

ALESSIO BALBO

Progettista Embedded  
Develer

# Rendiamo Smart una macchina del caffè

Develer webinar

19/05/2021

develer



# I Sommario

- I Scopo del progetto
- I Scelta architettura
- I Selezione dei componenti elettronici
- I Configurazione Arduino
- I Uso del Bot di Telegram

# SCOPO DEL PROGETTO

Il progetto nasce per unire:

- Estetica di un prodotto vintage
- Accuratezza nel controllo della temperatura di estrazione
- Comodità del controllo remoto di un prodotto Smart

# La macchina – Scopo del progetto



Feb. 16, 1960

E. SALATI

2,925,027

COFFEE INFUSION PREPARING MACHINE WITH HYDROCOMPRESSOR

Filed April 11, 1957

3 Sheets-Sheet 1

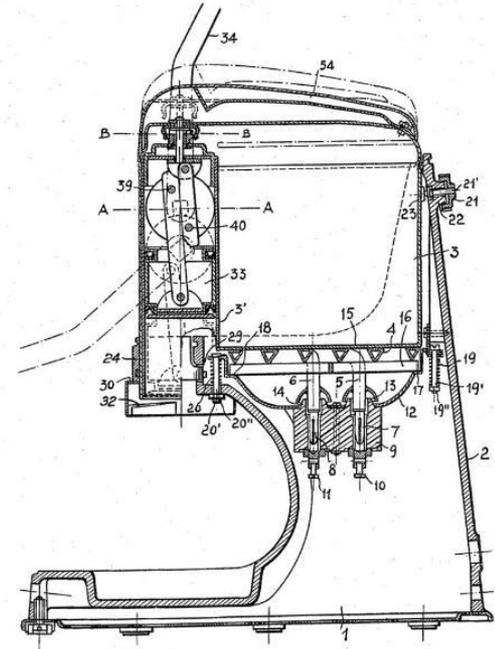


Fig. 1

Scopo del progetto

# Controllo della temperatura di estrazione

Termostato originale  
lamella bimetallica

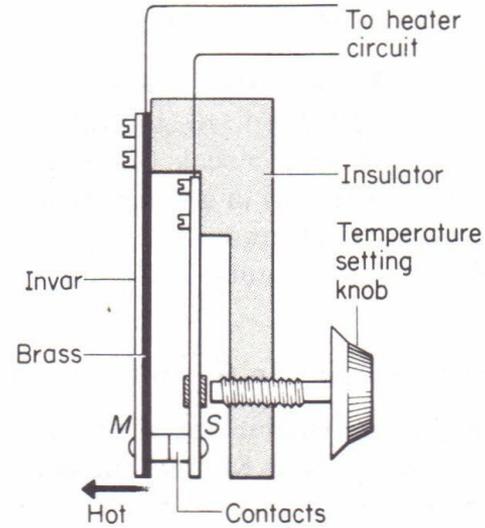
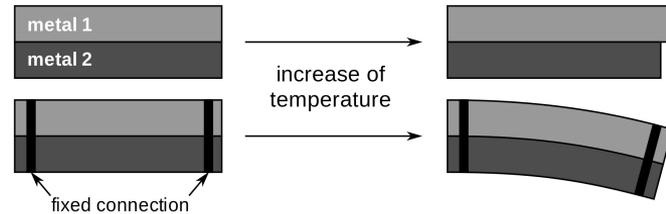


Fig. 15.6. Thermostat



Scopo del progetto

# Controllo della temperatura di estrazione

Requisiti del controllo di temperatura

Requisiti per il controllo di temperatura:

- Possibilità di regolare la temperatura tra 89°C e 95°C
- Mantenimento della temperatura
- Regolazione di temperatura “nascosta” all’utente
- Controllo PID

Scopo del progetto

## Funzioni smart

Migliorare la comodità dell'utente

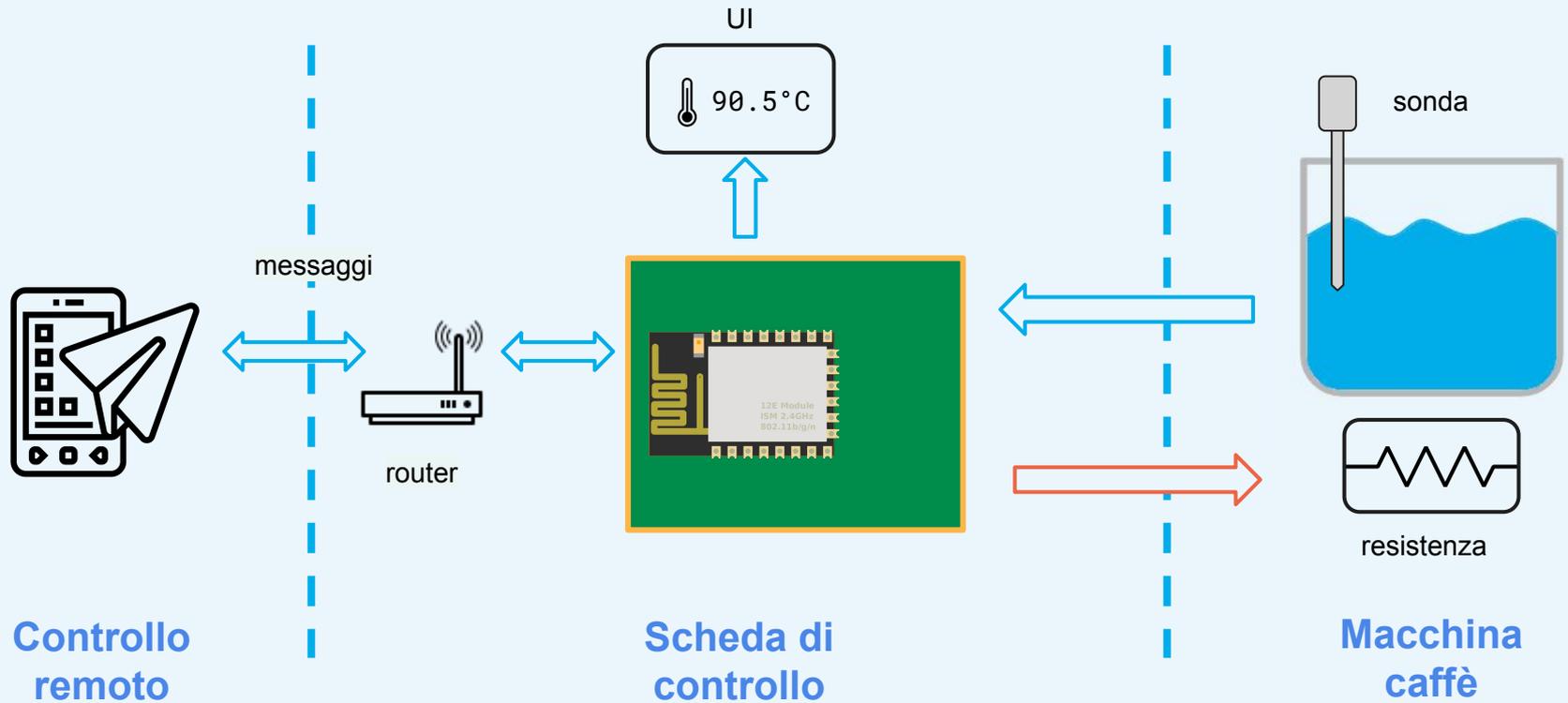
Controllo remoto per:

- Accensione
- Preriscaldamento
- Impostazione della temperatura
- Configurazione PID

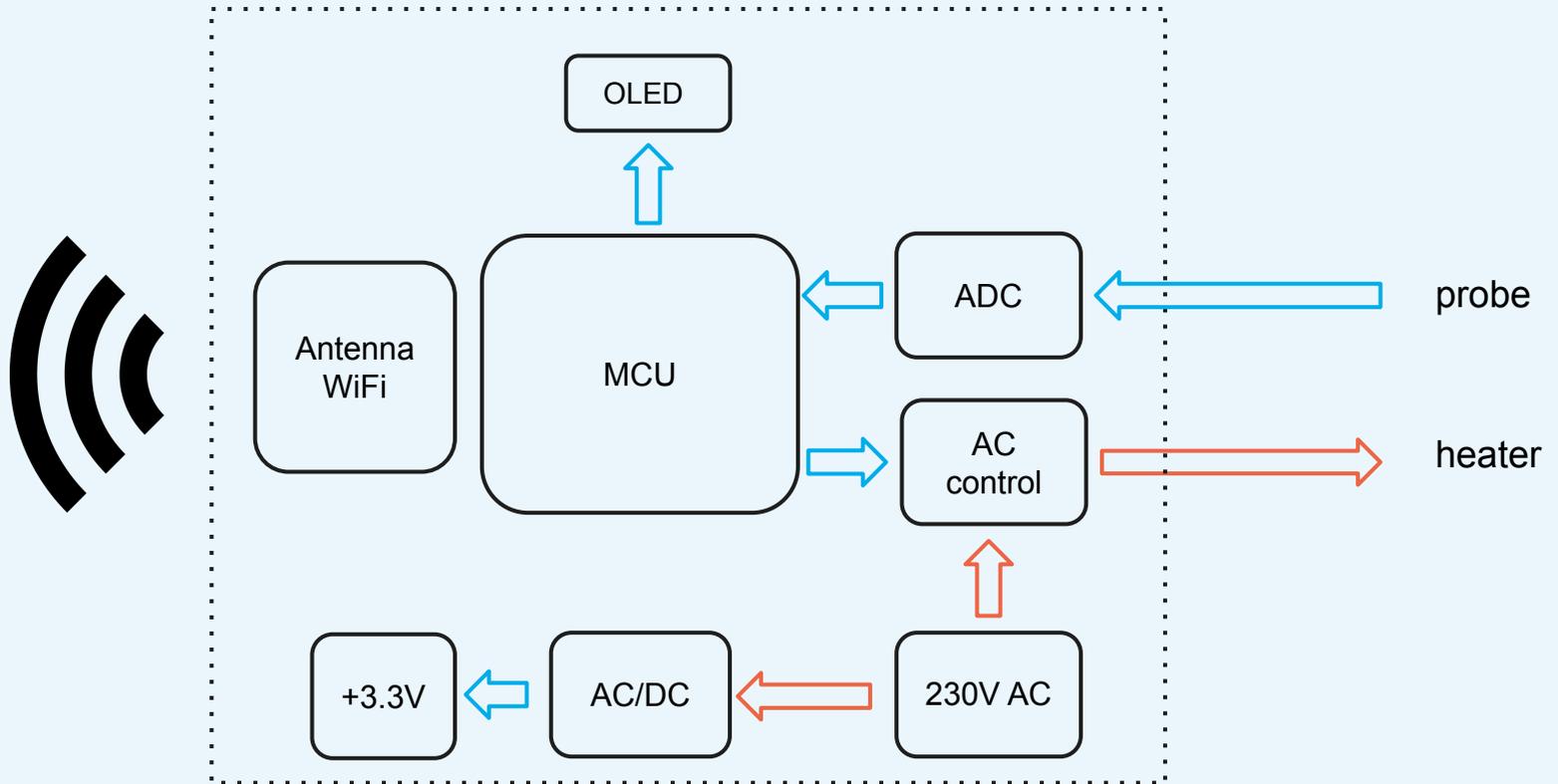
# SCELTA ARCHITETTURA

- | Architettura di sistema
- | Architettura della scheda elettronica

# Schema a blocchi del sistema



# Schema a blocchi della scheda



# Connettività: perché WiFi?



Perché è stato scelto il WiFi:

- Utilizzo domestico
- Device sempre connesso
- Messaggi da più utenti

Scelta architettura

# Perché Telegram?



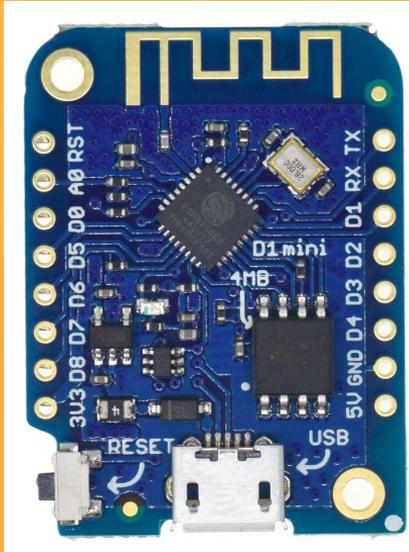
- | App di messaggistica diffusa
- | API Bot dettagliata
- | Configurazione iniziale guidata

# SELEZIONE COMPONENTI

- | MCU WiFi
- | ADC
- | Sonda temperatura
- | Controllo resistenza
- | Display OLED

Selezione componenti

## Scheda WiFi

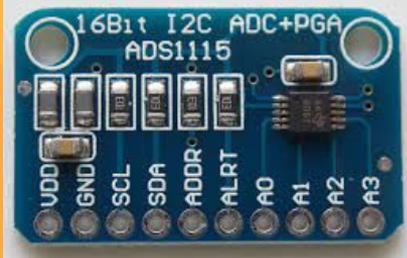


### Wemos LOLIN D1 mini

- MCU: Espressif ESP8266
- Costo basso
- Compatibile con Arduino
- Disponibilità di librerie e supporto
- Disponibilità di vari shield

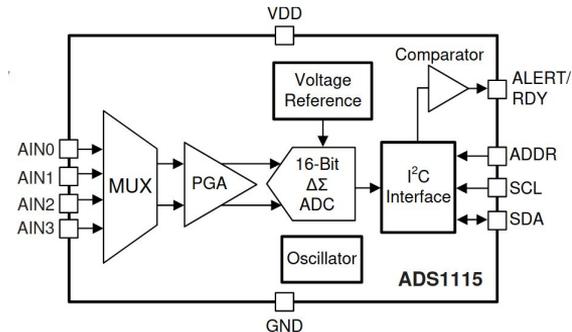
Selezione componenti

## ADC esterno



### TI ADS1115

- ADC esterno 16bit
- Interfaccia I2C
- 4x canali singoli o 2x differenziali
- Vref interna



Selezione componenti

# Sensore di temperatura

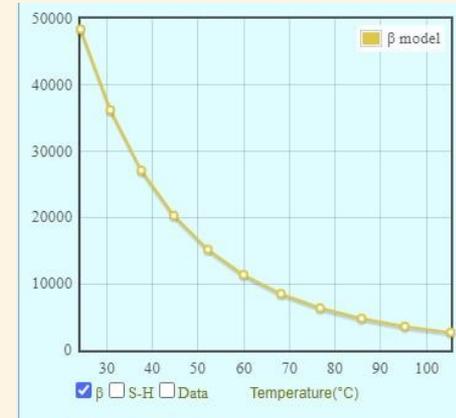
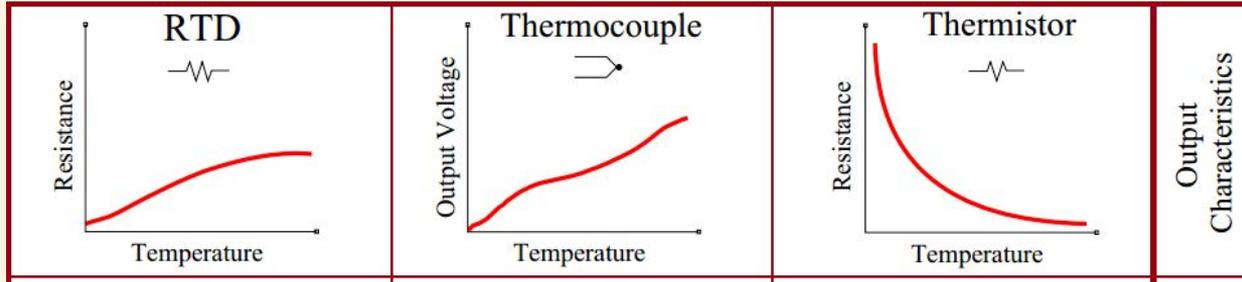
## Requisiti

- Range di misura 0 - 100°C
- Cavo resistente alle alte temperature
- Rimovibile dalla macchina
- Accuratezza 0.5°C

## Sensori di temperatura

Sensore	Range tipico [°C]	Accuratezza [°C]	Pro	Con
Termistore (NTC)	-55÷150	0.1÷1.5	<ul style="list-style-type: none"><li>• Stabilità</li><li>• Media responsività</li><li>• Costo</li></ul>	<ul style="list-style-type: none"><li>• Risposta esponenziale</li><li>• Range</li></ul>
RTD (PT100)	-200÷600	0.1÷1	<ul style="list-style-type: none"><li>• Stabilità</li><li>• Range</li><li>• Linearità</li></ul>	<ul style="list-style-type: none"><li>• Lento</li></ul>
Termocoppia (K)	-200÷1000	>2	<ul style="list-style-type: none"><li>• Temperature elevate</li><li>• Veloce</li></ul>	<ul style="list-style-type: none"><li>• Complessità di lettura</li><li>• Giunzione fredda</li><li>• Drift</li></ul>
Digitale (DS18B20)	-25÷125	0.5÷2	<ul style="list-style-type: none"><li>• Semplicità di lettura (I2C/1W)</li></ul>	<ul style="list-style-type: none"><li>• Range</li><li>• Solitamente si usa per T ambiente</li></ul>

# Sensori di temperatura



# Sonda di temperatura



## Sonda selezionata

- Termistore NTC
- $R_0 = 47k\Omega$ ,  $\beta=4030$
- Accuratezza  $\pm 1^\circ\text{C}$  (0-100 $^\circ\text{C}$ )
- Cavo metallo (fino 300 $^\circ\text{C}$ )
- Connettore jack audio 3.5mm

Selezione componenti

# Lettura sensore

Circuito base

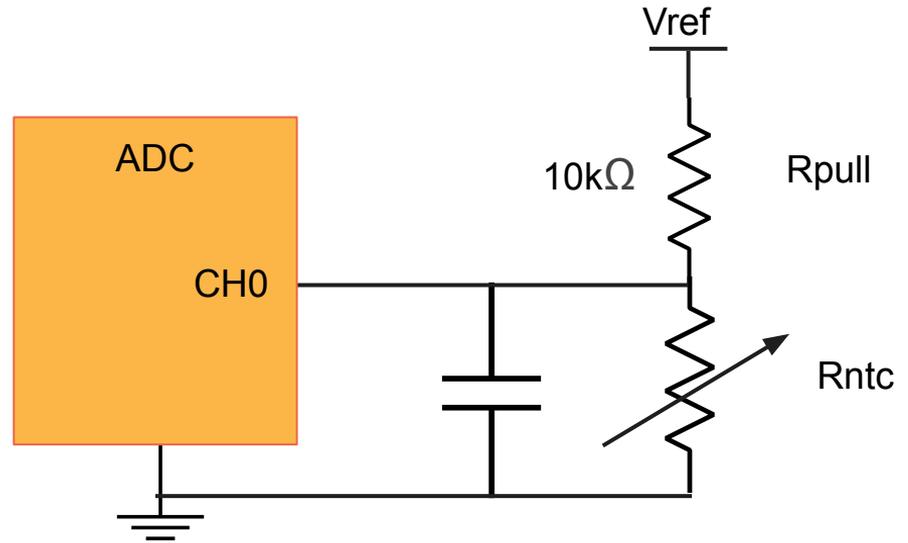
Pull-up dimensionato per dinamica ADC

Con  $R_{pull} = 10k\Omega$ :

@25°C ->  $R = 47k\Omega$  ->  $V_{adc} = 2.72V$

@85°C ->  $R = 4.9k\Omega$  ->  $V_{adc} = 1.08V$

@100°C ->  $R = 3.1k\Omega$  ->  $V_{adc} = 0.78V$



# Controllo AC resistenza

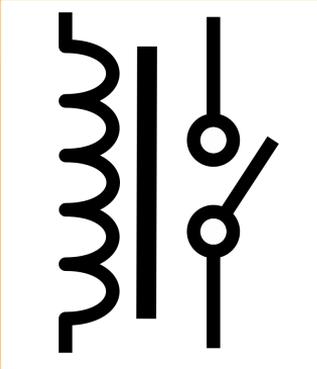
## Specifiche resistenza:

- Resistenza 103Ω
- Potenza dissipata 500W (2Arms)
- Resistenza non immersa

Selezione componenti

## Controllo AC resistenza

Relè elettromeccanico



Primo prototipo -> con **relay shield** della wemos.

Pro:

- Costo basso
- Semplicità di controllo
- Dimensioni (15x19mm)

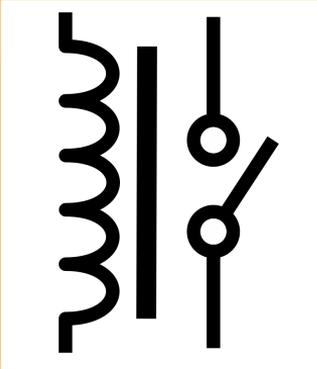
Con:

- Lento, periodo di controllo almeno 10/20 secondi
- Rumoroso, fa click ad ogni commutazione

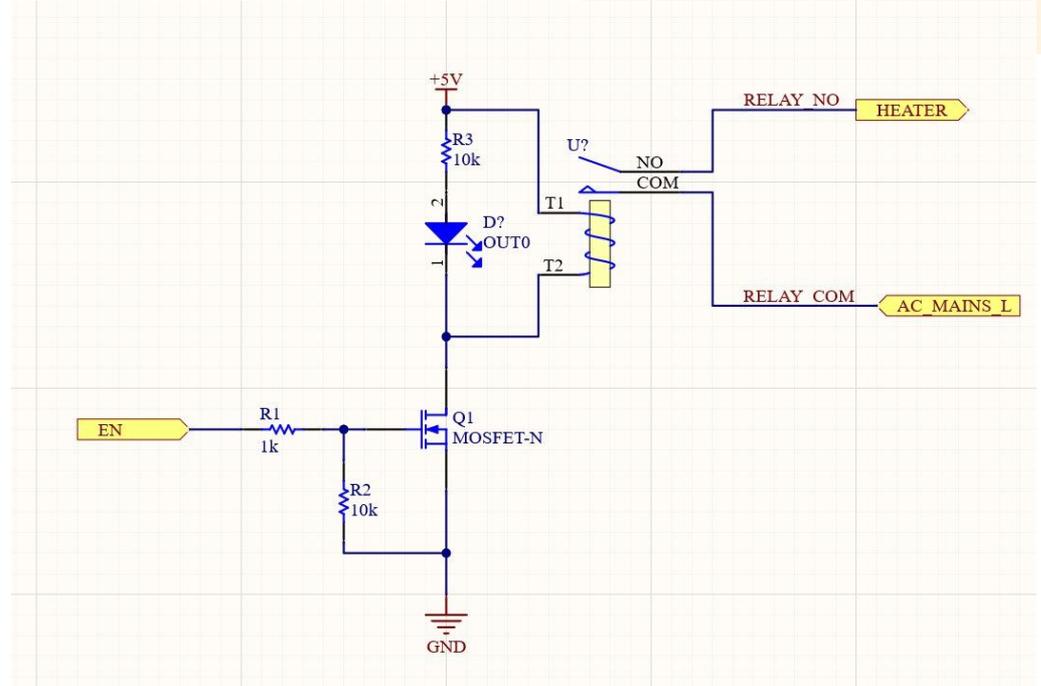
Selezione componenti

# Controllo AC resistenza

Relè elettromeccanico

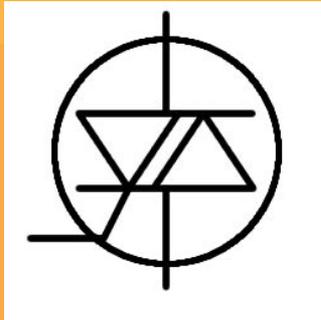


## Circuito di controllo del relè



# Controllo AC resistenza

Triac



In una seconda fase viene dimensionato un Triac adeguato per la resistenza della macchina (**500W**->**2Arms**).

Pro:

- Veloce
- Silenzioso
- Modalità di controllo avanzate (taglio di fase o zero-cross)

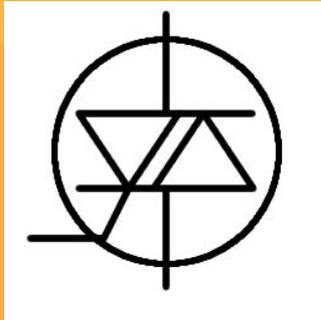
Con:

- Complessità del circuito di controllo
- Fototriac per isolamento
- Dissipazione termica

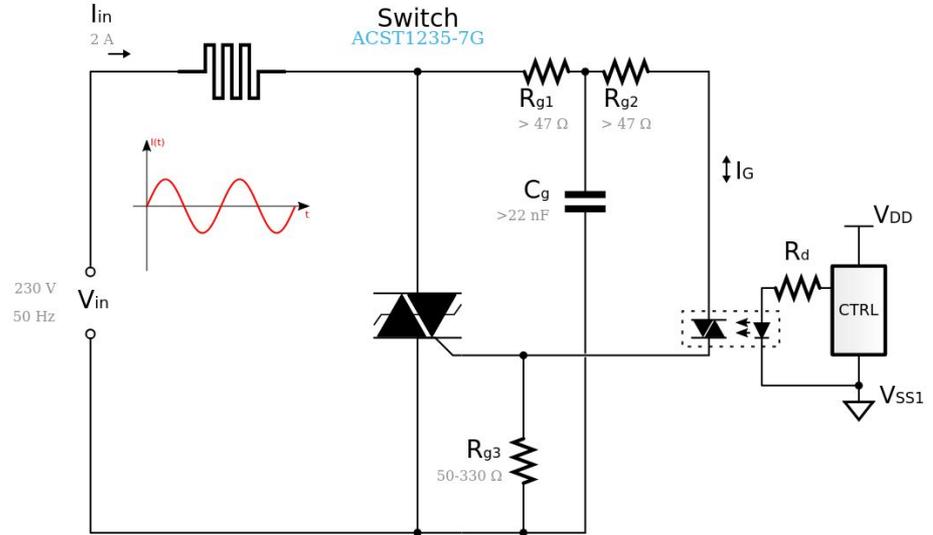
Selezione componenti

# Controllo AC resistenza

Triac



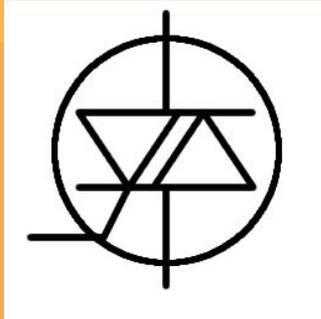
Circuito di controllo Triac:



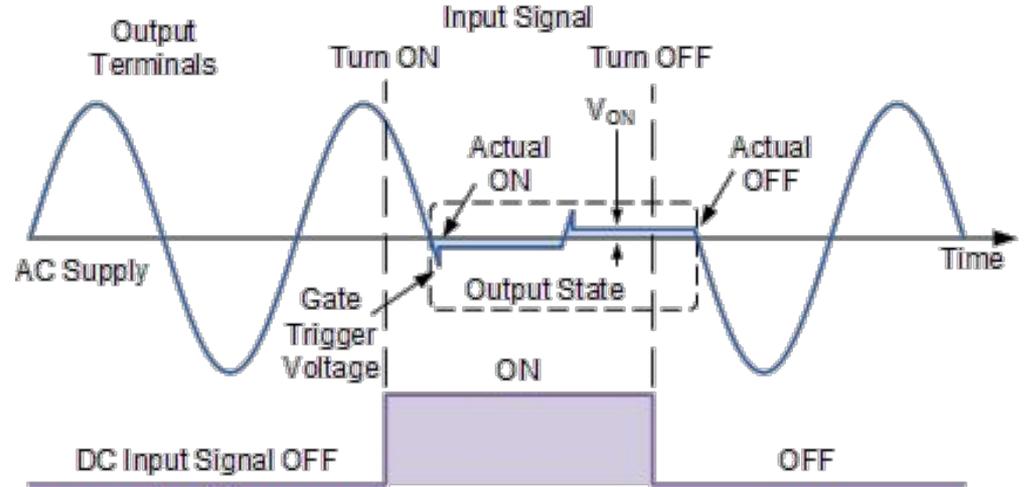
Selezione componenti

# Controllo AC resistenza

Triac



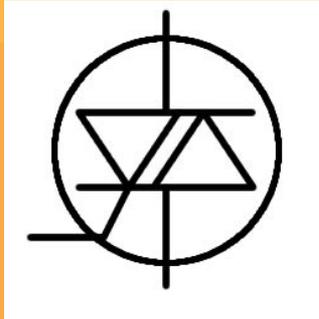
## Algoritmo zero-crossover switching



Selezione componenti

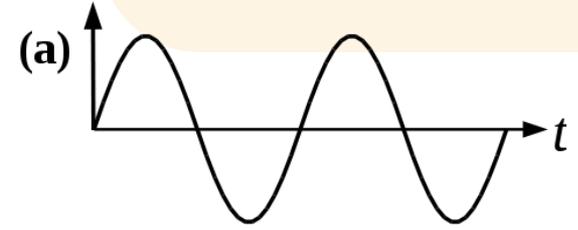
# Controllo AC resistenza

Triac

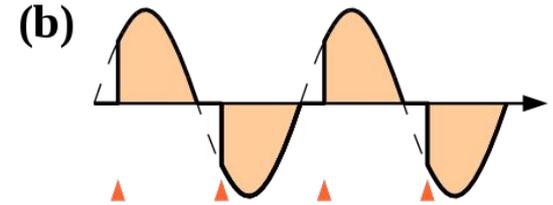


## Algoritmo in taglio di fase

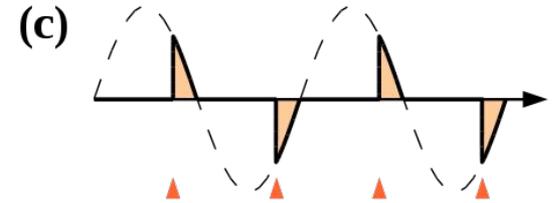
Tensione di rete:



Taglio di fase 75%



Taglio di fase 25%



Selezione componenti

## Controllo AC resistenza

Solid State Relay (SSR)



### Alternativa: SSR

- Integra Triac e circuito di controllo
- Input digitale
- Fissaggio dissipatore
- Cons: costo e dimensioni

# Display OLED

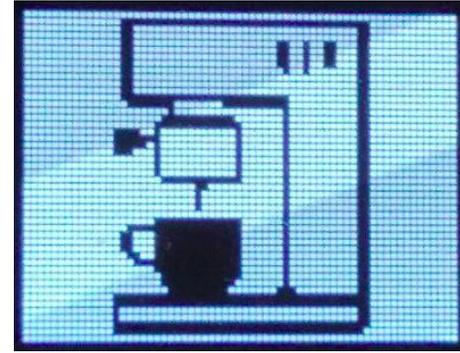


## Shield OLED 0.66in

- Shield per d1 mini
- OLED piccolo (17mm diametro)
- Libreria Adafruit GFX
- Mostra:
  - Temperatura target
  - Temperatura attuale
  - % Potenza Heater
  - Eventi / Stati di errore

Selezione componenti

# Display OLED



# SETUP ARDUINO

- | Installazione ESP8266
- | Installazione librerie
- | Esempi

# Installazione ESP8266 Board

- In File>Preferences>Additional Board Manager URLs aggiungi  
`https://arduino.esp8266.com/stable/package_esp8266com_index.json`
- Apri Tools>Board>Board manager
- Installa ESP8266

*<https://github.com/esp8266/Arduino>*

Arduino

# Librerie aggiuntive

- UniversalTelegramBot
- Adafruit SSD1306 Wemos Mini OLED
- Adafruit ADS1X15

# Esempi di riferimento

## UniversalTelegramBot/ESP8266

- EchoBot
- CustomKeyboard
- SendMessageFromEvent

## Adafruit SSD1306 Wemos Mini OLED

- ssd1306\_64x48\_I2C

## Adafruit ADS1X15:

- singleended

# Bot di Telegram

Bots are third-party applications that run inside Telegram.

Users can interact with bots by sending them messages, commands and inline requests.

You control your bots using HTTPS requests to our Bot API.

*<https://core.telegram.org/bots>*

Bot di Telegram

# Come creo un Bot?

Con BotFather!

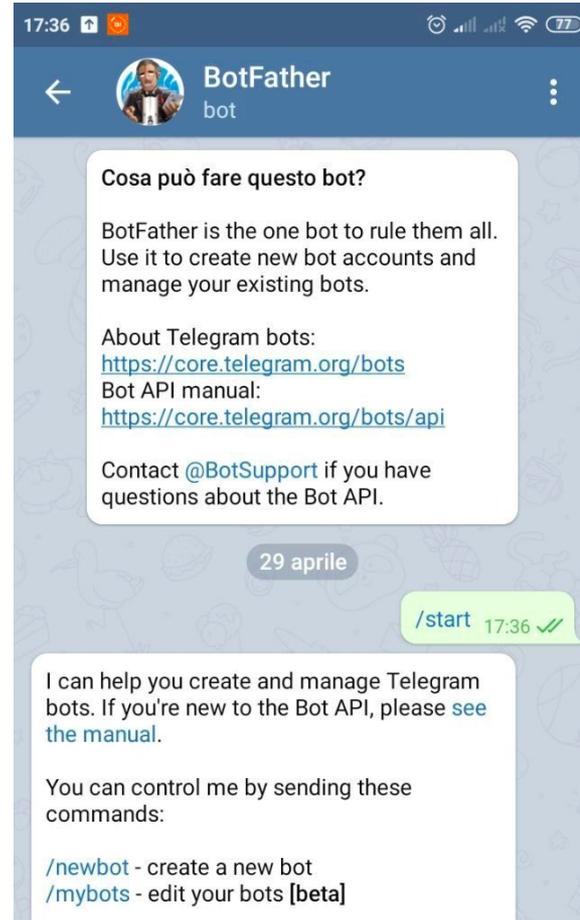
Telegram mette a disposizione un Bot (**BotFather**) per creare nuovi Bot e configurarli



Bot di Telegram

# Come creo un Bot?

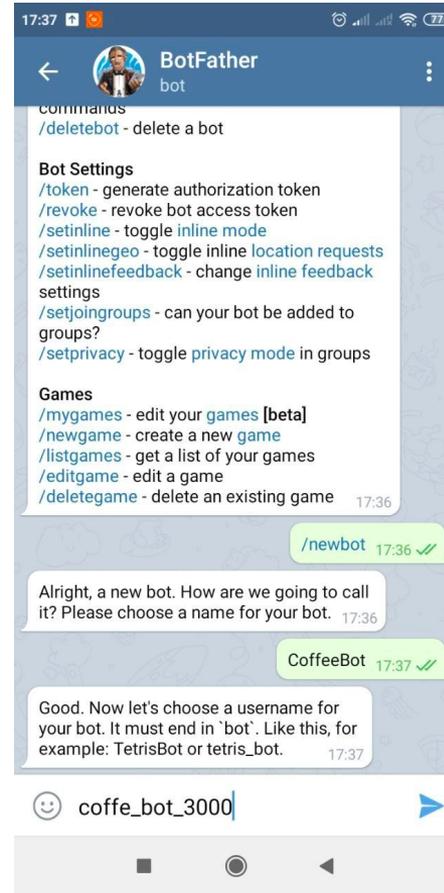
/start



Bot di Telegram

# Come creo un Bot?

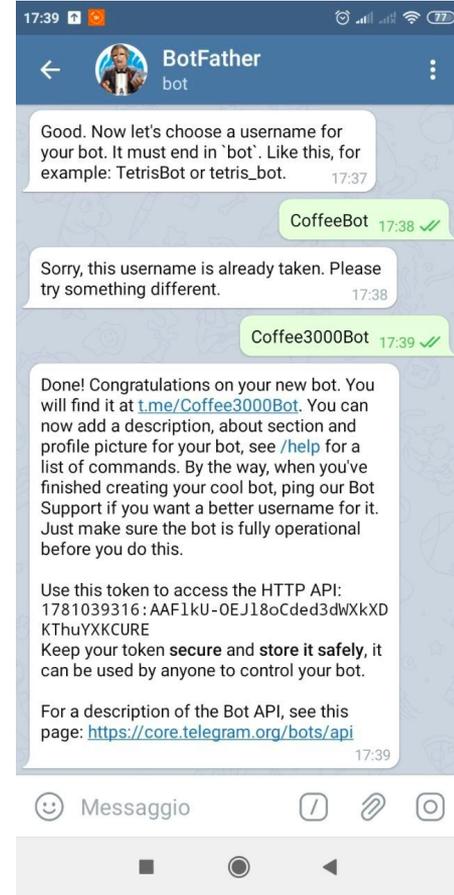
/newbot



Bot di Telegram

# Come creo un Bot?

Generazione Token



## Impostare il token

```
#include <UniversalTelegramBot.h>

// your Bot Token (Get from Botfather)
#define BOTtoken "1781039316:AAF1kU-OEJl8oCded3dWXkXDKTh"

WiFiClientSecure client;
UniversalTelegramBot bot(BOT_TOKEN, client);
```

Si definisce il Token che ci ha fornito BotFather per il nostro Bot.

Bot di Telegram

# Come creo un Bot?

Altri comandi:

/setname

/setabouttext

/setuserpic

/setcommands

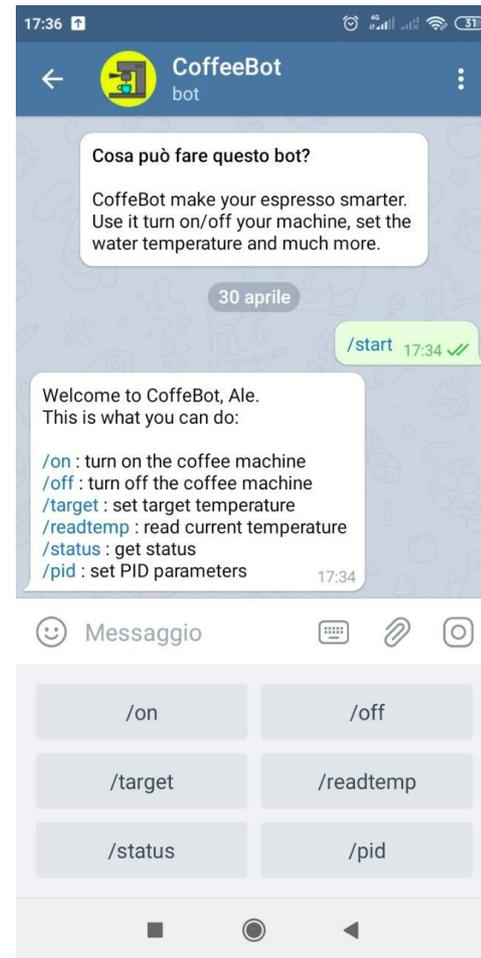
/deletebot



Bot di Telegram

# Esempio di utilizzo CoffeeBot

/start



Bot di Telegram

# Esempio di utilizzo CoffeeBot

/target



Bot di Telegram

# Esempio di utilizzo CoffeeBot

/pid



Bot di Telegram

# Esempio di utilizzo CoffeeBot

/pid



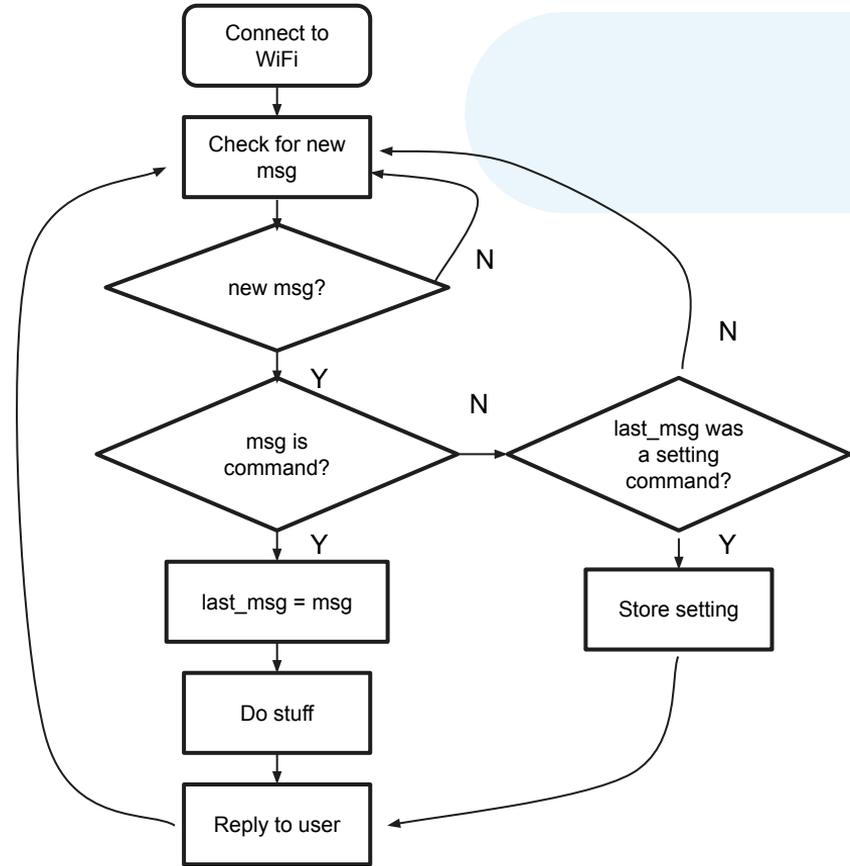
Bot di Telegram

# Esempio di utilizzo CoffeeBot

/status



# Interprete messaggio



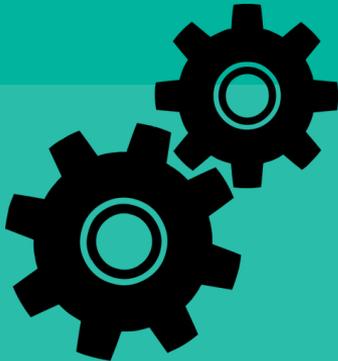
# CONCLUSIONI

- | Altre applicazioni
- | Next steps
- | Extra

# Altre applicazioni

- | Controllo remoto di altri prodotti
- | Domotica

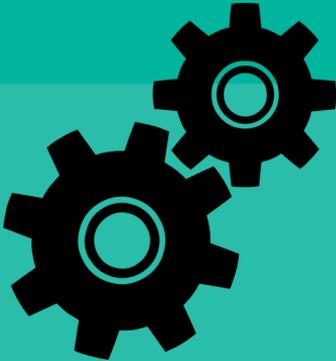
# Next Steps



FW:

- Aggiornamento OTA
- Accensione programmata

# Next Steps



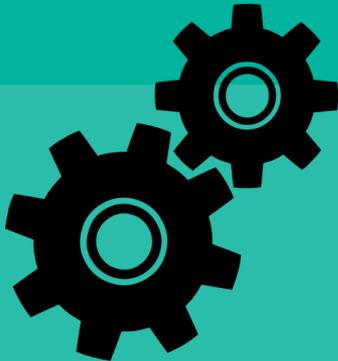
HW:

- Progettazione PCB compatto
- Integrazione all'interno della macchina
- Ottimizzazione RF

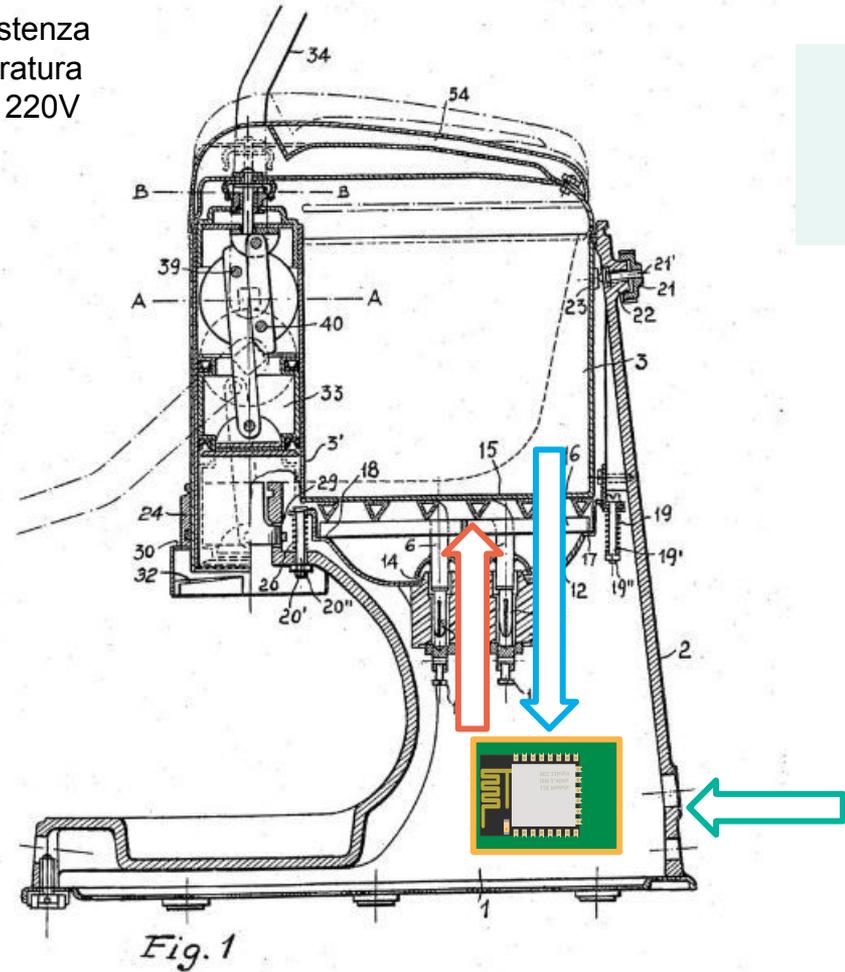
Conclusioni

# Next Steps

Integrazione



- Controllo Resistenza
- Lettura Temperatura
- Alimentazione 220V



Conclusioni

**Extra**



# Alessio Balbo

alessio@develer.com

Vuoi rimanere aggiornato sugli eventi Develer?  
Seguici nei nostri canali social:



develer



[www.develer.com](http://www.develer.com)