



GUIDO CARMASSI

ARDUINO IoT E SMART AUTOMATION

COSTRUIRE UN ALLARME CONTROLLATO DA REMOTO

Sorgenti software completi / Datasheet / Schemi e sketch di collegamento
Modello tridimensionale del case / Foto dettagliate per il setup



**PRONTO
GRAFILL**

Clicca e richiedi di essere contattato
per **informazioni e promozioni**



WEBAPP INCLUSA
CON AGGIORNAMENTO AUTOMATICO

GRAFILL

Guido Carmassi

ARDUINO IOT E SMART AUTOMATION. COSTRUIRE UN ALLARME CONTROLLATO DA REMOTO

Ed. I (12-2020)

ISBN 13 978-88-277-0198-0

EAN 9 788827 701980

Collana **COME FARE PER** (39), versione eBook



Prima di attivare Software o WebApp inclusi
prendere visione della licenza d'uso.

Inquadrare con un reader il QR Code a fianco
oppure collegarsi al link <https://grafill.it/licenza>

© **GRAFILL S.r.l.** Via Principe di Palagonia, 87/91 – 90145 Palermo
Telefono 091/6823069 – Fax 091/6823313 – Internet <http://www.grafill.it> – E-Mail grafill@grafill.it

**CONTATTI
IMMEDIATI**



Pronto GRAFILL
Tel. 091 226679



Chiamami
chiamami.grafill.it



Whatsapp
grafill.it/whatsapp



Messenger
grafill.it/messenger



Telegram
grafill.it/telegram

Edizione destinata in via prioritaria ad essere ceduta nell'ambito di rapporti associativi.

Tutti i diritti di traduzione, di memorizzazione elettronica e di riproduzione sono riservati. Nessuna parte di questa pubblicazione può essere riprodotta in alcuna forma, compresi i microfilm e le copie fotostatiche, né memorizzata tramite alcun mezzo, senza il permesso scritto dell'Editore. Ogni riproduzione non autorizzata sarà perseguita a norma di legge. Nomi e marchi citati sono generalmente depositati o registrati dalle rispettive case produttrici.



**PRONTO
GRAFILL**

**CLICCA per maggiori informazioni
... e per te uno SCONTO SPECIALE**

DEDICA E RINGRAZIAMENTI

Dedico questo libro e questo progetto al piccolo **Leo** diminutivo di Leonardo, un piccoletto che ha preso tutti i pregi della madre e qualche difetto dal sottoscritto. Un ringraziamento doveroso va anche alla moglie, in quanto mi ha sopportato durante la lavorazione del libro ma soprattutto durante la «quarantena».

Un grazie a miei genitori ed ai miei nonni, che mi hanno trasferito i valori dell'impegno e della famiglia.

Un ringraziamento particolare va anche a **Walter**, amico e collega che mi aiutato nello sviluppo e nell'ideazione del Case e nella realizzazione della Stampa 3D, del modello allegato a questo testo.

«Ogni ostacolo, ogni muro di mattoni, è lì per un motivo preciso. Non è lì per escluderci da qualcosa, ma per offrirci la possibilità di dimostrare in che misura ci teniamo. I muri di mattoni sono lì per fermare le persone che non hanno abbastanza voglia di superarlo. Sono lì per fermare gli altri.»

Randy Pausch informatico statunitense 1960 – 2008



**PRONTO
GRAFILL**

**CLICCA per maggiori informazioni
... e per te uno SCONTO SPECIALE**

SOMMARIO

➤	DEDICA E RINGRAZIAMENTI	p.	3
➤	PREMESSA	"	7
1.	MICROCONTROLLORI	"	9
2.	PANORAMA ATTUALE E SCHEDE DI SVILUPPO	"	12
	Arduino.....	"	12
	Raspberry Pi	"	13
	Teensy 4.0.....	"	14
	ESP32.....	"	15
	ESP8266.....	"	16
3.	ARDUINO UNO R3	"	18
	Alimentazione	"	19
	Ingressi analogici – AI	"	20
	Ingressi digitali – DI.....	"	21
	Che cos'è il PWM?.....	"	22
	Protocolli di comunicazione	"	22
	Che cos'è la comunicazione seriale?	"	23
	Che cos'è SPI?	"	23
	Che cos'è I2C?	"	23
	ICSP header	"	24
4.	ARDUINO MEGA 2560 R3	"	25
	Comunicazione	"	27
5.	SHIELDS	"	29
	Ethernet Shield	"	29
	Shield Relay.....	"	30

L293D Motor Drive Shield	p.	30
SIM900 GSM GPRS Shield.....	"	30
Data Logging Shield	"	31
Adafruit GPS Logger Shield	"	31
6. IDE DI SVILUPPO	"	32
7. ESEMPI ARDUINO UNO/MEGA.....	"	35
7.1. Blink	"	35
7.2. Plot seriale.....	"	37
7.3. Comunicazione Display 1602 I2C.....	"	38
7.4. Moduli RF 433 Mhz.....	"	41
8. COSTRUIAMO UN ALLARME WIRELESS.....	"	49
Premessa e disclaimer.....	"	49
9. LAYOUT ALLARME	"	53
10. CODICE ALLARME.....	"	55
11. CONTROLLO REMOTO BLYNK.....	"	62
12. ASSEMBLAMENTO COMPONENTI: MONTAGGIO CASE, SCHEDA ELETTRONICA E SETUP	"	72
13. STRUMENTI UTILI E LINK.....	"	76
14. LA WEBAPP INCLUSA.....	"	77
14.1. Contenuti della WebApp.....	"	77
14.2. Requisiti hardware e software	"	77
14.3. Utilizzo della piattaforma WebApp Grafill	"	77
14.4. Assistenza tecnica sui prodotti Grafill (TicketSystem).....	"	78

PREMESSA

Questo libro, non vuole essere un testo per tecnici esperti ma più che altro un esempio di quello che si può fare con l'attuale tecnologia, un po' di pazienza e un po' di creatività.

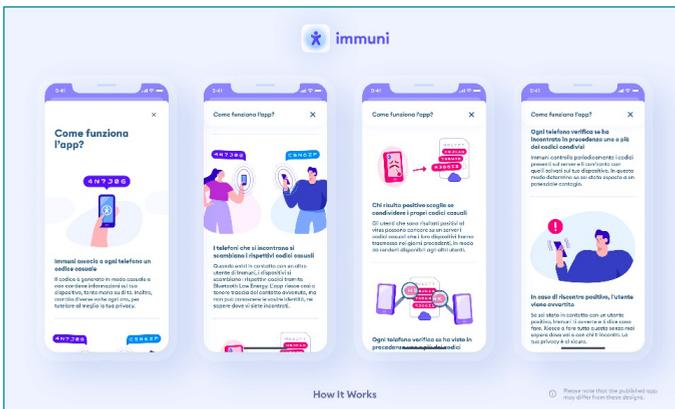
Il testo purtroppo viene redatto in un momento non facile per l'Italia e per il Mondo. Nel momento in cui scrivo siamo in quarantena bloccati a casa per ridurre quanto prima i contagi dal Virus (Covid-19), che ha ucciso non solo migliaia di persone, ma un po' anche le nostre libertà.

Il mondo dopo la Pandemia, non sarà più lo stesso.

Ecco perché ora più che mai c'è bisogno che la tecnologia sia smart e ci aiuti insieme alla medicina, alla Scienza ed alla civile responsabilità di ognuno, a sconfiggere questo «nemico» invisibile.

La Corea del Sud, il Giappone, la Cina ed in prima linea anche l'Italia (con l'App Immuni), stanno sviluppando software dedicati al controllo ed al contenimento dei contagi causati dal Coronavirus.

Le App sfruttano: GPS, Bluetooth, Cloud, Banche dati, Mappe Interattive e molto altro ancora. Questo è il mondo dell'**IoT** (*Internet of Things*), letteralmente l'Internet delle cose.



App Immuni



**PRONTO
GRAFILL**

**CLICCA per maggiori informazioni
... e per te uno SCONTO SPECIALE**

MICROCONTROLLORI

I **microcontrollori** sono presenti all'interno di un gran numero di dispositivi elettronici, dal forno a microonde passando per TV e smartphone e giocattoli.



Figura 1.1. Microcontrollore ATMEL ATMEGA328 montato sulla scheda di sviluppo ARDUINO UNO

Un microcontrollore (MCU) utilizza **tecnologie produttive di miniaturizzazione** per ridurre in un package la CPU (*Central Processing Unit*) e la memoria. È dotato di diversi pin di ingresso e uscita, attraverso i quali è possibile interfacciarsi con diverse periferiche: porte d'ingresso, porte di uscita, memorie, porte seriali, convertitori analogico-digitali, temporizzatori, contatori di eventi, controllori di interruzione, DMA, etc..

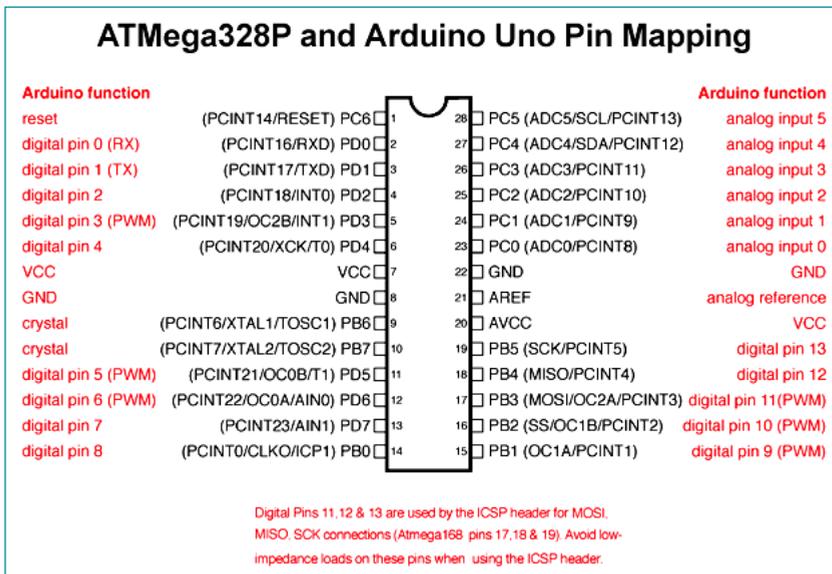


Figura 1.2. PIN OUT Microcontrollore ATMEL ATMEGA328P

PANORAMA ATTUALE E SCHEDE DI SVILUPPO

Il panorama attuale delle schede di sviluppo è veramente vasto. Di seguito, due delle schede di sviluppo più famose seguite da altre in continua crescita ed evoluzione.

Arduino

Arduino è una società hardware e software open source tutta italiana. Il nome Arduino deriva da un bar di Ivrea, in Italia, dove si incontravano alcuni dei fondatori del progetto.

Il progetto è iniziato nel 2005 come programma per gli studenti dell'*Interaction Design Institute Ivrea*, di Ivrea, con l'obiettivo di fornire un modo semplice ed economico per i principianti ed i professionisti per creare dispositivi in grado di interagire utilizzando sensori e attuatori.

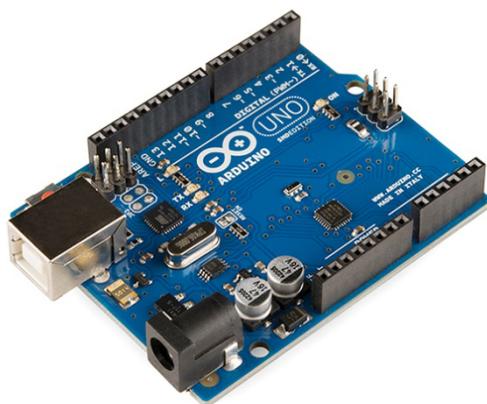
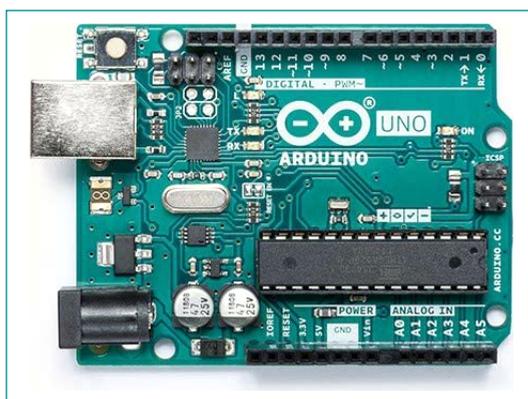


Figura 2.1. Arduino UNO R3

I prodotti della azienda di Ivrea sono concessi sotto licenza GNU Lesser General Public License (LGPL) o GNU General Public License (GPL), e questo permette a chiunque la fabbricazione di schede Arduino e la distribuzione del software.

ARDUINO UNO R3

Cerchiamo di capire com'è fatta la scheda, com'è possibile collegarla, quali sono le interfacce di comunicazione, i pin di alimentazione e i limiti di utilizzo.

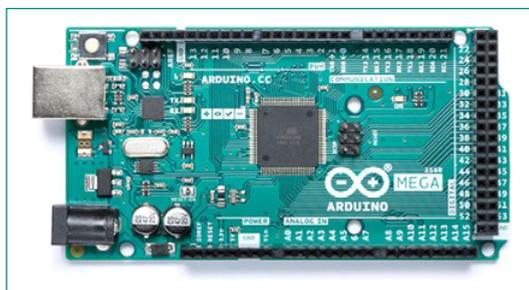


Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

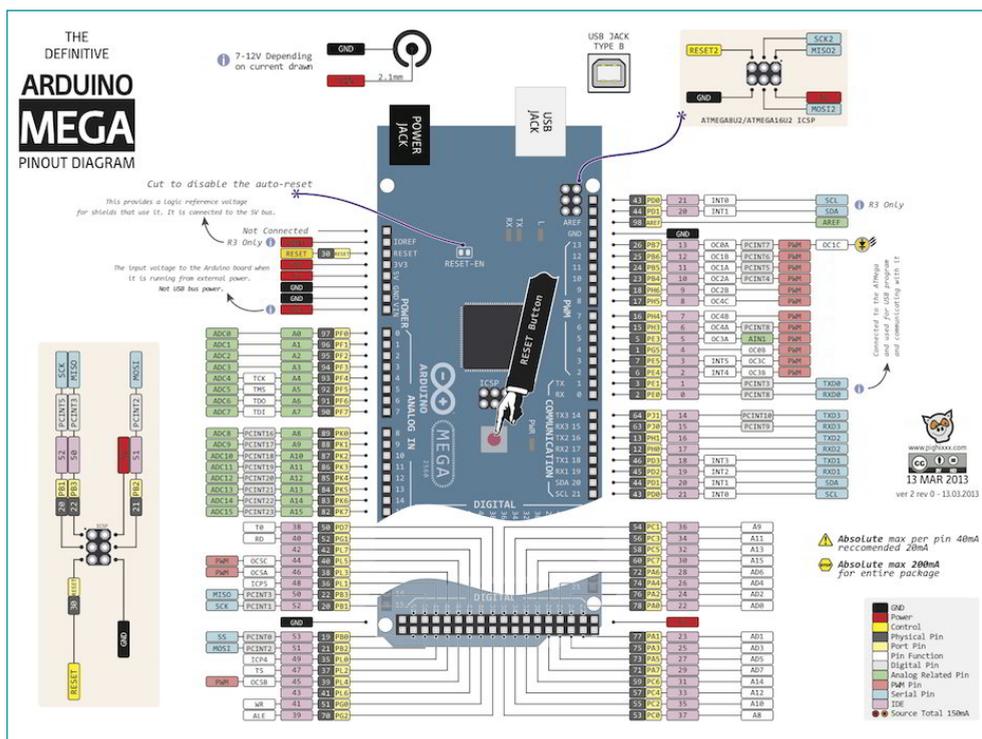
ARDUINO MEGA 2560 R3

L'ATmega2560 ha 256 KB di memoria flash per l'archiviazione del codice (di cui 8 KB utilizzati per il bootloader), 8 KB di SRAM e 4 KB di EEPROM (che possono essere letti e scritti con la [libreria EEPROM](#)).

La differenza con la scheda Arduino Uno è notevole, soprattutto in termini di programmazione.



Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
LED_BUILTIN	13
Length	101.52 mm
Width	53.3 mm
Weight	37 g



Ognuno dei 54 pin digitali su Mega può essere utilizzato come input o output, utilizzando le [funzione pinMode\(\)](#), [digitalWrite\(\)](#) e [digitalRead\(\)](#). Operano a 5 volt. Ogni pin può fornire o ricevere 20 mA come condizione operativa consigliata ed ha una resistenza di pull-up interna (scollegata per impostazione predefinita) di 20-50 k ohm. Un valore massimo di 40 mA è il valore che non deve essere superato per evitare danni permanenti al microcontrollore.

Inoltre, alcuni pin hanno funzioni dedicate così come per la UNO:

- **Seriale 0:** Pin 0 (RX) e 1 (TX); **Seriale 1:** Pin 19 (RX) e 18 (TX); **Seriale 2:** Pin 17 (RX) e 16 (TX); **Seriale 3:** Pin 15 (RX) e 14 (TX). Il pin per ricevere è ovviamente (RX) e per trasmettere (TX) dati seriali TTL. I pin 0 e 1 sono anche collegati ai pin corrispondenti del chip seriale da USB a TTL dell'ATmega16U2.
- **Interrupt** esterni: Pin 2 (interrupt 0), Pin 3 (interrupt 1), Pin 18 (interrupt 5), Pin 19 (interrupt 4), Pin 20 (interrupt 3) e Pin 21 (interrupt 2). Vedere la funzione [attachInterrupt\(\)](#) per i dettagli.
- **PWM:** dal Pin 2 a 13 e dal Pin 44 a 46. La scheda fornisce un'uscita PWM a 8 bit con la funzione [analogWrite\(\)](#).
- **SPI:** Pin 50 (MISO), Pin 51 (MOSI), Pin 52 (SCK), Pin 53 (SS). Questi pin supportano la comunicazione SPI utilizzando la [libreria SPI](#). I pin SPI

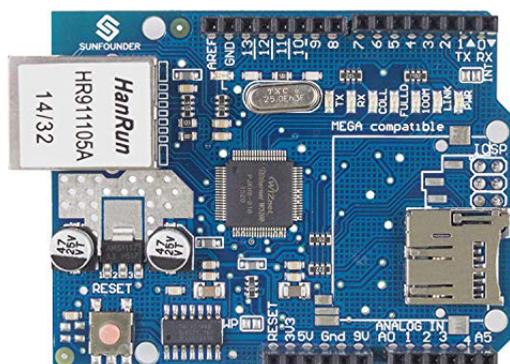
SHIELDS

Gli shields compatibili con Arduino sono veramente tanti e qui di seguito ne farò una carrellata dei più usati, ma questa non vuole essere una lista definitiva, in quanto ve ne sono di svariate marche e con le più disparate funzionalità.

Gli shields sono componenti hardware predisposti per essere montati direttamente sopra Arduino ed in grado di fornire specifiche funzionalità. Ad esempio, si puoi utilizzare uno Shield per semplificare il controllo dei motori od uno Shield Ethernet per connettere Arduino alla rete. Molti Shields hanno librerie dedicate che vengono richiamate nel Software di sviluppo. La maggioranza è compatibile sia con la versione di Arduino UNO che la Mega 2560. Si possono inoltre sovrapporre più Shields uno su l'altro.

Ethernet Shield

Grazie allo Shield Ethernet, Arduino può acquisire le funzionalità di client o server. Lo shield Ethernet è inoltre dotato di uno slot per schede SD per l'archiviazione dei file per il servizio in rete. Sarà possibile creare così, un "mini" Server WEB, o fornire alla scheda la connessione Internet, magari per il controllo remoto. La scheda si basa sul chip ethernet Wiznet W5100 ed è in grado di supportare sia il protocollo TCP che UDP. Supporta fino a quattro connessioni socket simultanee.



ESEMPI ARDUINO UNO/MEGA

7.1. Blink

Il codice di esempio che abbiamo appena caricato sulla scheda, fa lampeggiare il Led integrato all'accensione della stessa, con un timer impostato di 1 secondo (1000 ms). Il codice è molto semplice, ma ci è utile per capire la struttura ed alcune delle principali funzioni del linguaggio di programmazione usato.

Codice di esempio Blink

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}
```

Void Setup()

La funzione setup () viene chiamata all'avvio dello sketch e viene solitamente utilizzata per inizializzare variabili, configurare la modalità dei pin, configurare le librerie, ecc.. La funzione di Setup verrà eseguita una sola volta, dopo ogni accensione o reset della scheda Arduino.

Void loop()

La funzione loop() permette di eseguire tutto il codice posto al proprio interno, come dice la parola stessa in loop, quindi ciclicamente e consecutivamente.

COSTRUIAMO UN ALLARME WIRELESS

Premessa e disclaimer

Quello che si andrà a descrivere nelle prossime pagine ha puro scopo illustrativo ed educativo.

I sistemi antifurto sono impianti ben più complessi e dedicati alla protezione di beni e/o persone, sulle quali è necessario porre la massima attenzione.

Solo tecnici esperti nonché ditte od aziende abilitate possono certificarne la corretta fornitura, installazione e funzionalità.

Il sottoscritto e la casa editrice non rispondono di un eventuale uso improprio di tali informazioni atte alla costruzioni di apparecchi antifurto e declina ogni responsabilità derivata e connessa.

Vediamo come poter sfruttare le potenzialità di ARDUINO ed alcuni Shields appena descritti per poter costruire un sistema d'allarme completo senza fili tramite l'uso di sensori RF 433, facilmente reperibili in commercio. Molti di questi sensori, funzionano a batterie e sono di semplice installazione.

L'idea è quella di captare il segnale proveniente da uno o più sensori installati attivando una sirena.



Esempio di sensori magnetici e di movimento (PIR)

LAYOUT ALLARME

Quello che segue è uno Sketch «*semplificato*» di quello che potrebbe essere un piccolo impianto d'allarme configurabile con più sensori 433 Mhz ed allo stesso tempo dotato di connessione Internet, per notifiche su cellulare e controllo remoto attraverso App dedicata.

Quello che vi servirà per realizzare tale Sketch è:

- 1 x Modulo Arduino Mega 2560
- 1 x Shield Ethernet W5100
- 1 x Display 1602 I2C
- 1 x Modulo RF RXB6
- 1 x Relè 5V tipo KY-019 o similare
- 1 x Sirena 9-12Vdc
- 1 Pila 9Vdc
- 1 Alimentatore 9Vdc
- 1 Antenna (accordata per 433Mhz)

Il suddetto materiale può essere facilmente trovato sul web ed a costi veramente bassi. Sarà necessario poi acquistare i sensori magnetici per porte e finestre, i sensori PIR ed i telecomandi per il controllo. Il progetto allegato funziona con 5 sensori Magnetici a contatto REED 433Mhz e 2 Pirottici. Ci sono poi 2 telecomandi per l'inserimento a distanza dell'allarme.



Telecomandi 433Mhz



Sensori 433Mhz – MC-03b



Pir (doppia tecnologia)
433 Mhz

CODICE ALLARME

Una volta costruito lo Sketch potrete cominciare a sperimentare ed ha effettuare prove sull'apprendimento dei segnali e dei dispositivi che successivamente dovrete utilizzare nell'impianto d'Allarme.

Con questo codice semplicemente connettendo il display e la ricevente potrete mettervi in ascolto dei segnali 433 Mhz, visualizzandoli direttamente sul monitor seriale e sul display.

Codice Ricevente Allarme Step 1 (Allegato→Allarme_Step1):

```
//***** Ing. Guido Carmassi*****
// Librerie aggiunte
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <RCSwitch.h>

// Costanti
RCSwitch mySwitch = RCSwitch();

int value; // Variabile che indica il segnale inviato dai sensori radio e/o dai telecomandi

LiquidCrystal_I2C lcd(0x27, 16, 2); // Inizializza Display

void setup() {
  Serial.begin(9600);
  // initialize the LCD
  lcd.init();
  // Turn on the backlight and print a message.
  lcd.backlight();
  lcd.print("Allarme Receiver");
  delay(1500);
  lcd.clear();
  mySwitch.enableReceive(0); // La ricevente è collegata al pin INTERRUPT 0, cioè il pin
  digitale 2
```

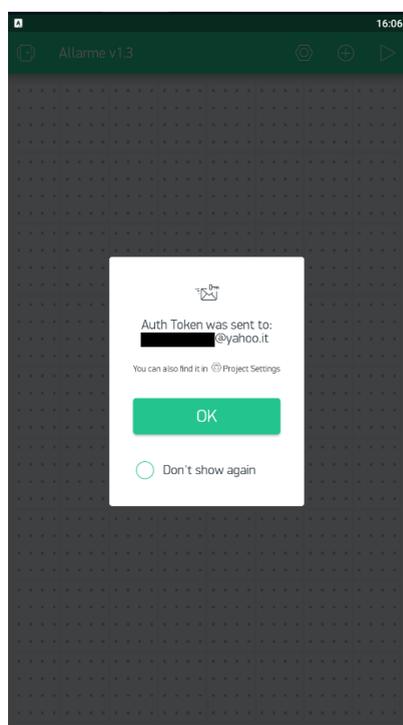
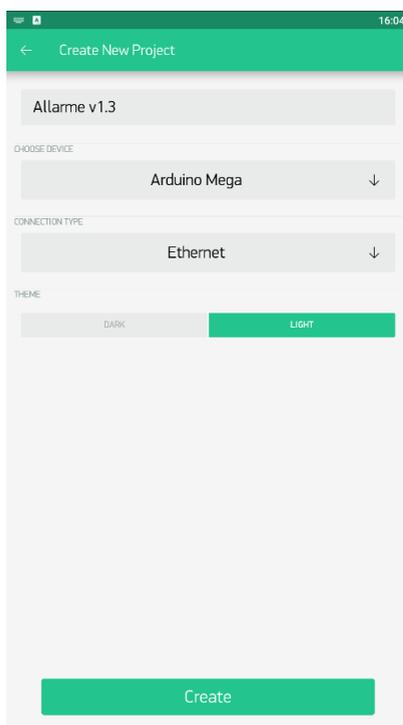
CONTROLLO REMOTO BLYNK

L'ultimo step del software permette alla scheda di comunicare con la rete e con la piattaforma di sviluppo Blynk (<https://blynk.io/>). Blynk combina una piattaforma cloud con applicazioni che permettono la fruizione e la gestione dei dati ovunque tramite un'applicazione per SmartPhone Android o IOS per oltre 400 moduli Hardware tra i quali ARDUINO.

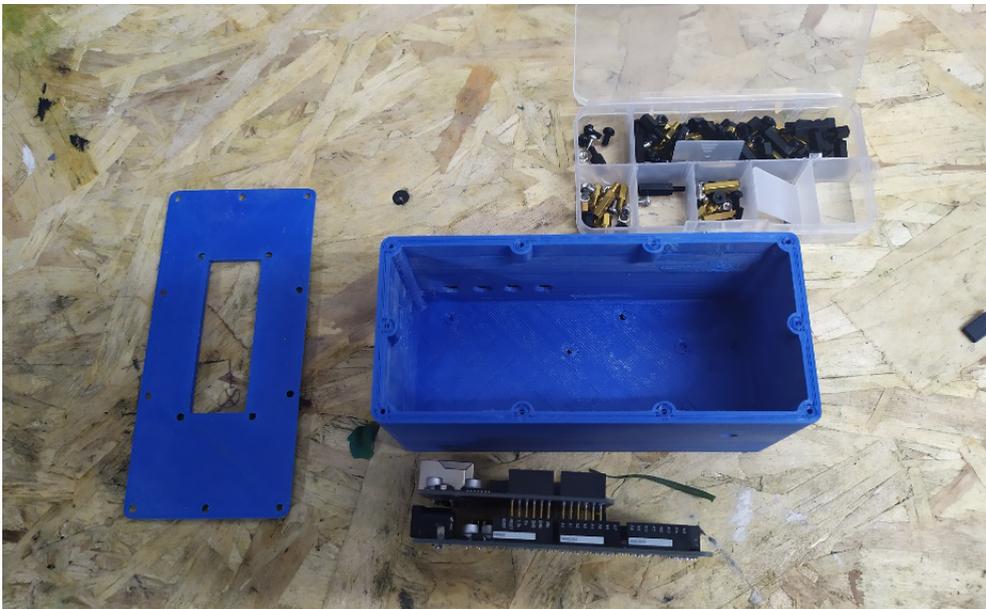
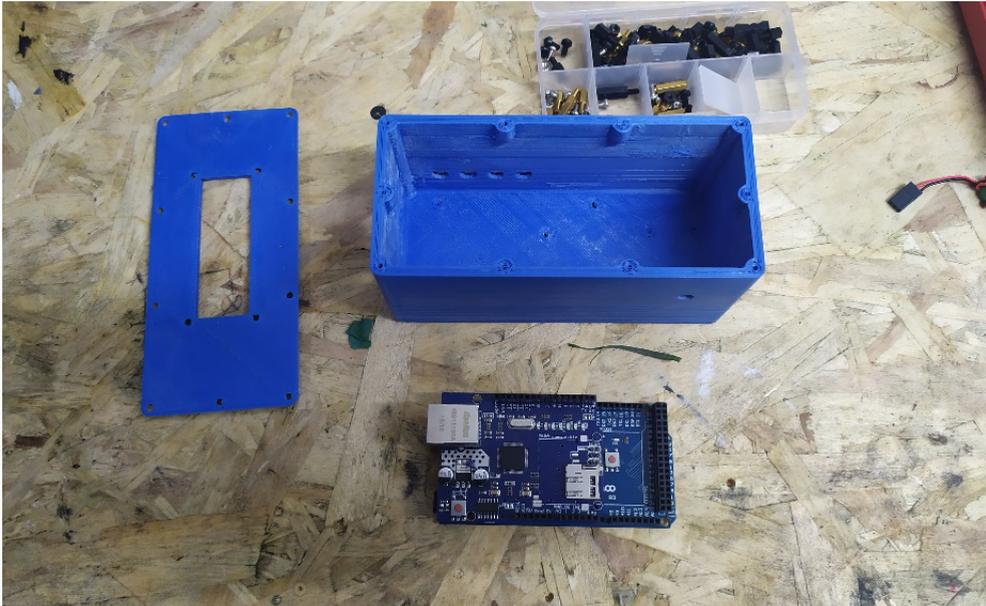
Link hardware compatibile

<https://github.com/blynkkk/blynkkk.github.io/blob/master/SupportedHardware.md>

Installata l'APP Blynk, seguire pochi semplici passi per costruire una APP che comunica con la scheda ARDUINO, rendendo disponibile la comunicazione e la gestione dei dati con il mondo mobile e non solo. Una volta che ci siamo registrati all'applicazione, creiamo un nuovo progetto, scegliamo la scheda di sviluppo Arduino mega e la connessione ethernet, poi premiamo su create.



ASSEMBLAMENTO COMPONENTI: MONTAGGIO CASE, SCHEDA ELETTRONICA E SETUP



LA WEBAPP INCLUSA

14.1. Contenuti della WebApp

La **WebApp inclusa** gestisce le seguenti utilità:

- Datasheet dettagliati Arduino Mega ed UNO.
- Foto montaggio della scheda elettronica dentro il CASE Stampato 3D.
- Librerie Arduino IDE. Le librerie sono quelle utilizzate per lo sviluppo dei progetti allegati al libro e, per evitare problemi di incompatibilità con la componentistica, possono essere importate direttamente nella cartella *Documents\Arduino\libraries*.
- Modello 3D pronto per la stampa 3D del CASE.
- Codice Sorgenti allegati al libro e divisi a step per l'ambiente di programmazione ARDUINO IDE.
- Sketch progetto realizzato con il software <https://fritzing.org/>.

Grazie ai **sorgenti software completi**, sarà possibile costruire: una ricevente 433 Mhz; un programmatore 433 Mhz; una centralina d'allarme; una vera e propria App per il controllo remoto della centralina attraverso Smartphone.

14.2. Requisiti hardware e software

- Dispositivo con MS Windows, Mac OS X, Linux, iOS o Android.
- Accesso ad internet e browser web con Javascript attivo.
- Software per la gestione di documenti Office e PDF.

14.3. Utilizzo della piattaforma WebApp Grafill

WebApp Grafill è la piattaforma per l'acquisto di eBook Grafill Editoria tecnica con modalità *on demand*. La piattaforma consente di acquistare ed attivare in tempo reale eBook, software e/o raccolte di documenti editabili.

Attraverso il *Tool di assistenza*, disponibile su tutti i prodotti, è possibile ottenere assistenza tecnica e supporto.

Per accedere alla piattaforma **WebApp Grafill** e alla **WebApp** parte integrante della presente pubblicazione procedere come segue:

- 1) Collegarsi al seguente indirizzo internet:

<https://webapp.grafill.it>

- 2) Cliccare su **[Accedi]**.
- 3) Inserire *Username* e *Password* dell'account **grafill.it** che hai utilizzato per l'acquisto dell'ebook.
- 4) Nella sezione **LE MIE APP** cliccare sulla copertina del prodotto acquistato e procedere all'uso della WebApp.

14.4. Assistenza tecnica sui prodotti Grafill (TicketSystem)

L'assistenza tecnica (gratuita per 365 giorni dall'acquisto) è relativa esclusivamente all'installazione e all'avvio del prodotto (*non è prevista assistenza per il recupero dei dati*), se la configurazione hardware rispetta i requisiti richiesti.

L'assistenza *TicketSystem* è disponibile all'indirizzo **<https://www.supporto.grafill.it>**.

Effettuare il login al *TicketSystem* utilizzando i dati del profilo utente di **www.grafill.it** ed aprire un ticket seguendo le istruzioni.

La cronologia dei ticket resterà disponibile sulla schermata principale del *TicketSystem*.

