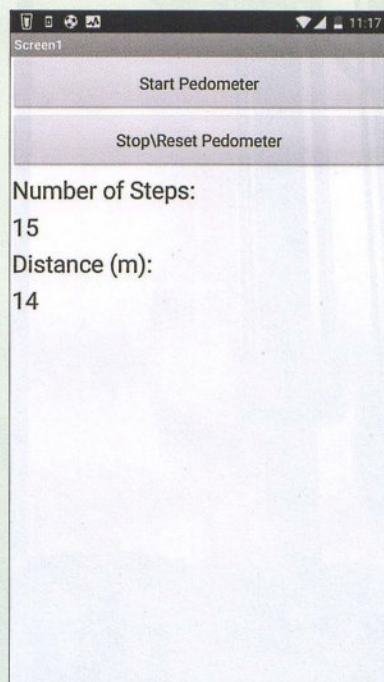


Fig. 14 - Screenshot dell'applicazione di esempio su un dispositivo reale.

NUOVA STAMPANTE 3D IN KIT

DMAKE

COMPATTA DALLE GRANDI
PRESTAZIONI!

Facile da montare, veloce e precisa anche alle velocità più elevate. DMAKE è compatibile con il software Cura, ed è in grado di realizzare oggetti in ABS, PLA o Nylon di qualsiasi forma con dimensioni massime di Ø150 x 300 mm.

Grazie alla scheda di controllo di cui dispone è in grado di stampare anche senza l'ausilio del computer.

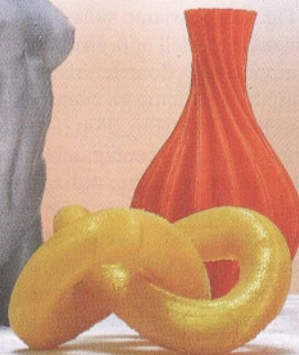
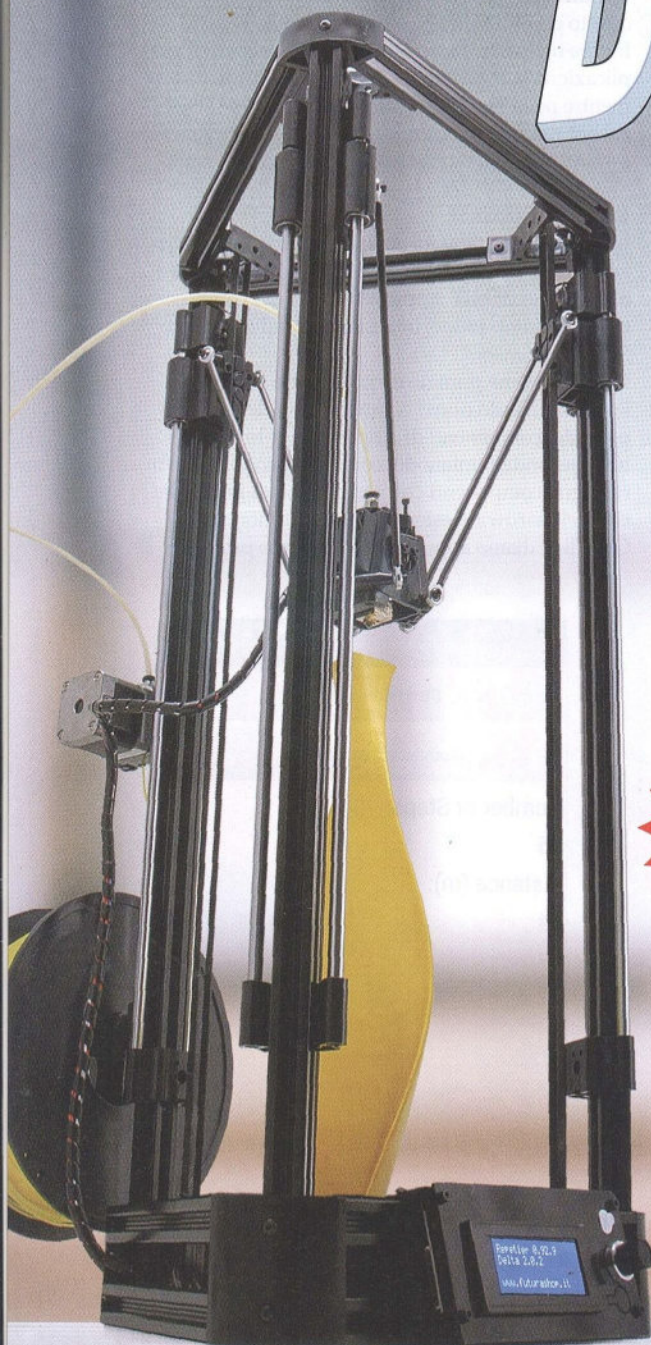
*Istruzioni
in ITALIANO!*

cod. DMAKE

NUOVO PREZZO

€ 449,00

- › Struttura in alluminio con elementi di raccordo in materiale plastico.
- › Estrusore a caldo con ugello da 0,4 mm.
- › Ventola per raffreddamento integrata nella testina di stampa.
- › Movimentazione con 3 motori stepper NEMA 17.
- › Display LCD 20 x 4 caratteri.
- › Interfaccia USB / SDCard.
- › Velocità di stampa 20-100 mm/s.
- › Precisione di stampa 0,005 mm.
- › Dimensioni 690 x 340 x 340 mm.
- › Peso 8 kg.
- › Alimentatore incluso.



Prezzi IVA inclusa.

**FUTURA
ELETTRONICA®**

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

Caratteristiche tecniche
e vendita on-line su:

www.futurashop.it

Il Pentagono e il Dipartimento della Difesa americano hanno deciso di spostare sul cloud la loro tecnologia informatica per difendersi da eventuali cyber attacchi e sfruttare le nuove capacità offerte dal machine learning e dall'intelligenza artificiale.

L'attuale infrastruttura IT è un ambiente legacy federato basato su una tecnologia che sta invecchiando e la transizione verso il cloud migliorerà la sicurezza, l'accessibilità dei dati, l'accessibilità e le prestazioni sia per il warfighter che per le attività di tutto il Dipartimento.

L'uso dell'intelligenza artificiale rappresenterà un aspetto decisivo anche in campo militare e l'idea del Dipartimento della Difesa nasce dalla preoccupazione che sia la Cina, sia la Russia stanno per sviluppare capacità in questo settore; inoltre i cyber attacchi cedono sempre

Intelligenza artificiale al Pentagono



più il passo a nuove forme di guerra ibrida che alcuni stati potenzialmente avversari starebbero mettendo a punto.

Il cloud computing è l'uso di server uniti per fornire virtual computing, storage e servizi aggiuntivi, oltre che lo sfruttamento dei big data. Con il programma JEDI Cloud, il Dipartimento della Difesa persegue l'infrastruttura e la piattaforma come strato fondamentale necessario per abilitare funzionalità avanzate come il machine learning e l'intelligenza artificiale. Il progetto è attualmente nella fase di ricerca di una

piattaforma cloud e si vociferava dell'interessamento per la piattaforma di intelligenza artificiale di Google.

www.defense.gov

➤ Tech Data entra nella IQRF alliance per l'IoT

Tra i link wireless che si contendono la scena dell'IoT e dell'Industrial IoT c'è l'IQRF, che è una piattaforma per la connettività RF a bassa potenza capace di consentire connessioni a distanze da decine a centinaia di metri punto a punto e fino a diversi chilometri in reti utilizzate per la telemetria. Il ridotto data-rate (20 kb/s) rende l'IQRF idoneo al controllo industriale e all'automazione di edifici e città (luci, parcheggi ecc.). Gli sviluppatori e le aziende produttrici sono riuniti nel consorzio IQRF Alliance, nel quale recentemente è entrato a far parte Tech Data, primo distributore di prodotti informatici a entrare nell'alleanza. Il protocollo IQRF contempla la creazione di un ecosistema formato da dispositivi inte-



roperabili funzionanti sulle bande di frequenza 868 MHz, 916 MHz o 433 MHz, comprendenti gateway, cloud, app mobili e software; proprio qui entra in gioco Tech Data, la cui partnership con IQRF punta a fornire soluzioni software per IoT.

www.techdata.it

► Le borse connesse di Piquadro

L'azienda distintasi in questi anni per le ottime borse per computer portatili, ha deciso di passare da custodie per contenere dispositivi elettronici a borse che integrano elettronica; così è nata la serie di trolley e zaini arricchiti da una "bag automation" che sono in grado di comunicare con chi li possiede via Bluetooth e attraverso smartphone e/o smartwatch, grazie all'app Connequ, disponibile su App Store e Play Store, sia per smartphone sia per smartwatch.

Tramite questa app si gestiscono tutte le funzioni

disponibili nel prodotto, tra cui l'allarme in caso di allontanamento (che previene la perdita e il furto) e la ricarica contactless di smartphone e tablet; grazie a sensori integrati nella maniglia, borse, trolley e zaini possono rilevare il peso del contenuto e comunicarlo alla app (funzione utile in aeroporto) nonché comandare il lucchetto TSA bluetooth dietro riconoscimento dell'impronta digitale da parte di uno smartphone o tablet, accendere i LED integrati negli zaini, sempre dall'app.

I nuovi Zaini e trolley Piquadro integrano un power-

► LoraWAN incontra le Smart City

Kerlink, azienda fornitrice di soluzioni end-to-end per l'Internet of Things e la geolocalizzazione, e Telemar, tra i primi operatori mobile in Italia ad aver realizzato una rete LoRaWAN per applicazioni dell'IoT, hanno avviato un progetto in prova (P.O.C - Proof of Concept) per monitorare l'aria e la temperatura in alcuni edifici pubblici del comune di Piovene Rocchette (Vicenza). Il progetto prevede l'utilizzo di una rete di sensori e termostati in

ogni ambiente, che generano dati e informazioni sulla temperatura dell'aria allo scopo di gestire in maniera ottimale la climatizzazione, nell'ottica di ridurre i consumi energetici e costruire modelli di utilizzo. La rete LoRaWAN installata nella città vicentina comprende i gateway WirnetTM iBTS Standard per l'esterno e WirnetTM iFemto-Cell per l'interno (tutti e due realizzati da Kerlink), 17 sensori di temperatura e 14 attuatori che

controllano la climatizzazione. Gli utenti possono controllare la temperatura di ogni stanza, e hanno la possibilità di accendere o spegnere gli impianti anche da remoto, direttamente tramite smartphone; grazie a un'app sviluppata appositamente per il progetto.

Gli edifici coinvolti nel progetto di Kerlink e Telemar comprendono una scuola dell'infanzia, una scuola primaria e secondaria ed una biblioteca.

I dati rilevati riguarderanno le condizioni di clima, temperatura, umidità, consumi elettrici negli edifici osservati e altri parametri utili a favorire la gestione ottimale.

Questo è uno dei tanti esempi di come l'IoT può essere d'aiuto nel settore civile, concorrendo a rendere realtà il concetto di Smart City.

Il progetto non sarà fine a se stesso ma sarà l'esperienza pilota per collaudare l'automazione tramite IoT ed estenderla, in futuro, a tutta la regione Veneto.

www.kerlink.com



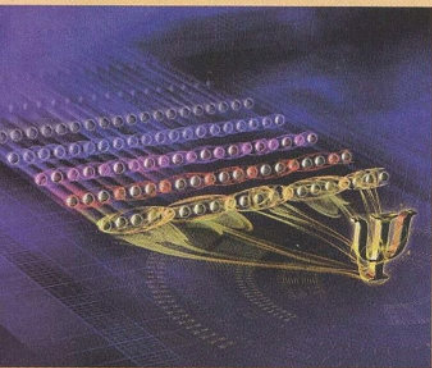
bank per ricaricare gli smartphone e comunicano all'apparato mobile (tablet o smartphone abbinati) con un allarme da parte dell'app, quando il livello di carica della batteria è basso, nonché di un modulo GSM che permette di rintracciarli in caso di furto o smarrimento.

Piquadro realizza anche accessori collegabili all'app, come i portachiavi antimarrimento connessi che in caso di allontanamento eccessivo dal tablet o smartphone avvertono il proprietario con un allarme. Oltre a dare l'allarme, l'app mostra su una mappa dove ha localizzato l'oggetto.

www.piquadro.com



» Il quantum computer vicino alla realtà



Sebbene i ricercatori siano riusciti a creare qubit che sopravvivano abbastanza a lungo per prendere parte ai calcoli, ciò che ha ostacolato la realizzazione dei quantum computer è stata l'impossibilità di intrappolarli in modo da formare registri quantici abbastanza grandi da mantenere i dati all'interno di un processore durante un cal-

colo. Ora una ricerca condotta da ricercatori dell'Institute for Quantum Optics and Quantum Information hanno fatto un passo avanti nel superare questo limite, grazie alla realizzazione di un registro quantistico di 20 qubit che può memorizzare più di un milione di stati quantici.

Per creare singoli qubit, i ricercatori hanno intrappolato 20 ioni di calcio in una trappola elettrostatica dove un campo elettrico allinea gli ioni e un campo elettrico laterale li mette insieme, in modo da occupare posizioni distanti 5 micrometri. Gli ioni di calcio hanno un elettrone di valenza il cui spin può occupare due stati corrispondenti agli stati quantici 0 o 1 del qubit.

Quando due qubit in sovrapposizione sono collegati possono memorizzare tutte le possibili combinazioni degli stati quantistici dei qubit, risultanti in quat-

tro valori. L'aggiunta di un altro qubit alla coppia raddoppierà il numero di combinazioni e quindi i valori che potranno essere memorizzati e così via.

Dopo 20 di questi raddoppi, 20 qubit intrappolati possono memorizzare 2^{20} , ossia 1.048.576 valori.

A differenza di un computer classico, che fa i calcoli eseguendo un gran numero di passaggi sequenziali dettati dal programma, il registro di qubit riceve l'intera istruzione per un calcolo in un colpo solo, e restituisce il risultato quasi istantaneamente, in un singolo processo. Pertanto, il registro quantico deve contenere qubit sufficienti (almeno alcune migliaia) per eseguire le istruzioni necessarie al calcolo. Il passo successivo è ripetere l'esperimento con una fila di 50 ioni.

<https://iqoqi.at/en>

Sentinel-3B, in orbita un altro satellite per il monitoraggio ambientale

Con il lancio e la messa in orbita del satellite Sentinel-3B si completa la prima serie di missioni Sentinel per la rete del programma di monitoraggio ambientale dell'Unione Europea Copernicus, il più grande programma di osservazione della Terra, con una completa gamma di tecnologie per monitorare gli oceani, la terraferma e l'atmosfera della Terra.

Sentinel-3B è stato lanciato ieri, unendosi al suo gemello in orbita Sentinel-3A. Il satellite, del peso di 1.150 kg, è stato messo in orbita da un lanciatore Rockot partito da Plesetsk, Russia, alle 19:57 (17:57 GMT, 21:57 ora locale di Plesetsk) del 25 aprile.

Lo stadio superiore del razzo Rockot ha portato Sentinel-3B nell'orbita stabilita e appena 92 minuti dopo il decollo, Sentinel-3B ha inviato i primi segnali alla stazione di terra di Kiruna, in Svezia. I collegamenti dati sono stati prontamente stabiliti dalla squadra ESA del centro operativo di Darmstadt, Germania, consentendo di assumere il controllo del satellite.

Durante i tre giorni di lancio e di fase iniziale dell'orbita, gli operatori controlleranno che tutti i sistemi del sa-

tellite funzionino, e cominceranno a calibrare gli strumenti per la fase di messa in servizio del satellite. La missione dovrebbe cominciare le operazioni di routine entro cinque mesi.

Il programma Copernicus si basa sui satelliti Sentinel per fornire dati per il monitoraggio ambientale ed a sostegno delle attività di sicurezza civile. Sentinel-3B ha a bordo una serie di sensori all'avanguardia.

Sopra gli oceani, misura la temperatura, il colore e l'altezza della superficie del mare, così come lo spessore del ghiaccio marino. Queste misurazioni sono utilizzate, per esempio, per monitorare i cambiamenti del clima terrestre e per altre applicazioni pratiche, come l'inquinamento marino.

Sulla terraferma, questa innovativa missione monitora gli incendi boschivi, traccia il modo in cui il terreno viene utilizzato, controlla la salute della vegetazione e misura l'altezza di fiumi e laghi.

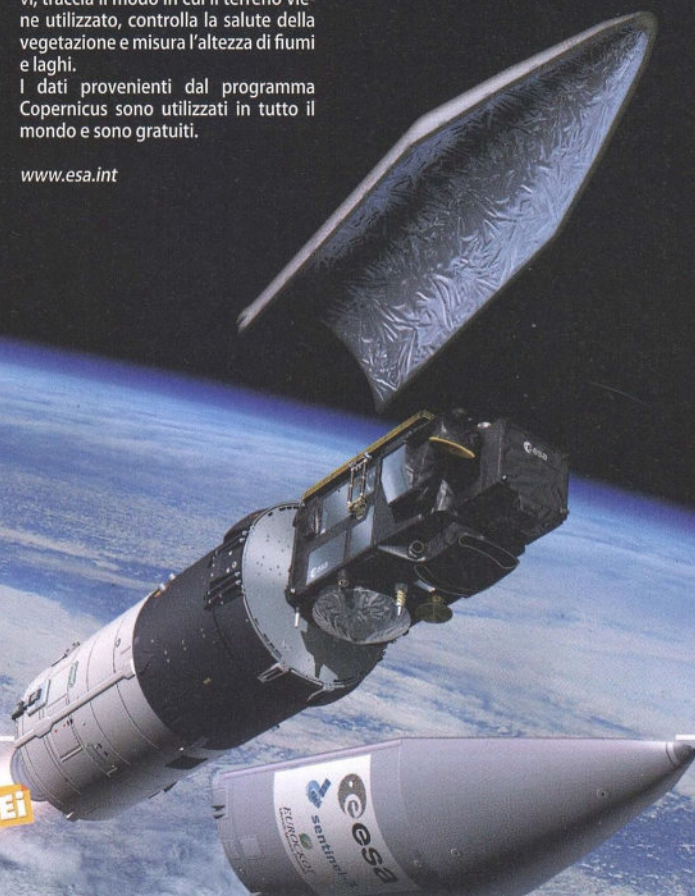
I dati provenienti dal programma Copernicus sono utilizzati in tutto il mondo e sono gratuiti.

www.esa.int

Un collettore solare gonfiabile per generare energia dove serve

Il progetto HELIOtube ha scartato la tecnologia CPS convenzionale, che richiede riflettori parabolici in vetro a specchio pesanti e costosi, e si è focalizzato sullo sviluppo di un'alternativa economica e leggera che utilizza cilindri gonfiabili di grandi dimensioni per concentrare i raggi solari. I cilindri vengono mantenuti in posizione grazie ad anelli metallici che fanno parte di un'architettura modulare e contribuiscono alla rotazione del tubo per seguire la posizione del sole.

È stato utilizzato un materiale altamente duraturo, leggero e ricicla-



bile per creare il cilindro, che è composto da tre pellicole di plastica: una di base, una trasparente e una a specchio. La pellicola a specchio separa il concentratore in due camere ermetiche, che si estendono per tutta la lunghezza del tubo. Piccole differenze di pressione tra la camera superiore e quella inferiore flettono verso il basso la pellicola a specchio, in modo da creare un canale a specchio che concentra fino a 100 volte i raggi solari e riscalda il fluido termico che scorre all'interno del ricevitore nella camera superiore sino a temperature che raggiungono i 400 °C. I collettori HELIOtube producono circa un megawatt di energia termica ciascuno e possono essere collegati in modo da formare una centrale termoelettrica solare.

<https://heliolis.com>



Enel installa il suo primo sistema di accumulo energetico in Canada

Enel allarga la propria esperienza nei sistemi di accumulo in Canada grazie alla firma di un accordo con Algoma Orchards, società dell'Ontario specializzata nella produzione, trasformazione e commercializzazione di mele, per l'installazione e gestione di un sistema behind-the-meter a batterie a ioni di litio da 1 MWh. Il software di Enel DEN.OS1 ottimizzerà l'uso delle batterie nelle ore di picco consentendo ad Algoma Orchards di rafforzare la sua partecipazione al programma di gestione della domanda dell'Ontario. L'impianto dovrebbe entrare in servizio entro il primo semestre del 2018 e assicurare a Algoma Orchards significativi risparmi, distribuiti su un periodo contrattuale di 11 anni. "Questo accordo dimostra l'importanza dello sviluppo di sistemi energetici flessibili, come quelli

di accumulo, per consentire risparmi ai clienti business e migliorare la sostenibilità e affidabilità della rete elettrica per tutti i consumatori", ha commentato Francesco Venturini, responsabile di Enel X. "Il continuo calo dei costi delle batterie ci permetterà di sviluppare molti altri impianti come quello di Algoma Orchards, che valorizza l'eccellenza del nostro software DEN.OS1 e consente risparmi ai clienti attraverso offerte commerciali personalizzate". L'accordo prevede che Enel acquisterà, installerà e gestirà la batteria per conto di Algoma Orchards, e condividerà i risparmi consentiti dal software di ottimizzazione DEN.OS1 che controlla il sistema di accumulo.

www.enel.com

enel

ABB lancia il caricabatterie per veicoli elettrici più veloce al mondo: 200 km in 8 minuti

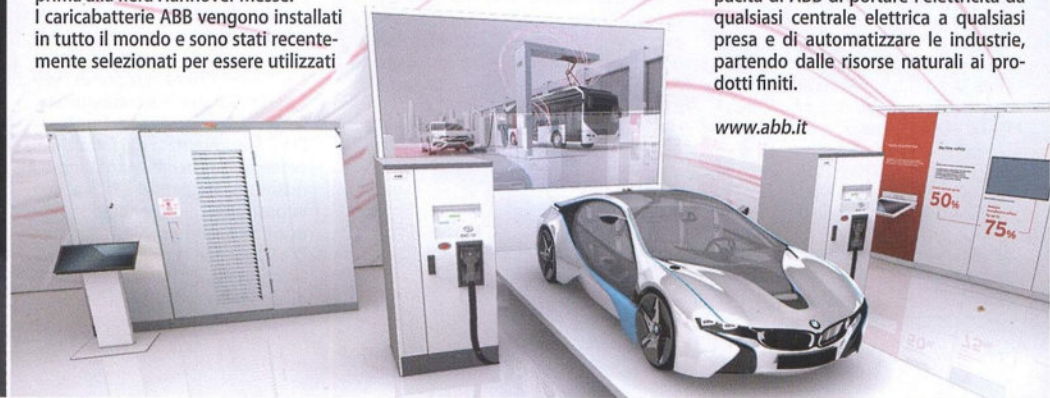
Operando con potenze fino a 350 kilowatt, la nuova colonnina di ricarica di ABB, Terra High Power, aggiunge fino a 200 chilometri di autonomia a un veicolo elettrico in soli 8 minuti. I nuovi caricatori sono ideali per l'uso nei punti di ristoro autostradali e nelle stazioni di servizio e sono stati presentati in anteprima alla fiera Hannover Messe. I caricabatterie ABB vengono installati in tutto il mondo e sono stati recentemente selezionati per essere utilizzati

da Electrify America, il più grande progetto di infrastruttura per veicoli elettrici mai realizzato negli Stati Uniti. Con oltre 6.500 stazioni di ricarica rapida in corrente continua installate in 60 paesi, ABB è leader mondiale in questo settore. Il portafoglio di prodotti ABB, che comprende tecnologie di ricarica

per auto elettriche, autobus e camion, nonché soluzioni per l'elettrificazione di navi, ferrovie e funivie, pone saldamente ABB come un campione globale nell'e-mobility.

Ulteriori tecnologie all'avanguardia nelle utility, nell'industria, nei trasporti e nelle infrastrutture, evidenziano la capacità di ABB di portare l'elettricità da qualsiasi centrale elettrica a qualsiasi presa e di automatizzare le industrie, partendo dalle risorse naturali ai prodotti finiti.

www.abb.it



Diesel: emissioni nocive ridotte a un decimo grazie a Bosch



"Il diesel ha un futuro. Oggi vogliamo archiviare definitivamente il dibattito sulla fine del diesel". Con queste parole Volkmar Denner, CEO del Board of Management di Bosch, ha annunciato una svolta decisiva nell'ambito della tecnologia diesel in occasione della presentazione dei dati di bilancio dell'azienda.

Nei mesi scorsi, infatti, gli ingegneri di Bosch sono riusciti a ridurre le emissioni di NOx ad appena 13 milligrammi per chilometro su percorsi conformi alle prescrizioni RDE, un valore dieci volte inferiore rispetto al limite di 120 mg che entrerà in vigore dal 2020. Un valore che può essere paragonato a quello delle autovetture

completamente elettriche.

Questo annuncio apre nuovi scenari sul futuro dell'automobile dal momento che il costo di una vettura diesel è mediamente pari ad un terzo rispetto a quello di un modello elettrico: che senso ha spendere tanto di più quando, con la tecnologia Bosch, l'inquinamento complessivo può essere considerato simile?

L'altra cosa sorprendente è che gli ingegneri hanno migliorato le tecnologie già esistenti senza fare ricorso a componenti aggiuntivi, che farebbero lievitare i costi. A rendere possibile il raggiungimento di tali valori è stata la combinazione di una avanzata tecnologia di iniezione con un sistema per l'aria di nuova generazione e un sistema intelligente di gestione della temperatura. Ora le emissioni di ossidi di azoto possono essere mantenute al di sotto dei valori limite in qualsiasi condizione di guida, indipendentemente dallo stile di guida, dalle condizioni atmosferiche e dalle situazioni di traffico.

www.bosch.com

Energia eolica offshore a basso costo

Un progetto finanziato dall'UE ha introdotto nuove strategie e strumenti all'avanguardia al fine di ridurre i costi per gli sviluppi di parchi eolici offshore e per incrementare la loro competitività.

L'eolico offshore è una componente energetica importante che potrebbe contribuire a garantire gli impegni dell'Europa riguardanti cambiamenti climatici, sicurezza energetica ed economia a basse emissioni di carbonio. Integrazione logistica e ottimizzazioni nell'ambito di catena di fornitura, installazione, funzionamento e manutenzione molto probabilmente aiuteranno a raggiungere gli obiettivi relativi alla riduzione dei costi. Il fine principale del progetto LEANWIND, era quello di sviluppare tecnologie e strategie per aiutare a risolvere i problemi

attuali e futuri dell'industria e produrre strumenti integrati per il costo dell'intero ciclo di vita e per l'analisi logistica.

Il consorzio guidato dalla University College Cork si estende in 11 paesi differenti e riunisce esperti provenienti da molteplici settori tra cui il settore industriale petrolifero e del gas, quello marittimo, quello delle spedizioni via mare e quello eolico offshore, con rappresentanti in tutta la catena logistica inclusi sviluppatori, servizi pubblici, fornitori di turbine, proprietari di imbarcazioni, cantieri navali, società di classificazione e docenti.

www.leanwind.eu

Nuova tecnologia di produzione di idrogeno pulito



L'idrogeno ha il potenziale per essere un vettore energetico pulito e sostenibile per la nostra società, in quanto le celle a combustibile che lo utilizzano forniscono un meccanismo efficiente e pulito per la conversione dell'energia. Tuttavia, l'idrogeno viene attualmente prodotto all'interno di grandi impianti centralizzati che impiegano combustibili fossili, per cui non può essere considerato una fonte di energia pulita.

RGH2 OSOD system combina in una sola unità un generatore di idrogeno e un dispositivo di stoccaggio, consentendo la fornitura locale dell'idrogeno. Il dispositivo utilizza idrocarburi quali il biogas, la biomassa o il gas naturale, che vengono riscaldati e miscelati con vapore al fine di generare idrogeno di purezza estremamente elevata, senza la necessità di ulteriori processi di purificazione.

L'idrogeno viene stoccato in modo sicuro sotto forma di materiale non gassoso e, su richiesta, viene impiegato come combustibile nelle stazioni di rifornimento di idrogeno o utilizzato per produrre calore ed energia. Caratteristiche importanti di OSOD system sono la sua scalabilità e configurabilità, che gli permettono di soddisfare qualsiasi requisito (da bassi volumi a unità di grandi dimensioni) per adattarsi a esigenze specifiche e lo rendono idoneo per processi industriali e per l'alimentazione dei veicoli elettrici a celle a combustibile. Un'ulteriore importante funzionalità è rappresentata dal fatto che il sistema entra in modalità stand-by quando la domanda è bassa, con la possibilità di riprendere la produzione quando richiesto. Inoltre, OSOD elimina la costosa e rischiosa necessità di trasportare l'idrogeno dagli impianti di produzione centralizzata ai consumatori.

<http://rgh2.com>

KIT RETROPIE e MINIRETROPIE con Raspberry Pi 3 !



KIT RETROPIE PER VIDEOGIOCHI ARCADE



Costruisci una postazione per videogiochi Arcade come quelle degli anni '80 e '90! Grazie a questo kit potrai emulare vari tipi di console: Nintendo, Game Boy, MAME, Sega Master System, Amiga, Commodore...

€ 115,00

Cod. **RETROPIEKIT_N**



Attenzione: il cabinet non è compreso, ma i disegni per realizzarlo sono scaricabili dalla scheda online del prodotto.

KIT MINI RETROPIE PER VIDEOGIOCHI ARCADE



Il kit permette di realizzare un mini cabinet per videogiochi Arcade degli anni '80 e '90. Consente di emulare vari tipi di console: Nintendo, Game Boy, MAME, Sega Master System, Amiga, Commodore...
Dimensioni contenute: 250x250x250 mm.

€ 184,00

Cod. **MINIRETROPIE**



Attenzione: il case mini cabinet realizzato con stampante 3D è acquistabile separatamente (9 colori disponibili). Cod. CASEMINICABINET • € 189,00.

Prezzi IVA inclusa

**FUTURA
ELETTRONICA®**

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

Caratteristiche tecniche
e vendita on-line su:

www.futurashop.it

BBC micro:bit

DAI VITA ALLE TUE IDEE DIGITALI!

Micro:bit è una piccola scheda di sviluppo che integra un microcomputer a 32 bit; può essere utilizzata per imparare le basi della programmazione ed è adatta a tutte le età. Dispone di sensori di movimento, una bussola elettronica, un buon numero di I/O, alcuni pulsanti e numerosi LED disposti a matrice per lo sviluppo di interfacce grafiche. Grazie al bluetooth integrato è possibile programmarla facilmente dal web, da PC, tablet e smartphone.

Utilizzala subito con
il prototyping kit!

Incluso nel kit
IL NUOVO MANUALE
a colori che spiega nel dettaglio
come utilizzare e programmare micro:bit



Utilizzala subito con il nostro prototyping kit



Il kit comprende: la scheda Micro:bit, una scheda con connettore per Micro:bit, un cavo USB-microUSB, un portabatterie per 2 batterie da 1,5 volt tipo AA, due batterie da 1,5 volt tipo AA, una breadboard a 400 contatti, 1 LED RGB, un Buzzer da 5 volt, jumper, mini pulsante, 3 LED (1rosso, 1verde, 1giallo), resistenze di diversi valori e una fotoresistenza.

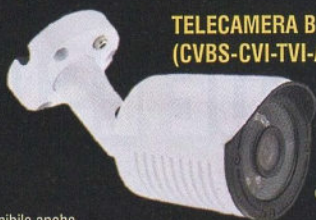
Cod: MICROBITKIT

€ 39,90

La linea Start comprende le telecamere "entry level", più economiche rispetto a quelle appartenenti alla linea STARVIS, comunque con un buon standard qualitativo. Integrano molte funzionalità presenti nelle telecamere di fascia superiore, ad un costo ridotto.

TELECAMERE **START**

le "entry-level" di qualità superiore!



TELECAMERA BULLET 4 IN 1 (CVBS-CVI-TVI-AHD) DA 2 MPX

€ 29,00
cod. **CP678FIX**

Disponibile anche varifocal 2,8-12 mm.



Cod. **CP678VAR**
€ 49,00

- Sensore di immagine: F22 CMOS colori
- Risoluzione immagine: 1080P
- Pixel effettivi: 2 MPX 1920(H)*1080(V)
- Obiettivo: 3,6 mm
- Illuminatore IR: 14μ x 18 Pz
- Portata Illuminatore IR: 20 metri
- Grado di protezione: IP66
- Sistema TV: PAL/NTSC
- Alimentazione: 12 Vdc - 550 mA
- Dimensione: 151x64x65,5 mm
- Temperatura di lavoro: da -10°C a +50°C
- Peso: circa 700 grammi

TELECAMERA DOME 4 IN 1 (CVBS-CVI-TVI-AHD) DA 2 MPX



€ 29,00
cod. **CP679FIX**

- Sensore di immagine: 1/2.7" CMOS F22
- Risoluzione immagine: 1080P
- Pixel effettivi: 2 MPX 1920(H)*1080(V)
- Obiettivo: 3,6 mm
- Illuminatore IR: 14μ x 18 Pz
- Portata Illuminatore IR: 20 metri
- Grado di protezione: IP66
- Sistema TV: PAL/NTSC
- Alimentazione: 12 Vdc - 550 mA
- Dimensione: Ø131x93 mm
- Temperatura di lavoro: da -10°C a +50°C
- Peso: circa 450 grammi

Disponibile anche varifocal 2,8-12 mm con 24 LED IR.

Cod. **CP679VAR**
€ 58,00



TELECAMERE **STARVIS**

vedono quello che le altre non riescono a vedere!



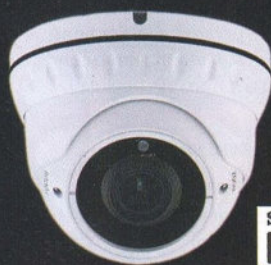
€ 126,00
cod. **CP682**



TELECAMERA BULLET STARVIS 4 IN 1 (AHD-CVI-TVI-CVBS 5X AF) DA 2 MPX

- Sensore di immagine: STARVIS SONY IMX290
- Risoluzione immagine: 1080P
- Pixel effettivi: 2 MPX 1920(H)*1080(V)
- Obiettivo: Varifocal 2,7-13,5 mm
- Messa a fuoco: automatica
- Illuminatore IR: 42μ x 4 Pz

- Portata Illuminatore IR: 40 metri
- Grado di protezione: IP66
- Sistema TV: PAL/NTSC
- Alimentazione: 12 Vdc - 700 mA
- Dimensione: 220 x 80 x 80 mm
- Temperatura di lavoro: da -10°C a +50°C
- Peso: circa 1400 grammi



TELECAMERA DOME STARVIS 4 IN 1 (AHD-CVI-TVI-CVBS 5X AF) DA 2 MPX

€ 98,00
cod. **CP683**



- Sensore di immagine: STARVIS SONY IMX290
- Risoluzione immagine: 1080P
- Pixel effettivi: 2 MPX 1920(H)*1080(V)
- Obiettivo: Varifocal 2,7-13,5 mm
- Messa a fuoco: automatica
- Illuminatore IR: 14μ x 24 Pz

- Portata Illuminatore IR: 30 metri
- Grado di protezione: IP66
- Sistema TV: PAL/NTSC
- Alimentazione: 12 Vdc - 800 mA
- Dimensione: Ø131x93 mm
- Temperatura di lavoro: da -10°C a +50°C
- Peso: circa 800 grammi

Sony Starvis è il nuovo sensore d'immagine CMOS, retroilluminato ad altissima sensibilità in grado di riprendere in condizioni di luminosità proibitive. Pensato per applicazioni industriali e videosorveglianza, riesce a restituire immagini di alta qualità perfino quando ci si trova nel range degli infrarossi.