

OPEN SOURCE IOT BLOCKCHAIN



IN QUESTO NUMERO:

BLYNK: COME CREARE UN PROGETTO IOT IN 5 MINUTI

LA NUOVA SCHEDA ARDUINO EDGE CONTROL

PROGETTO DI UN SISTEMA DI CONTROLLO DI SERVOMOTORI CON WEB SERVER - PARTE 3

SMALL CIRCUITS REVIVAL - PARTE 7 (DI ELEKTOR)

HEXARCHIA: LA BLOCKCHAIN APPLICATA AI VIDEOGIOCHI

E MOLTO ALTRO!

COSA LEGGERAI NEL 2021?

<i>TOPICS</i>	<i>MAKERS ZONE</i>	<i>DATA DI PUBBLICAZIONE</i>
Open Source IoT	Sensors	1 Febbraio
Automation	Industry4.0	1 Marzo
Energy Management	Energy Harvesting	1 Aprile
Wireless/RF	Smart Projects	1 Maggio
Automotive	Smart Mobility	1 Giugno
Open Source IoT	Blockchain	1 Luglio
Artificial Intelligence	Robotics	1 Settembre
LED/Optoelectronics	Smart Lighting	1 Ottobre
Power/Motor	Power Management	1 Novembre
Open Source IoT	Embedded Systems Design	1 Dicembre

GLI ARTICOLI

Elettronica Open Source

&

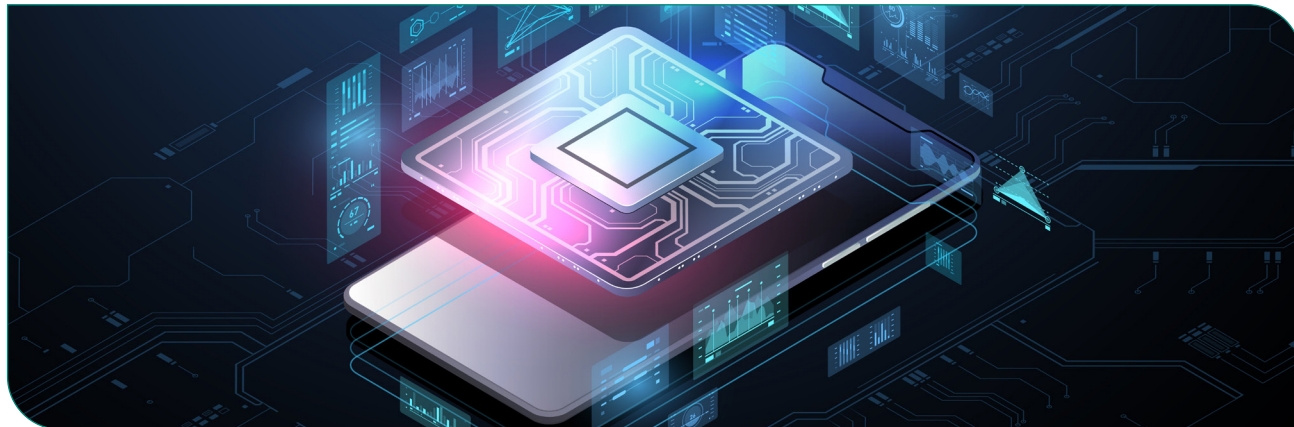
elektor
design > share > sell

**IN UN SOLO
MAGAZINE!**

**OGNI MESE
SU FIRMWARE 2.0**

**2 ARTICOLI
DI ELEKTOR**

OPEN SOURCE IOT BLOCKCHAIN



Founder&Editor

Emanuele Bonanni

CFO

Lidia Balica

Editorial Assistant

Maria Pisani

Maker in Chief

Giordana Francesca Brescia

Advertising & Marketing

Cristian Balica

cristian@contangosl.com

Graphic Designer

Marilde Mirra

Circulation

Users - 141.503

Social Network - 128.767

© Copyright

Tutti i diritti di riproduzione o di traduzione degli articoli pubblicati sono riservati. Manoscritti e disegni sono di proprietà di Contango SL.

E' vietata la riproduzione anche parziale degli articoli salvo espressa autorizzazione scritta dell'editore. I contenuti pubblicitari sono riportati senza responsabilità, a puro titolo informativo.

EDITORIALE

IOT E BLOCKCHAIN,
PILASTRI DELLA
TRASFORMAZIONE DIGITALE

4

BLYNK: COME CREARE
UN PROGETTO IOT IN
5 MINUTI

5

BLEB: COME DARE
VITA ALLA TUA
APP E INTEGRARE
COMPONENTI ESTERNI
CON I BLEBRICKS

11

L'INTERNET DELLE
COSE INDUSTRIALE
(IIOT) E I SISTEMI
DI CONTROLLO
DELL'AUTOMAZIONE

16

COS'È L'INTERNET OF
MUSICAL THINGS

20

UN SISTEMA IOT DI
CONTROLLO REMOTO
DEGLI ATTUATORI CON
L'ESP32

25

SERVOMOTORI E
ARDUINO: COME USARLI
CON LA SCHEDA A
MICRO-CONTROLLORE

34

SERVOMOTORI
E RASPBERRY PI:
COME USARLI CON IL
COMPUTER SINGLE
BOARD

37

PROGETTO DI UN
SISTEMA DI CONTROLLO
DI SERVOMOTORI CON
WEB SERVER - PARTE 3

40

LA NUOVA
SCHEDA ARDUINO
EDGE CONTROL
AUTOMATIZZA
L'AGRICOLTURA
INTELLIGENTE

52

MONITORAGGIO IOT
CON I SISTEMI SCADA
OPEN SOURCE

58

SMALL CIRCUITS
REVIVAL - PARTE 7
(SHORT-WAVE
AUDION)

62

RICONOSCIMENTO
FACCIALE PER LA
DOMOTICA CON
L'ESP32-CAM - PARTE 3

64

COME SEMPLIFICARE
IL DESIGN FOR
MANUFACTURING

85

TUTTO CIÒ CHE TI SERVE
PER COMINCIARE CON
BLOCKCHAIN

87

HEXARCHIA:
LA BLOCKCHAIN
APPLICATA AI
VIDEOGIOCHI

92

BLOCKCHAIN TRA
SCIENZA E SALUTE

97



PROTECT DETECT RECOVER

```
PLATFORM_FIRMWARE_RESILIENCY
mutable Root of Trust
#define FIRMWAREID 0x34
#define BUILDLABEL cec1712_G...
#define BUILDNUMBER 0x1400
*****
* sb_get_image();
Get next image to authenticate
@param None
@return SUCCESS or error codes
*****
uint8_t sb_get_image(void)
uint8_t ret_val;
bool valid_image, error
trace0(0, SECURE_BOOT_...
ret_val = SB_SUCCESS
do
/* Found a image to authenticate - I
```

Fidati del tuo firmware

Boot Protetto e Aggiornamenti del Firmware

Un rootkit è una forma di malware particolarmente insidiosa perché si carica prima dell'avvio del sistema operativo. Può nascondersi dai normali software antimalware ed è notoriamente difficile da rilevare. Un modo per difendersi dai rootkit è l'avvio sicuro. Il microcontroller CEC1712 Arm® Cortex®-M4-based ed il firmware Soteria G2 sono progettati per proteggere dalle minacce prima che queste possano essere caricate.

Il firmware CEC1712 e Soteria G2 semplificano l'aggiunta di una soluzione di boot sicuro/Hardware Root of Trust (HRoT) senza la necessità di modificare il tuo codice esistente. Puoi utilizzare il tuo firmware esistente così com'è! Questa soluzione agnostica è destinata a qualsiasi piattaforma che si avvia da una Flash SPI esterna. CEC1712 e Soteria G2 soddisfano le NIST SP 800-193 Platform Firmware Resiliency Guidelines per la protezione, il rilevamento e il ripristino da codice danneggiato. La soluzione è l'autoriparazione, con la possibilità di recuperare in caso di errore di autenticazione. Inoltre, la soluzione consente aggiornamenti firmware autentici.

Soteria G2 è altamente configurabile in base ai requisiti della tua piattaforma, potendo supportare fino a due processori applicativi e quattro dispositivi flash SPI, oltre alla protezione rollback del codice e revoca delle chiavi.



IOT E BLOCKCHAIN, PILASTRI DELLA TRASFORMAZIONE DIGITALE

Cari lettori, oggi vi presentiamo il nuovo numero della rivista tecnica digitale Firmware 2.0. Questo numero è dedicato alle tecnologie emergenti dell'Open Source IoT e della Blockchain. Sappiamo bene quanto l'IoT sia diventato fondamentale nel processo di digitalizzazione. Se possiamo scambiare in tempo reale grandi quantità di dati e informazioni, è proprio grazie alla tecnologia dell'Internet of Things. Attraverso oggetti intelligenti interconnessi possiamo monitorare in tempo reale i parametri associati alla nostra attività.

Grazie all'IoT possiamo creare applicazioni intelligenti che collegano il mondo fisico con quello virtuale. Nella vita di tutti i giorni utilizziamo servizi dell'IoT e dispositivi smart connessi alla rete, pensiamo alle applicazioni di domotica intelligente, alle piattaforme per la gestione dei dati sulla mobilità, ai dispositivi indossabili per monitorare le condizioni di salute, ai pagamenti digitali e molto altro. L'utilizzo di dispositivi connessi a Internet, unitamente ai sensori, è ormai parte integrante dello sviluppo di qualunque applicazione smart.

Oggigiorno sono molte le realtà industriali che implementano le tecnologie dell'IoT per offrire servizi ai propri clienti con soluzioni scalabili e customizzabili, riducendo anche i costi e i rischi di errore umano. Un dispositivo intelligente dotato di sensore, in grado di comunicare con altri device e con l'ambiente circostante, se ci pensiamo, fa esattamente questo, offre un servizio all'utente finale acquisendo dati e consentendone il processamento.

Accanto all'IoT, la tecnologia distribuita Blockchain, inizialmente applicata solo al contesto finanziario, trova spazio anche in ambiti prima inimmaginabili. I principali settori di business utilizzano le Blockchain per gestire la tracciabilità della filiera produttiva. La tecnologia Blockchain attribuisce alle transazioni digitali sicurezza tramite algoritmi crittografici, trasparenza, affidabilità e scalabilità nello scambio di informazioni. Gestire le enormi quantità di dati tra dispositivi senza la necessità di un controllo centralizzato è la vera rivoluzione.

Pertanto, se da un lato l'IoT rappresenta la tecnologia per lo scambio dei dati, la Blockchain ne fornisce il metodo e gli attributi. Possiamo considerare la tecnologia Blockchain come l'infrastruttura di riferimento per i miliardi di dispositivi interconnessi che popolano la rete.

Questo rende IoT e Blockchain un binomio indissolubile. Due tecnologie in grado di creare valore, due strumenti che, seppur apparentemente diversi per definizione e funzionamento, appartengono entrambi al complesso e affascinante mondo digitale in continua evoluzione.

Buona lettura!

Giordana Francesca Brescia

BLYNK: COME CREARE UN PROGETTO IOT IN 5 MINUTI

di Andrea Garrapa

Blynk è la più popolare piattaforma per connettere i dispositivi IoT al cloud, progettare app per controllarli, analizzare i dati di telemetria e gestire prodotti distribuiti su larga scala. Blynk è stato progettato appositamente per l'Internet of Things ed offre la possibilità di controllare l'hardware da remoto, visualizzare i dati dei sensori, memorizzarli e fare molte altre cose. Sebbene esistano altri servizi per il controllo dell'hardware in rete (ThingSpeak, IFTTT), Blynk è tra le più user-friendly, è gratuita ed anche open source. In questo articolo descriveremo attraverso l'implementazione di alcuni esempi pratici, la semplicità d'uso e le ampie potenzialità della piattaforma Blynk.

INTRODUZIONE

Blynk è una piattaforma che consente di creare rapidamente interfacce per il controllo e il monitoraggio dei progetti hardware dal proprio dispositivo iOS e Android. L'applicazione Blynk consente di creare una dashboard del progetto e disporre pulsanti, cursori, grafici e altri widget sullo schermo. Utilizzando i **widget** è possibile attivare e disattivare i pin, visualizzare i dati dai sensori e molto altro.

Ci sono tre componenti principali che costituiscono la piattaforma Blynk:

- **Blynk App** - è l'applicazione per smartphone che consente di creare interfacce per i progetti utilizzando i vari widget forniti.
- **Blynk Server** - è il servizio responsabile di tutte le comunicazioni tra lo smartphone e l'hardware. Si può utilizzare il Blynk Cloud o eseguire il proprio server Blynk privato in locale. È **open source**, può facilmente gestire migliaia di dispositivi e **può anche essere lanciato su un Raspberry Pi**.
- **Librerie Blynk** - consentono la comunicazione con il server per tutte le piattaforme hardware più diffuse ed elaborano tutti i comandi in entrata e in uscita.

Uno scenario applicativo della piattaforma potrebbe essere quello rappresentato in **Figura 1**. Ogni volta che si preme un pulsante nell'**App Blynk**, un messaggio viaggia nello spazio **Blynk Cloud**, dove viene instradato verso

l'**hardware** di destinazione. Funziona allo stesso modo nella direzione opposta e tutto accade in un batter d'occhio.

Caratteristiche

Le principali caratteristiche della piattaforma **Blynk** sono:

- API e interfaccia utente simili per tutti i dispositivi e hardware supportati
- Connessione al cloud tramite: Wi-Fi, Bluetooth e BLE, Ethernet, USB (seriale), GSM
- Set di widget facili da usare
- Manipolazione diretta dei pin senza scrittura di codice
- Facile da integrare e aggiungere nuove funzionalità utilizzando pin virtuali
- Monitoraggio dei dati storici tramite widget Super-Chart
- Comunicazione da dispositivo a dispositivo tramite Bridge Widget
- Invio di e-mail, tweet, notifiche, etc.

La **Libreria Blynk**, una volta installata nell'IDE Arduino, dà accesso a sketch di esempio che coprono le funzionalità base di Blynk.

Requisiti per l'uso

Per iniziare ad utilizzare la piattaforma, sono necessarie solo un paio di cose:

1. **Hardware** - Blynk funziona su Internet. Ciò signifi-

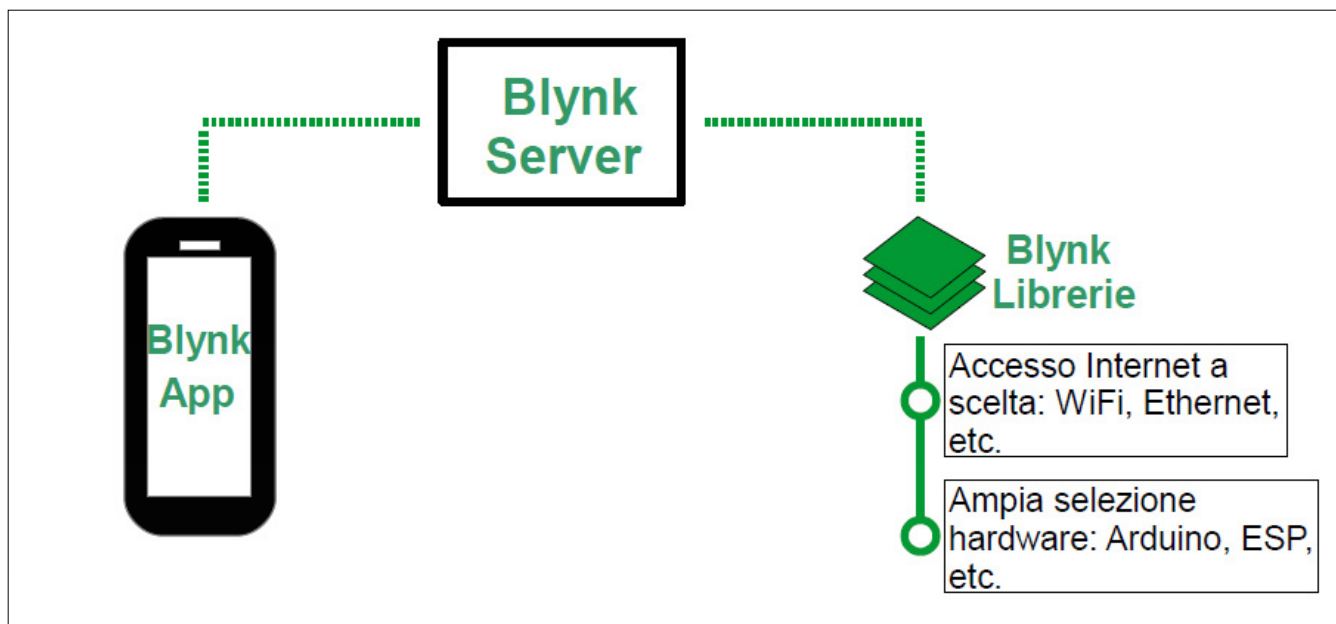


Figura 1: rappresentazione schematica di uno scenario applicativo della piattaforma Blynk

fica che l'hardware scelto dovrebbe essere in grado di connettersi a Internet. Alcune delle schede, come Arduino Uno, avranno bisogno di uno Shield Ethernet o Wi-Fi per poter comunicare, mentre altre sono già abilitate come l'ESP8266. La cosa interessante è che l'elenco di hardware supportati da Blynk è enorme e in continua espansione.

2. **Smartphone** - L'App Blynk è un generatore di interfacce ben progettato. Funziona sia su iOS che su Android.

Il miglior modo per descrivere la piattaforma è attraverso l'implementazione di alcuni esempi pratici che mettano in mostra la semplicità e la rapidità con cui è possibile realizzare delle applicazioni IoT.

Configurazione dell'App Blynk

Per prima cosa scarichiamo l'**App Blynk** dallo store. Terminata l'installazione, aprendo l'app dovremo creare un **nuovo account Blynk**. Un account è necessario per salvare i progetti e accedervi da più dispositivi da qualsiasi luogo. Dopo aver effettuato correttamente l'accesso all'account, passiamo alla creazione di un **nuovo progetto**, facendo click su "New Project" (**Figura 2a**).

Inseriamo il nome del progetto nell'apposito spazio (**Figura 2b**) e selezioniamo il dispositivo hardware da utilizzare, tra quelli disponibili nel menu a tendina. Nel nostro caso si tratta del modulo **ESP8266**. A seconda del dispositivo selezionato potremo scegliere anche il tipo di connessione: Wi-Fi, Ethernet, etc.

Infine, facendo click su "Create" (**Figura 2b**) avremo cre-

QUELLO CHE HAI LETTO E' UN ESTRATTO, L'ARTICOLO COMPLETO E' RISERVATO AGLI ABBONATI AD ELETTRONICA OPEN SOURCE.

PERCHE' ABBONARSI A PLATINUM 2.0?

UN ANNO DI **FIRMWARE 2.0**
TUTTI GLI ARTICOLI TECNICI RISERVATI
CONTEST E PROMOZIONI RISERVATI



VOGLIO ABBONARMI!

BLEB: COME DARE VITA ALLA TUA APP E INTEGRARE COMPONENTI ESTERNI CON I BLEBRICKS

di **Lorenzo Neri**

Bleb ti permette di dare vita alla tua app Android e di integrare componenti esterni con i Blebricks. Tutto questo è possibile senza saper programmare per Android e tramite la GPIO dei Blebricks stessi. In questo articolo ti accompagnerò in questa avventura un passo alla volta, scoprendo come sia possibile realizzare un'app Android senza neanche una riga di codice e farla interagire con i Blebricks, ma non solo. Vedremo assieme come sia possibile mettere in comunicazione i Blebricks stessi con componenti elettronici esterni.

Di **Bleb**, la realtà italiana nata grazie all'Ingegnere Fabrizio Innocenti con l'aiuto di suo figlio Lorenzo, ho già parlato ampiamente in [questo articolo](#) offrendo una panoramica completa della loro dimensione ma non solo.

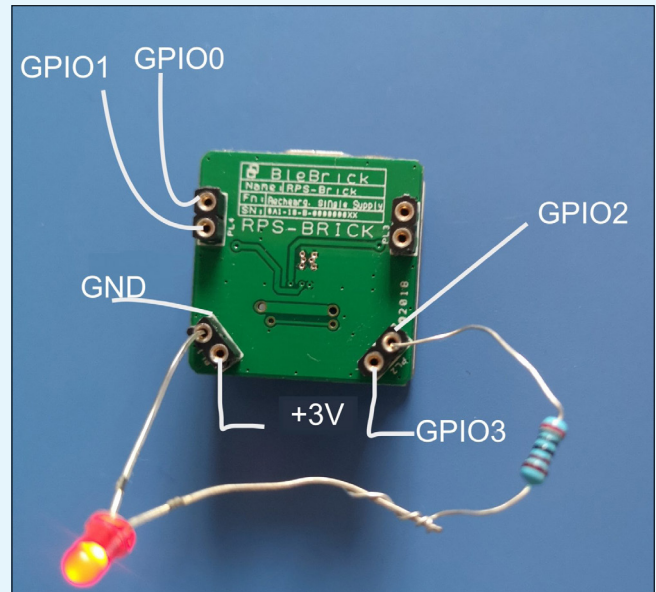
Devi sapere che il loro obiettivo, è quello di permetterti di dare vita ai tuoi progetti IoT in pochi passi attraverso i **Blebricks** e la prototipazione istantanea. Il processo è veramente semplice: se vuoi dare vita al tuo primo progetto e capire come funzionano i Blebricks, puoi leggere [questo mio secondo articolo](#), ma **ricordati di tornare qui!**

Perché in questo articolo, ti mostrerò come creare la tua app Android senza saper programmare per Android **ma non solo**: scoprirai come sia possibile unire i Blebricks a componenti elettronici esterni.

INTEGRARE I COMPONENTI ESTERNI CON I BLEBRICKS

Partiamo dal punto più immediato: integrare componenti esterni con i Blebricks, perché non si limitano a connettersi fra loro no!

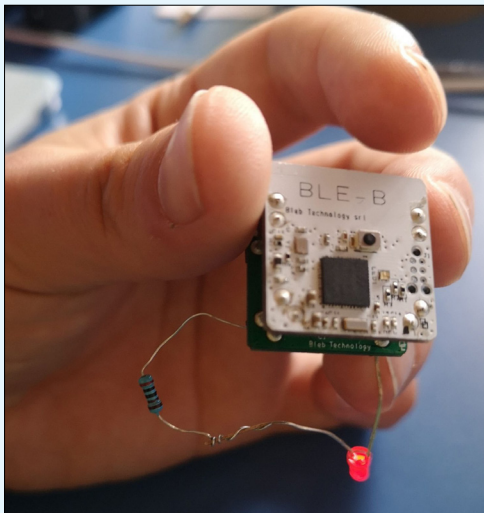
Al contrario offrono una GPIO a tutti gli effetti con quattro pin, alimentazione e massa:



Il Blebrick RPS con i dettagli della GPIO

Quello che hai di fronte a te è un **Blebrick RPS in versione Expert**, ovvero senza housing. Ma Bleb ha pensato alle esigenze di tutti: offrono i Blebricks anche in formato Elite dotati di case **che puoi scoprire direttamente qui!** Nel dettaglio, puoi vedere che vengono offerti quattro pin GPIO: ciascuno di essi può essere impiegato *sia come*

sorgente di input, sia come sorgente di output, **digitale e analogica a seconda delle tue esigenze**. Nel nostro esempio, utilizzerò un LED con una resistenza per mostrarti come avviene la comunicazione tra Blebricks e componenti esterni:



Poiché la GPIO eroga al massimo 15mA di corrente, può andare bene una resistenza da 1kOhm come nel mio esempio, collegata all'anodo e al pin numero due della GPIO stessa.

Per accendere il LED, mi è sufficiente impostare come sorgente di output digitale il pin #2 e **impostarne il valore a 255**. I valori ammessi sono compresi **fra 0 e 255 inclusi**:



L'impostazione della GPIO tramite MakeApp

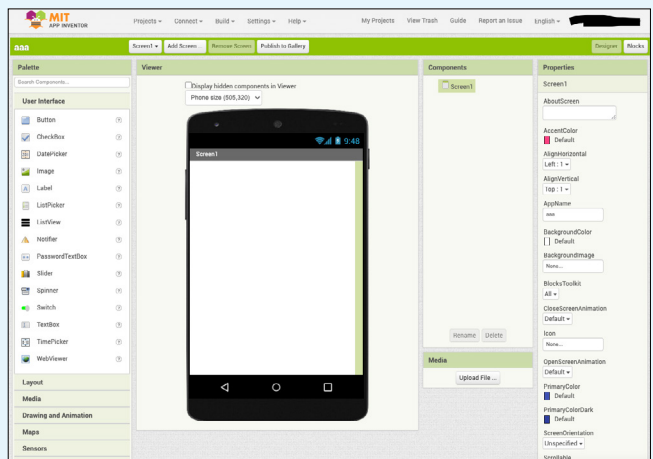
Così facendo, il LED si illuminerà: **ma voglio fare di più**. Una volta capito come integrare componenti esterni con i Blebricks vediamo assieme come realizzare un'applicazione Android tutta nostra da zero e **senza saper programmare e senza scrivere una singola riga di codice!**

COME DARE VITA ALLA TUA APP GRAZIE AD APP INVENTOR E L'ESTENSIONE DI BLEB

Bleb non ci vuole far mancare nulla nemmeno lato software. Certamente, c'è MakeApp, l'applicazione ufficiale che puoi scaricare **su PlayStore con un click qui**, ma possiamo anche realizzarne una nostra. Come? Grazie ad App Inventor: il software realizzato dal MIT per permettere ai bambini di imparare a creare applicazioni Android **senza saper programmare**. Bleb, che come realtà è stata premiata proprio dal MIT **per il suo contributo, ci offre un'estensione semplificata per usare i Blebricks!**

Il mio obiettivo è quello di realizzare un'applicazione che mi mostri la temperatura rilevata dal sensore presente sul BLE-B base e tramite un bottone mi permetta di accendere e spegnere il LED collegato alla GPIO. La prima cosa da fare è dirigersi sul sito di App Inventor che ti lascio a distanza di un **click qui** e cliccare sul bottone "Create Apps" messo bene in evidenza. Verrà richiesta la registrazione: non ti preoccupare, è completamente gratis!

A questo punto, **creo un nuovo progetto cliccando su "Projects" e poi "Start a new project"**. Con molta fantasia dò il nome della mia app che sarà **"LaMiaAppBleb"** per trovarmi infine su questa schermata:



L'interfaccia di App Inventor con tutti gli strumenti messi a disposizione

Vediamo che ci sono parecchie cose a nostra disposizione, ma prima di proseguire, abbiamo bisogno di un pezzo

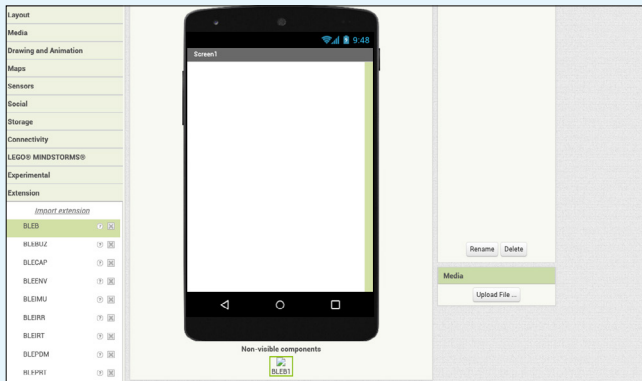
fondamentale: *l'estensione creata da Bleb per far interagire la nostra applicazione con i Blebricks.*

SCARICA L'ESTENSIONE PER APP INVENTOR

Nella sezione a sinistra di App Inventor, ovvero "Palette", scorrendo verso il basso troveremo la voce "Extension": la espando e tramite l'opzione "Import extension" carico l'estensione.

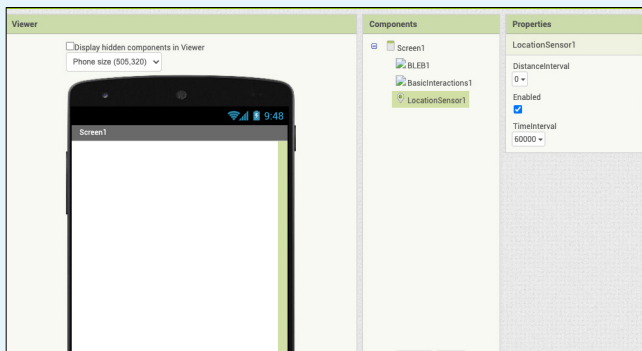
A questo punto, ciò di cui ho bisogno sono tre componenti che non saranno visibili nell'applicazione ma sono essenziali per la comunicazione BLE fra il nostro smartphone Android e il BLE-B. Sempre tramite la stessa sezione "Extension" potrai notare che sono apparse parecchie voci in più!

Tra queste, seleziono "**BLEB**" e "**BasicInteractions**" una per volta semplicemente trascinandole sopra lo smartphone che appare al centro della schermata:



Il componente "BLEB" appare come "non visibile"

La stessa cosa la dovrò ripetere tramite la sezione "**Sensors**": la espando e aggiungo il terzo elemento non visibile necessario per dare vita a tutto, ovvero "**LocationSensor**".



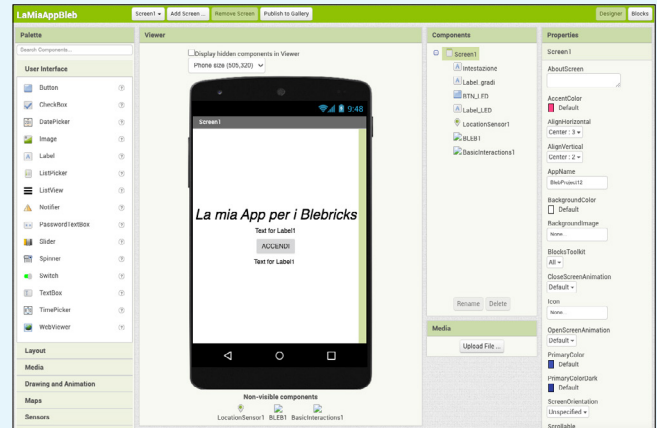
I componenti che vanno a costituire la nostra applicazione

A questo punto, possiamo passare ad inserire i **compo-**

nenti grafici della nostra applicazione.

DIAMO VITA ALL'ASPETTO GRAFICO DELLA TUA APP

Ciò che dovremo fare è raggiungere questo aspetto:



La tua applicazione fatta e finita da un punto di vista grafico

Tramite la sezione "**User Interface**", trascino sullo schermo un oggetto di tipo "**Label**". Attraverso la sezione "**Properties**", dopo aver cliccato sull'oggetto stesso, faccio il necessario per posizionarlo al centro, renderlo in corsivo. Dimenticavo: ci voglio scrivere "*La mia App per i Blebricks*".

Prima ancora di ciò, le voglio dare un nome ben specifico, ovvero "**Intestazione**". Per farlo, mi è sufficiente fare un **click** sull'elemento e tramite il pulsante "**Rename**" procedere a rinominarlo.

Per apportare le modifiche grafiche di cui abbiamo parlato tramite il pannello "Properties" spunto la casella "FontTalic", imposto il "FontSize" a 30, mentre la voce "Text" è quella che mi permette di impostare il testo come vedi dallo screenshot. In ultima battuta, per allineare al centro imposto la voce "TextAlignment" con il valore "center: 1".

Aggiungo una seconda label subito sotto la prima, limitandomi solo ad impostare l'allineamento al centro, per poi ripetere l'operazione con un oggetto di tipo "Button" e una terza label: questi tre oggetti li rinomino rispettivamente in "Label_gradi", "BTN_LED" e "Label_LED". Immagino tu abbia già capito qual è il loro scopo.

Ultima cosa che faccio da un punto di vista grafico è impostare il testo del bottone ad "ACCENDI" tramite la field "Text" nella sezione "Properties".

“Le due label piccole servono rispettivamente a mostrare la temperatura rilevata in tempo reale dal BLE-B e lo stato del LED collegato tramite la GPIO

IL FUNZIONAMENTO DELLA TUA APP: PASSIAMO AI BLOCCHI

Non sarebbe un'applicazione senza la giusta dose di logica, ma non ti preoccupare: stiamo per darle vita! In alto a destra, sulla barra verde principale della GUI di App Inventor, noterai due pulsanti: "Designer" e "Blocks".

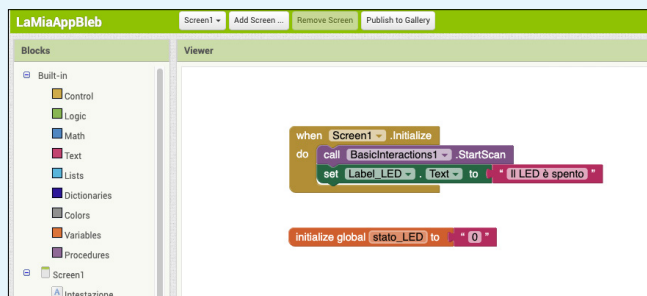
Fino ad ora abbiamo usato la prima sezione, per dare vita alla logica passiamo alla seconda con un click sul bottone. La logica di App Inventor, come ti dicevo all'inizio dell'articolo, si basa sui blocchi e quindi non dovremo scrivere codice!

Rispecchia di molto la logica IFTTT su cui si basa tutto il mondo Bleb, ovvero "Se succede questo allora fai quello". Ragionando sul primo avvio, quindi quando apriamo l'applicazione, dovremo inizializzare la connessione al BLE-B e impostare una variabile che gestirà lo stato del LED, inizialmente impostato a "spento".

Per fare ciò, attraverso la sezione "Blocks" faccio un click su "Screen1" e seleziono il blocco a ferro di cavallo che recita "When Screen1 initialize".

Lo trascino sull'area a destra, per poi cliccare su "BasicInteractions1": seleziono il blocco di colore viola con il testo "call BasicInteractions1 StartScan", il cui scopo è quello di attivare il Bluetooth del nostro dispositivo e instaurare la connessione con il BLE-B più vicino ad esso.

Tramite la voce **"Variables"** seleziono il blocco **"Initialize global"**, per trascinarlo sull'area:



Sulla sinistra abbiamo la sezione **"Blocks"** che ci permet-

te di trascinare su ciò che è il **"Viewer"** i mattoncini che compongono la logica della nostra applicazione.

Per prima cosa ci occuperemo dell'avvio dell'applicazione.

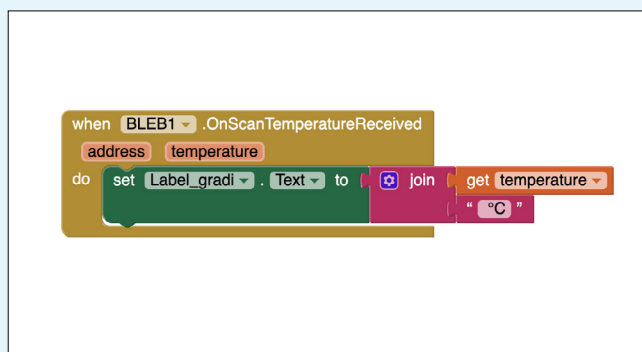
All'avvio, quindi tramite il blocco "When Screen1.Initialize", inseriremo ad incastro il blocco "call BasicInteractions1.StartScan" ed il blocco verde "set Label_LED" assieme a quello viola preso dalla sezione "Text".

Il primo, quello viola, **permette di connettere il nostro dispositivo al BLE-B più vicino**.

Il secondo **serve ad impostare il testo della label dove mostreremo lo stato del LED connesso alla GPIO**: per farlo, **valorizzo il contenuto del blocco viola con "II LED è spento"**.

Infine, inizializzo una variabile globale che chiamo **"stato_LED"** ed **inizializzo a "0"**: sarà necessaria per gestire lo stato del LED.

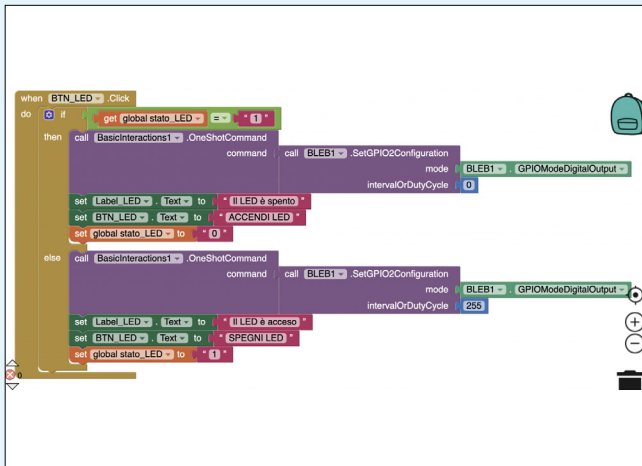
Proseguiamo nella nostra avventura per costruire blocco su blocco ciò che ci serve a visualizzare la temperatura sullo schermo tramite la seconda label:



Per far sì che la lettura della temperatura avvenga periodicamente, utilizzo il blocco "When BLEB1.OnScanTemperatureReceived".

Al suo interno, come ho fatto prima, sfrutto gli stessi blocchi per **impostare il testo della seconda label con la temperatura letta tramite il blocco "get temperature"**. Molto semplice e pratico.

Ma ora che hai visto come dare vita alla tua app, manca un pezzetto: **interagire con i componenti esterni!** Per farlo, dovremo **costruire la logica che sta dietro al bottone all'interno dell'app**:



Il blocco principale è un "when click": molto banalmente, quando il bottone viene premuto, per poi integrare dentro un "if then else".

Il blocco "if then else" serve a verificare lo stato della variabile globale "stato_LED" che quando valorizzata a zero, significa che dovremo accendere il LED, diversamente l'operazione da eseguire consiste nello spegnerlo.

E ovviamente, invertire il valore della variabile stessa, ma non solo! Il terzo blocco viola, composto in realtà da due elementi dello stesso colore, ci permette di lanciare un comando BLE che imposta il pin numero due della GPIO come sorgente digitale di output il cui valore è uguale a zero.

Tutto questo perché, è il pin a cui abbiamo collegato l'anodo del LED.

Seguono poi i tre blocchi che modificano la scritta all'interno del bottone stesso per indicare l'inversione dell'operazione a cui serve, altrettanto per la label che indica lo stato del LED e infine ciò che ci serve per invertire il valore della variabile globale.

“Devi sapere che Bleb ha pensato a tutto: non solo ad offrirci i Blebricks, non solo ad offrirci questa estensione per App Inventor, ma anche una guida dettagliata per tutti i blocchi che hanno realizzato per dare vita alle tue applicazioni!”

E la puoi trovare a distanza di **un click direttamente qui!** A questo punto, non ci manca nient'altro se non... **Testare**

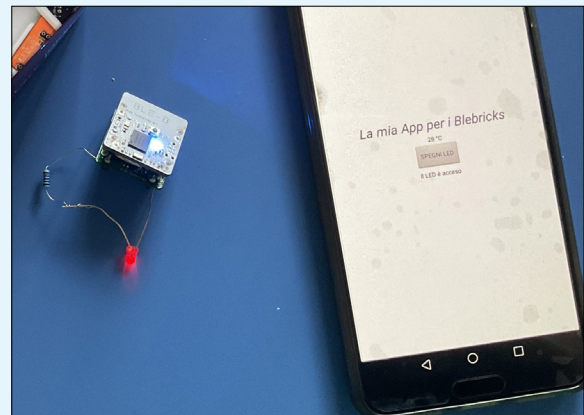
il tutto!

INSTALLAZIONE DELL'APP SUL NOSTRO SMARTPHONE

Abbiamo visto quanto sia davvero semplice dare vita ad un'applicazione e farla interagire con i Blebricks, ma è giusto testare anche il frutto del nostro impegno!

Tramite la voce "Build" posta in cima alla finestra di App Inventor, possiamo generare un file "apk" da installare direttamente sul nostro smartphone, diversamente possiamo generare un QR da far leggere all'applicazione **"MIT AI2 Companion"** anch'essa gratuita, che si occuperà di installare l'applicazione.

Una volta eseguita una delle due strade, **puoi testare con mano i risultati del progetto:**



L'applicazione in funzionamento con i Blebricks

In conclusione, devi sapere che anche gli altri Blebricks offerti da Bleb possono essere utilizzati nelle tue applicazioni fatte con App Inventor e non posso che invitarti a scoprirli!

SCOPRI I BLEBRICKS

L'autore è a disposizione nei commenti per eventuali approfondimenti sul tema dell'Articolo. Di seguito il link per accedere direttamente all'articolo sul Blog e partecipare alla discussione:

<https://it.emcelettronica.com/bleb-come-dare-vita-alla-tua-app-e-integrare-componenti-esterni-con-i-blebricks>

PROGETTO DI UN SISTEMA DI CONTROLLO DI SERVOMOTORI CON WEB SERVER - PARTE 3

di Fulvio De Santis

Il precedente articolo "Progetto di un sistema di controllo di servomotori con web server - Parte 2" è stato orientato alla trattazione della scheda di sviluppo ESP32 in quanto abbiamo ritenuto che sia propedeutica alla migliore comprensione di questo tutorial, ma anche per altri progetti che i lettori desiderino realizzare. In questa Parte 3, descriveremo l'altro componente del progetto, il servomotore SG90, poi tratteremo la programmazione dell'ESP32 mediante l'IDE di Arduino e, infine, la realizzazione e il collaudo del prototipo.

IL SERVOMOTORE SG90

Nel progetto/tutorial **Progetto di un sistema di controllo di servomotori con web server**, oltre all'ESP32, viene impiegato il micro servomotore SG90. Anche se di ridotte dimensioni da 32 mm x 23 mm, l'SG90 è leggero e ha un'elevata potenza di torsione (Torque) di 2,5 kg/cm, come indicato nel datasheet. In **Figura 1** viene illustrato il servomotore SG90.

Il servo può ruotare di circa 180° in ogni direzione (+/- 90°) e ha un funzionamento standard come gli altri servomotori. È possibile utilizzare qualsiasi **codice** per servomotori o apposite **librerie** per controllarne l'operatività. In **Figura 2**, estratta dal datasheet dell'SG90, viene riportata la forma d'onda impulsiva del segnale PWM di controllo da applicare al servomotore SG90.

Esaminando la **Figura 2** si possono già rilevare le specifiche tecniche del **segnale PWM** di controllo del servomotore: l'ampiezza del fronte positivo dell'impulso deve essere compresa fra 4,8 V e 5 V; la frequenza deve essere 50 Hz, corrispondente al periodo di 20 ms; la variazione della larghezza dell'impulso deve essere compresa fra 1 ms e 2 ms, corrispondente al range di regolazione della rotazione dell'albero motore da 0° a 180°. La rotazione di "0°", corrispondente alla posizione centrale dell'albero motore, si ottiene con un impulso di larghezza di 1,5 ms; la rotazione di "+90°" con un impulso di larghezza di 2 ms; infine, la rotazione di "-90°" con un impulso di 1 ms. Per generare questo segnale di controllo con l'ESP32 utilizzeremo

la **libreria "Servo" di Arduino**, come vedremo nella descrizione della programmazione dell'ESP32. Nella **Figura 3**, anch'essa estratta dal datasheet dell'SG90, sono mostrate le caratteristiche dimensionali e di funzionamento dell'SG90, ivi compresa l'indicazione dei pin evidenziata dai colori corrispondenti ai conduttori del servomotore. A questo proposito, si tenga conto che il pin PWM può capitare anche di colore diverso, ad esempio di colore giallo o bianco.

Nella tabella all'interno della **Figura 3**, sono riportate le specifiche di funzionamento: la tensione di alimentazione VCC deve essere compresa fra 4,8 V e 6 V; il tempo di variazione angolare (Speed) dell'albero motore per un'applicata larghezza dell'impulso PWM è di 0,1.

PROGRAMMAZIONE DELL'ESP32

Prima di procedere con la programmazione dell'ESP32, realizzate su una breadboard sperimentale i collegamenti dei componenti del progetto seguendo lo schema dei collegamenti di **Figura 4**. Fissate sull'albero motore una delle alette in dotazione con l'SG90, così potrete vedere distintamente la rotazione dell'albero motore durante il funzionamento.

Il pin GPIO13 dell'ESP32 viene utilizzato come uscita con funzionalità PWM, come mostrato nello schema di **Figura 4**. È possibile anche usare qualsiasi pin PWM per generare il segnale di controllo, tranne i pin di ingresso digitale GPIO34 e GPIO39, nel qual caso è necessario specificare



Figura 1: il servomotore SG90

1 - 2 ms
Duty Cycle

QUELLO CHE HAI LETTO E' UN ESTRATTO, L'ARTICOLO COMPLETO E' RISERVATO AGLI ABBONATI AD ELETTRONICA OPEN SOURCE.

PERCHE' ABBONARSI A PLATINUM 2.0?

**UN ANNO DI FIRMWARE 2.0
TUTTI GLI ARTICOLI TECNICI RISERVATI
CONTEST E PROMOZIONI RISERVATI**



VOGLIO ABBONARMI!

LA NUOVA SCHEDA ARDUINO EDGE CONTROL AUTOMATIZZA L'AGRICOLTURA INTELLIGENTE

di **Giordana Francesca Brescia**

Arduino Edge Control è il nuovo dispositivo della serie Arduino PRO, di facile implementazione, adatto per applicazioni di monitoraggio in tempo reale e controllo remoto, ottimizzato per applicazioni in ambienti esterni. Si amplia quindi la gamma di componenti della famiglia Arduino, la piattaforma a microcontrollore made in Italy, amatissima da makers, progettisti e appassionati di elettronica. In questo articolo vedremo cosa è Arduino Edge Control e cosa si può realizzare con questa potente e versatile scheda.

UNA PANORAMICA INTRODUTTIVA

Sono trascorsi ormai ben sedici anni dall'uscita della prima scheda a microcontrollore Arduino. Il progetto Arduino nasce presso l'Istituto di Interaction Design di Ivrea grazie alla collaborazione di cinque docenti. La scheda elettronica programmabile Arduino è da sempre considerata la base per la realizzazione di progetti in modo rapido e intuitivo, un must per i progettisti elettronici, gli appassionati ed i makers di tutto il mondo. Nel corso del tempo si sono susseguiti diversi modelli della scheda e, parallelamente, si è andata ad ampliare in modo consistente la gamma di applicazioni realizzabili, spaziando da piccoli progetti di hobbistica a prototipi di robotica, meccanica o monitoraggio ambientale. Un lasso di tempo nel quale sono state sviluppate e diffuse sul mercato soluzioni che solo fino a qualche anno prima erano del tutto impensabili. **Arduino Edge Control (Figura 1, Figura 2, Figura 3)** è la nuova scheda di casa Arduino, lanciata da poco sul mercato embedded delle schede a microcontrollore. Dal design robusto e accattivante, Arduino Edge Control ha sin da subito trovato la sua naturale collocazione nella moltitudine di applicazioni verso le quali si concentrano

gli sforzi dei produttori di componenti elettronici, per fornire soluzioni sempre al massimo delle possibilità e delle prestazioni. Vediamo ora, nello specifico, di cosa si tratta. L'hardware open source Arduino Edge Control apporta creatività e innovazione all'agricoltura moderna. Allo stesso tempo, è una soluzione sicura e altamente scalabile. La scheda è in grado di ottimizzare il monitoraggio e il controllo remoto nelle applicazioni dell'agricoltura intelligente, nell'agricoltura di precisione e in tutte quelle applicazioni che richiedono un controllo intelligente e automatizzato in luoghi remoti. Si tratta di settori in forte crescita, dei quali sentiremo sempre di più parlare in futuro, grazie anche alle moderne tecnologie che li supportano. Se volessimo invece definire il tipo di mercato al quale si rivolge, è sicuramente quello delle applicazioni high level, si tratta infatti di un hardware di livello professionale. Come sempre accade quando vengono lanciate nuove tecnologie nel mondo Arduino, **la partecipazione della community sarà preziosa per sviluppare e testare progetti anche con questo nuovo hardware.** L'innovativa piattaforma Arduino, infatti, vanta una grande community di milioni di utenti a livello mondiale, fatta di makers e progettisti

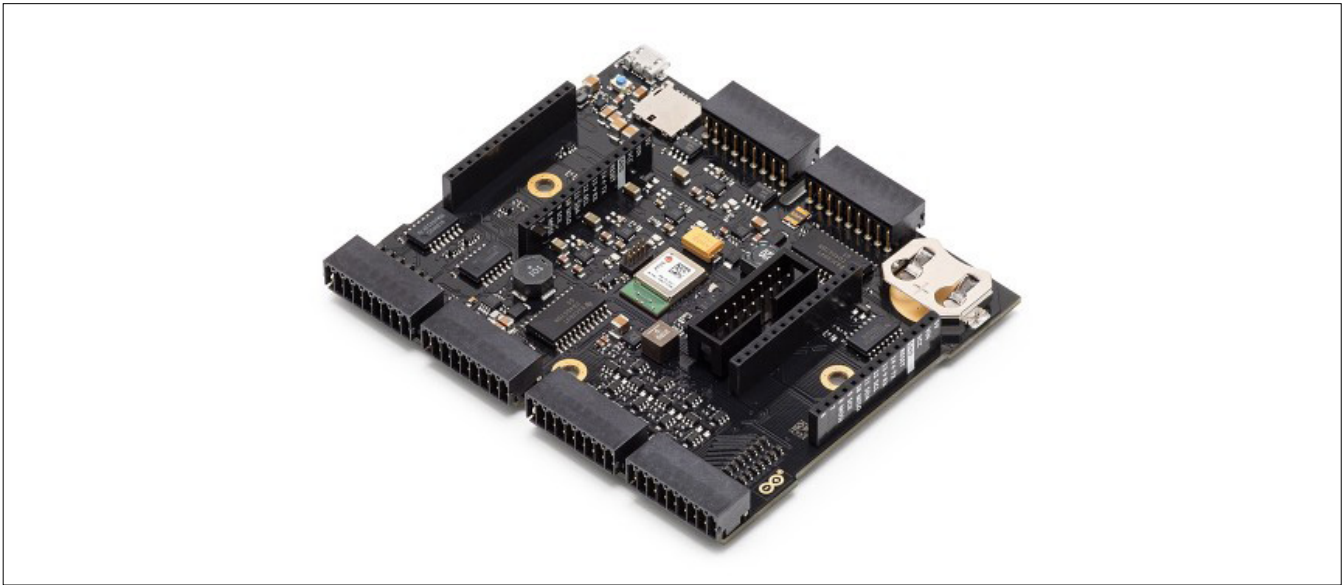
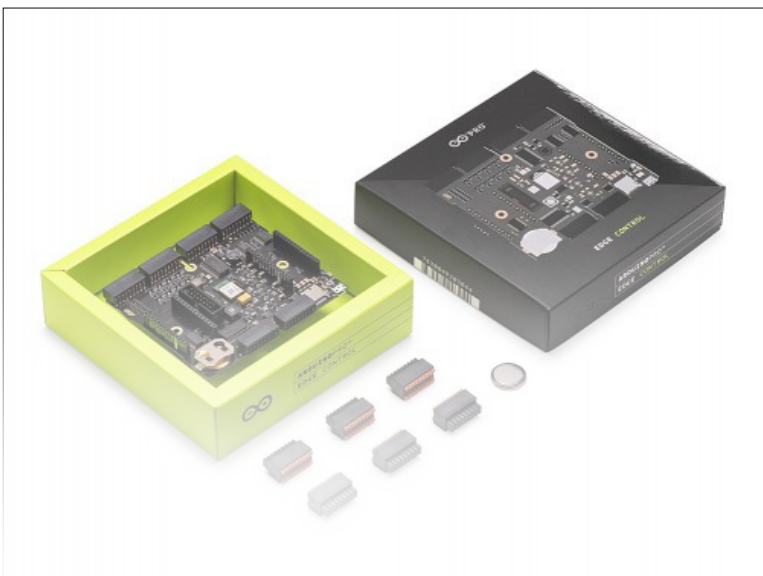


Figura 1. La scheda Arduino Edge Control



ne nell'agricoltura di precisione. Arduino Edge Control ha tutto ciò che serve per realizzare un progetto IoT. Se ci pensiamo, ciò che serve per realizzare un progetto IoT è, innanzitutto, un hardware open source flessibile e scalabile, una piattaforma di sviluppo, un set di sensori e una app per il monitoraggio da remoto.

Grazie all'Arduino Edge Control può essere migliorata la qualità del raccolto, oltre al monitoraggio intelligente in tempo reale di condizioni ambientali, qualità del suolo e crescita delle colture. C'è un fattore importante da tenere in considerazione. Oggigiorno i dispositivi IoT non sono percepiti solo come semplici prodotti, tutt'altro, sono dei veri e propri servizi che permettono di gestire in modo automatizzato

QUELLO CHE HAI LETTO E' UN ESTRATTO, L'ARTICOLO COMPLETO E' RISERVATO AGLI ABBONATI AD ELETTRONICA OPEN SOURCE.

PERCHE' ABBONARSI A PLATINUM 2.0?

**UN ANNO DI FIRMWARE 2.0
TUTTI GLI ARTICOLI TECNICI RISERVATI
CONTEST E PROMOZIONI RISERVATI**



VOGLIO ABBONARMI!

SMALL CIRCUITS REVIVAL – PARTE 7 (SHORT-WAVE AUDION)

Oltre al condensatore di tuning, il ricevitore short-wave a tre stadi qui descritto (Figura 1) ha un regolatore aggiuntivo per il feedback. In linea di principio, questo primo stadio è un **oscillatore** il cui punto di lavoro è regolabile, consentendo di ridurre il guadagno.

Il trucco sta nel regolare il guadagno in modo tale che tutte le perdite nel loop dell'oscillatore siano quasi compensate, ovvero, il tutto è sul punto dell'auto-oscillazione.

A questo punto il ricevitore mostra la sua massima sensibilità e la migliore selettività.

Lo stadio oscillatore PNP (in configurazione common-collector) funziona simultaneamente come un **audion** e un

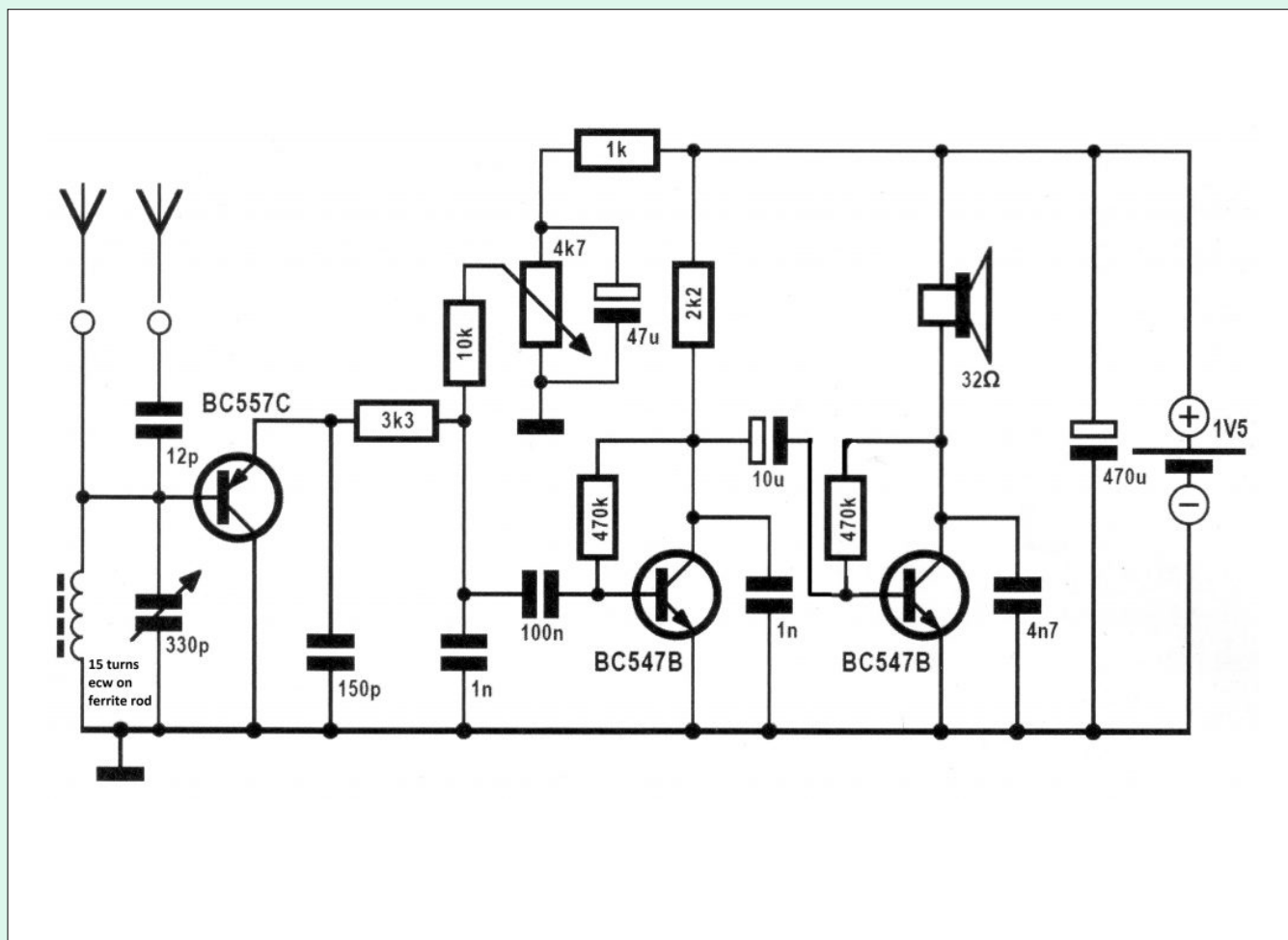


Figura 1

demodulatore per il segnale RF. I due stadi AF che seguono assicurano che ci sia sufficiente potenza audio in modo che anche un piccolo altoparlante possa essere pilotato.

Se l'audion short-wave può effettivamente essere sintonizzato al punto in cui non oscilla, dipende anche dallo smorzamento del circuito di ingresso e dall'antenna collegata.

Questo è il motivo per cui questo circuito è dotato di due collegamenti dell'antenna.

Quando è collegato tramite il condensatore da 12 pF, è debolmente accoppiato e lo smorzamento è piccolo.

Al contrario, il collegamento diretto è (anche) adatto principalmente per antenne molto corte **poiché un'antenna più lunga irradia anche energia RF che smorza il circuito dell'oscillatore.**

Se regolato correttamente, un audion è estremamente sensibile; agli albori della tecnologia radio questi tipi di ricevitori facevano parte dell'equipaggiamento standard. Anche con trasmettitori deboli, potrebbero essere coperte

distanze fino a mille miglia.

WEB LINK

ARTICOLO ORIGINALE IN INGLESE AL LINK:

SMALL CIRCUITS REVIVAL – EPISODE 7 | ELEKTOR MAGAZINE



L'autore è a disposizione nei commenti per eventuali approfondimenti sul tema dell'Articolo. Di seguito il link per accedere direttamente all'articolo sul Blog e partecipare alla discussione:

<https://it.emcelettronica.com/small-circuits-revival-parte-7-short-wave-audion>

QUELLO CHE HAI LETTO E' UN ESTRATTO, L'ARTICOLO COMPLETO E' RISERVATO AGLI ABBONATI AD ELETTRONICA OPEN SOURCE.

PERCHE' ABBONARSI A PLATINUM 2.0?

UN ANNO DI **FIRMWARE 2.0**
TUTTI GLI ARTICOLI TECNICI RISERVATI
CONTEST E PROMOZIONI RISERVATI



VOGLIO ABBONARMI!

HEXARCHIA: LA BLOCKCHAIN APPLICATA AI VIDEOGIOCHI

di **Maila Agostini**

La tecnologia blockchain è ormai fondamentale per tutti i settori fortemente in espansione, come IoT e AI. Le principali applicazioni sono per i servizi finanziari, visto che si tratta di una specie di "libro mastro" di transazioni di vari tipi, che consente di bypassare banche, notai ed istituti finanziari. Tuttavia, è ancora poco applicata al settore del gaming; la maggior parte dei videogiochi online si basa su un modello centralizzato, ma questa tecnologia promette di far avere ai giocatori il pieno controllo sui propri account. Cosa significa questo? Ce lo spiega in questa intervista Carlo Moretti, studente dell'Università di Firenze, che con la software house Overdroid ha creato il primo videogioco italiano basato sulla blockchain.

CIAO CARLO! RACCONTACI CHI SIETE E COME SI È FORMATO IL VOSTRO GRUPPO!

Mi presento. Sono Carlo Moretti, ho 25 anni e sono laureando nella Laurea Magistrale in Economics presso l'Università di Firenze con una tesi su blockchain e la proposta di regolamento dell'Unione Europea del settembre 2020, la cosiddetta Markets in Crypto-Asset Directive (MICA) che, appunto, si propone

di portare chiarezza regolatoria nel nascente settore delle crypto-monete della tecnologia blockchain.

Oggi sono qui insieme a voi per parlare della start up *Overdroid*, una software house che sviluppa videogiochi di nuova generazione. Io sono l'amministratore ma il team si compone di altre tre persone: Denis Dellapasqua, Game Designer, Joseph Aliaga, di formazione un Game Designer che però ha studiato anche ingegneria



Carlo Moretti, 25 anni



informatica e supervisiona tutti gli aspetti tecnici della start-up ed Andrea Tomei, Lead Game Programmer. Joseph e Denis erano compagni di classe all'Accademia di Design, e con loro ci siamo conosciuti in occasione di alcuni eventi organizzati dall'Università di Firenze. Andrea si è aggiunto al nostro team qualche mese più tardi; ci siamo conosciuti online poi è nata un'amicizia ed anche una collaborazione professionale.

SO CHE CON QUESTO PROGETTO AVETE VINTO UN CONCORSO!

Vero. *Overdrive SRL*, appunto, nasce come software house che sviluppa videogiochi di nuova generazione. Il nostro primo prodotto, *Hexarchia*, è un videogioco di guerra e strategia fondato sulle più importanti civiltà della

abbiamo avuto la possibilità di andare a validare l'idea di questo videogioco. Alla fine di gennaio 2020 sono state selezionate dieci start-up e, presso la Cassa di Risparmio di Firenze, si è tenuto l'evento finale di *Impresa Campus*, con start-up di vari settori, dall'alimentare al robotico. C'è stata una esibizione e il nostro videogioco fu premiato con il primo premio. Questo primo premio ci diede la possibilità, oltre che di ricevere un primo riconoscimento in denaro, di accedere ad un programma di accelerazione per start-up che cominciò nel febbraio del 2020 e si è concluso a fine ottobre del 2020. Ci fu assegnato un mentore, un ingegnere elettronico dell'azienda Nuovo Pignone che ci ha sostanzialmente fatto ripercorrere il percorso che avevamo fatto in *Impresa Campus*, ma in maniera molto più approfondita e dedicata esclusivamente a noi.

QUELLO CHE HAI LETTO E' UN ESTRATTO, L'ARTICOLO COMPLETO E' RISERVATO AGLI ABBONATI AD ELETTRONICA OPEN SOURCE.

PERCHE' ABBONARSI A PLATINUM 2.0?

UN ANNO DI **FIRMWARE 2.0**
TUTTI GLI ARTICOLI TECNICI RISERVATI
CONTEST E PROMOZIONI RISERVATI



 **VOGLIO ABBONARMI!**

+ 130.000

REGISTERED USERS

6.138

AVERAGE DAILY PAGEVIEWS (DEC2019)

824.057

2019 ANNUAL VISITORS

THE BIGGEST EMBEDDED COMMUNITY IN ITALY

CATEGORIES

COMPANIES/CONSULTANTS

53 %

ACADEMICS/STUDENTS

25 %

MAKERS/HOBBYISTS

22 %

SOCIAL CONNECTIONS

f + 83.000

in + 23.000

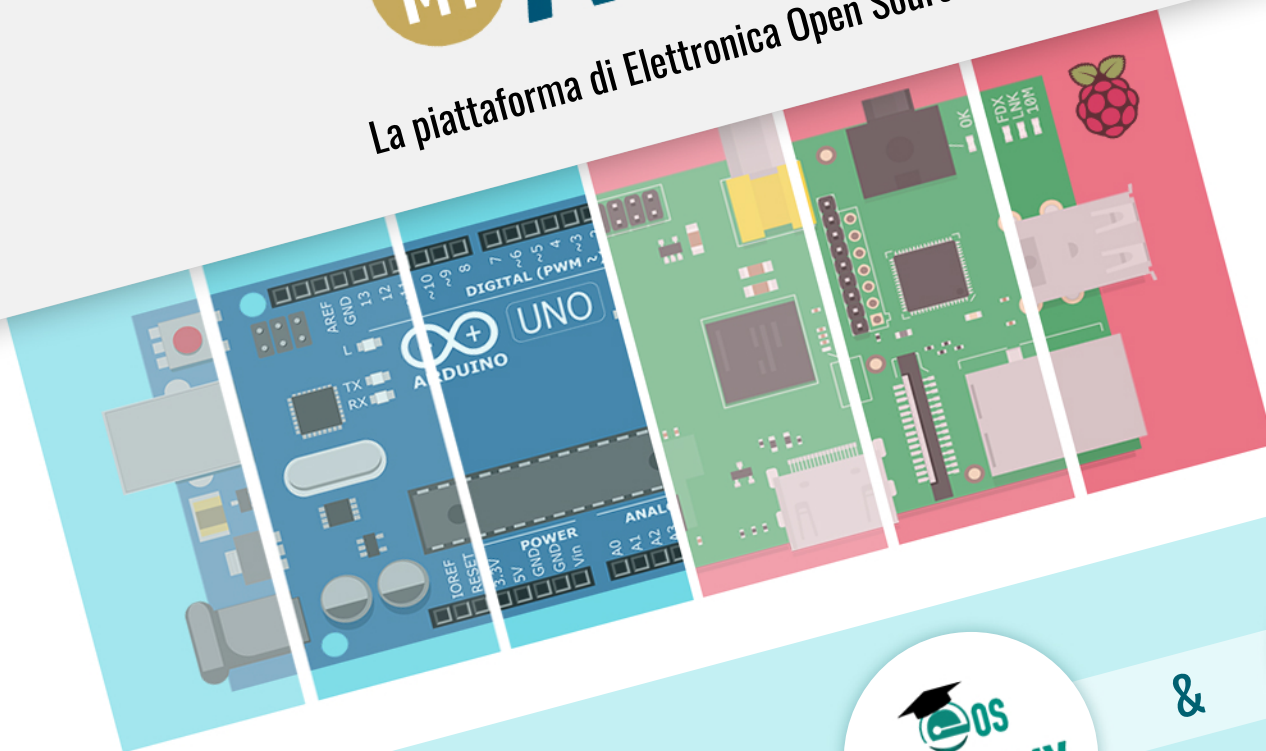


I NOSTRI CORSI DI ELETTRONICA
PER I PROFESSIONISTI
E I MAKERS



ACADEMY

La piattaforma di Elettronica Open Source dedicata ai corsi



PUOI AVERE TUTTI I CORSI DI



&



A PORTATA DI CLICK

