

1222·2022  
**800**  
ANNI



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

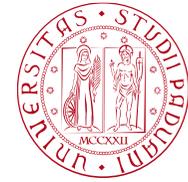


# Sviluppo di un BMS con bilanciamento passivo per batterie al piombo implementato su microcontrollore

University of Padova - Department of Management and Engineering (DTG)  
Corso di Laurea Triennale in Ingegneria Meccatronica (IMC)

Progetto di Tesi di Laurea Triennale

Anno accademico 2022/2023

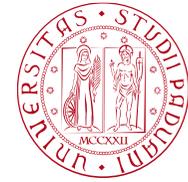


---

Relatore : Dott. Biadene Davide

Correlatore : Prof. Magnone Paolo

- Laureandi:
- Castagna Elia - 2008189
  - Mengo Edoardo - 2002720
  - Rizzi Tommaso - 2002724
  - Trento Luca - 2001556
  - Trevisan Manuele Moreno Gino - 2001885



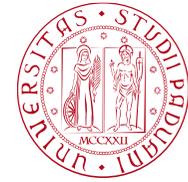
Sviluppo di un sistema di gestione della carica (BMS) di una batteria ad acido di piombo mediante bilanciamento passivo della carica, implementato su un microcontrollore.

I sistemi di battery management (BMS) monitorano la tensione delle diverse celle della batteria e attraverso opportuni algoritmi si occupano di eseguire il bilanciamento delle varie celle collegate in serie che, durante la carica e la scarica, possono presentare comportamenti diversi.

Lo scopo del progetto è quello di implementare un sistema BMS con bilanciamento passivo e di validarne il funzionamento. L'algoritmo per la gestione della batteria è stato sviluppato su microcontrollore tramite un ambiente di sviluppo e programmazione software. Al fine di validare il funzionamento del BMS, il gruppo di lavoro ha sviluppato un modello che rappresenti gli aspetti di interesse del funzionamento di una batteria all'acido di Piombo.

# Outline

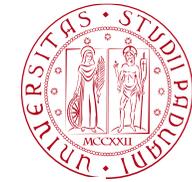
---



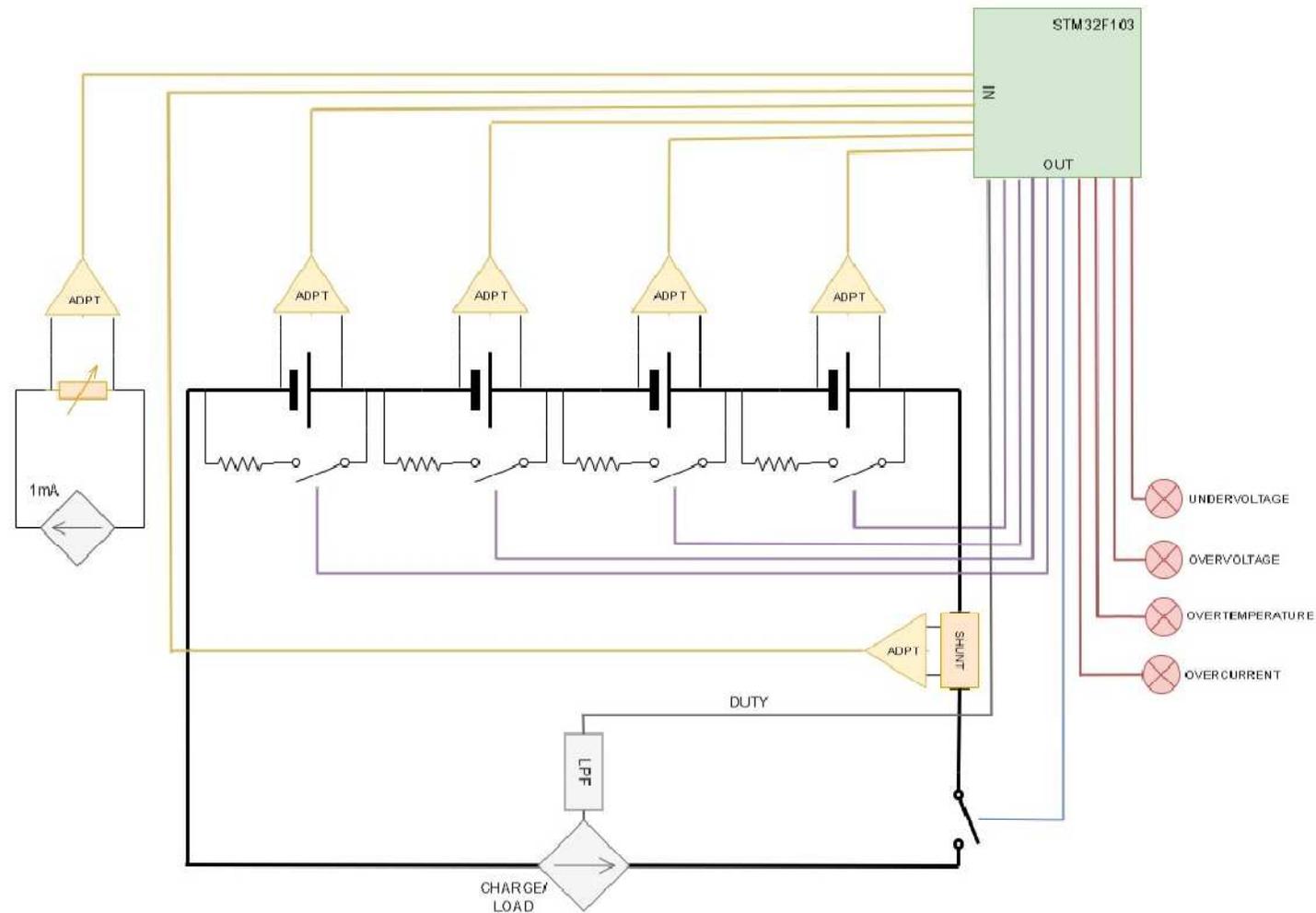
UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

1. Scelte progettuali
2. Modello della batteria
3. Modello termico
4. Configurazione periferiche
5. Algoritmo di bilanciamento passivo
6. Gestione degli allarmi
7. Strumentazione INA
8. Algoritmo di carica
9. Conclusioni
10. Bibliografia

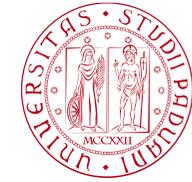
# 1. Scelte progettuali



## Schema generale



# 1. Scelte progettuali



## Batteria di riferimento



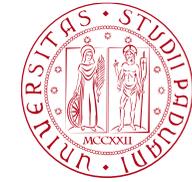
Nominal Voltage	2 V / cell
Nominal Capacity	21 W to 1.67 V per cell @ 25 °C
Internal resistance	23 mΩ approx
Maximum Discharge Current	60 A
Maximum Charge Current	2.1 A

Nominal Operating Temperature	25 °C
Discharge Temperature Range	[-15 50] °C
Charge Temperature Range	[-15 40] °C
Storage Temperature Range	[-15 40] °C

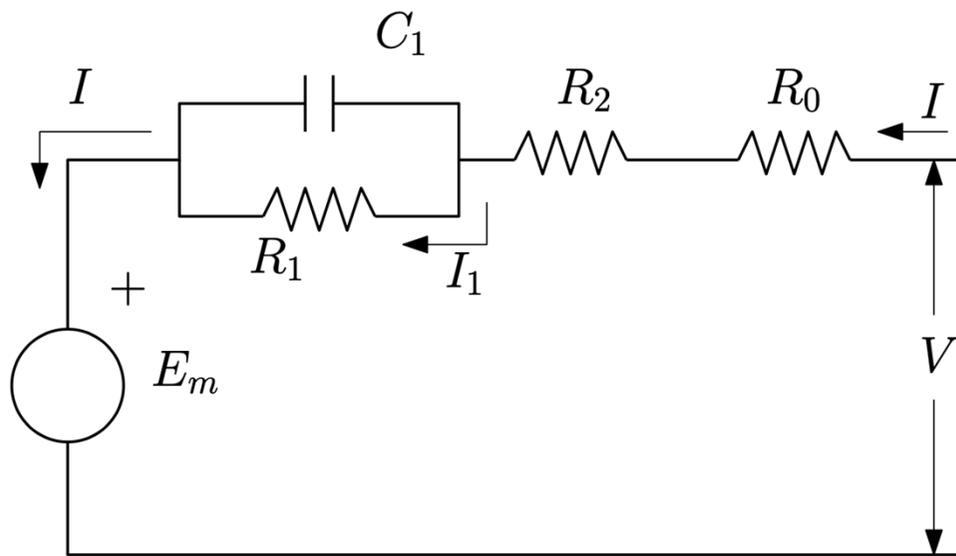
Constant Current Discharge Characteristics Unit: A (25°C, 77°F)												
F.V/Time	2MIN	4MIN	5MIN	6MIN	8MIN	10MIN	15MIN	20MIN	30MIN	45MIN	60MIN	90MIN
1.60V	46.4	31.2	27.3	24.4	20.3	17.4	12.9	10.4	7.29	5.15	4.02	2.83
1.67V	43.1	29.7	26.2	23.6	19.9	17.1	12.5	10.1	7.18	5.10	4.00	2.71
1.70V	41.3	28.9	25.6	23.2	19.7	16.8	12.3	9.93	7.14	5.08	3.99	2.65
1.75V	37.4	27.4	24.4	22.2	19.0	16.3	12.1	9.70	7.07	5.05	3.98	2.56
1.80V	33.1	25.2	22.8	21.0	18.2	15.7	11.8	9.51	7.00	5.02	3.96	2.46
1.85V	28.6	22.8	20.7	19.2	16.9	14.8	11.5	9.29	6.92	4.98	3.95	2.36



## 2. Modello della batteria



Grandezze di carica e tensione



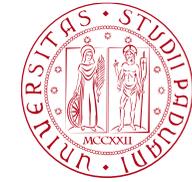
$$Q_e(t) = Q_{e\_init.} + \int_0^t I_m(\tau) d\tau$$

$$C(I, \vartheta) = \frac{K_c C_{0*} \left(1 + \frac{\vartheta}{-\vartheta_f}\right)^\varepsilon}{1 + (k_c + 1) \left(\frac{I}{I^*}\right)^\delta}$$

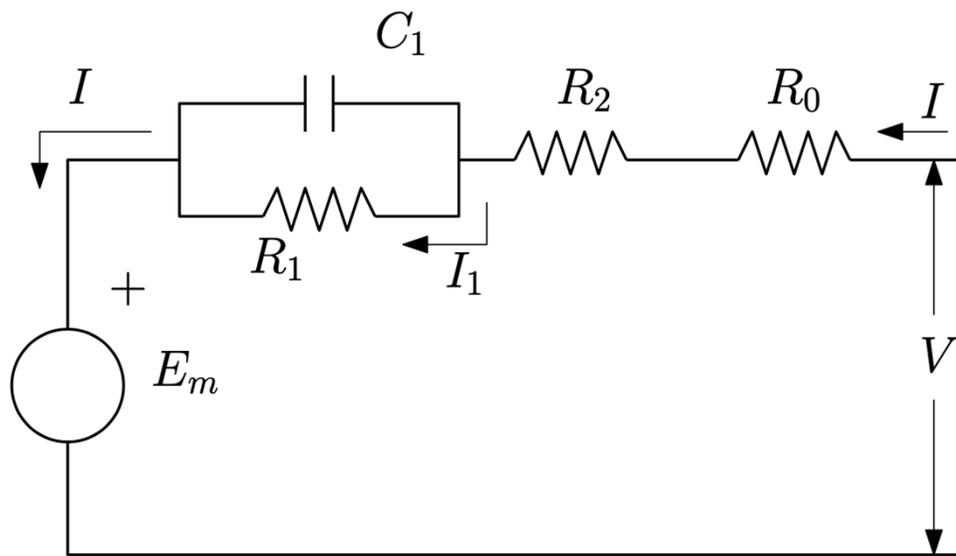
$$SOC = 1 - \frac{Q_e}{C(0, \vartheta)}$$

$$E_m = E_{m0} - K_e T (1 - SOC)$$

## 2. Modello della batteria



Grandezze di carica e tensione



$$R_1 = -R_{10} \ln(DOC)$$

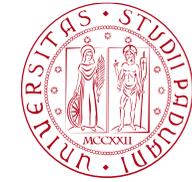
$$C_1 = \frac{\tau_1}{R_1}$$

$$R_0 = R_{00} (1 + A_0 (1 - SOC))$$

$$DOC = 1 - \frac{Q_e}{C(I_{avg}, \vartheta)}$$

$$R_2 = R_{20} \frac{e^{A_{21}(1-SOC)}}{1 + e\left(A_{22} \frac{I_m}{I^*}\right)}$$





## 2. Modello della batteria

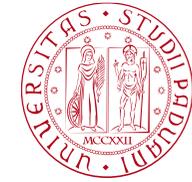
Discretizzazione delle espressioni

Carica estratta

$$Q_e(t) = Q_{e_{init}} + \int_0^t I(\tau) d\tau \quad \xrightarrow[\text{Laplace}]{\text{Trasformata di}} \quad Q_e(s) = Q_{e_{init}} + \frac{I}{s}$$

$$s = \frac{1 - z^{-1}}{T_c} \quad \xrightarrow[\text{di Eulero}]{\text{Approssimazione}} \quad Q_e(z) = Q_{e_{init}} + \frac{I}{\frac{1 - z^{-1}}{T_c}}$$

$$Q_e(z)(1 - z^{-1}) = Q_{e_{init}}(1 - z^{-1}) + I * T_c \quad \xrightarrow[\text{Zeta}]{\text{Antitrasformata}} \quad Q_e(k) = Q_e(k - 1) + I * T_c$$



## 2. Modello della batteria

Discretizzazione delle espressioni

Tensione ai capi del parallelo  $R_1 C_1$

$$\tau = R_1 C_1$$

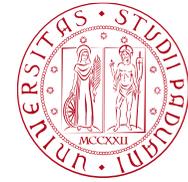
$$V_{din} = Z_1 I \longrightarrow Z_1 = R_1 // \frac{1}{sC_1} \longrightarrow Z_1 = \frac{R_1/sC_1}{R_1 + 1/sC_1} \longrightarrow Z_1 = \frac{R_1}{1 + s\tau}$$

$$V_{din} = \frac{R_1}{1 + s\tau} I \longrightarrow s = \frac{1 - z^{-1}}{T_c} \xrightarrow[\text{di Eulero}]{\text{Approssimazione}} V_{din} = \frac{R_1}{1 + \frac{1 - z^{-1}}{T_c} \tau} I$$

$$V_{din} = \frac{R_1 T_c}{T_c + \tau - \tau * z^{-1}} I \longrightarrow V_{din}(T_c + \tau - \tau * z^{-1}) = R_1 T_c I \xrightarrow[\text{Zeta}]{\text{Antitrasformata}}$$

$$(T_c + \tau)V_{din}(k) - \tau V_{din}(k - 1) = R_1 T_c I \longrightarrow V_{din}(k) = aV_{din}(k - 1) + bR_1 I$$

$$a = \frac{\tau}{T_c + \tau} \quad b = \frac{T_c}{T_c + \tau}$$



## 2. Modello della batteria

---

Valori dei parametri

```
Em_1 = 0.0;    Em0_1 = 2.135;    Kost_1 = 0.00058;    //parametri generatore di tensione ideale

Qe_1 = Q_set*C_set;    C_1 = 4*3600;    SOC_1 = 1;    //parametri di carica

Res1_1 = 0.00018295;    Res2_1 = 0.007811;    Res0_1 = 0;
Resb_1 = 0.002;    A0_1 = -0.3;    //parametri resistivi

Vdin_1 = 0;    tau_1 = 5000;
a_din_1 = tau_1 / (tau_1 + Tc);    b_din_1 = 1 - a_din_1;    //parametri dinamici del parallelo RC

Ibalance_1 = 0.0;    //corrente di bilanciamento

V_1 = 0.0;    //tensione ai capi della cella
```

Dal file di simulazione *simu\_celle*



## 2. Modello della batteria

Espressioni delle grandezze variabili

```
Ibalance_1 = (V_1/Rbalance) * ((PORTC & 0x00000002)>>1); //corrente di bilanciamento

Qe_1 = Qe_1 + I*Tc + Ibalance_1*Tc; //carica estratta

SOC_1 = 1 - Qe_1 / C_1; //State Of Charge (stato di carica)

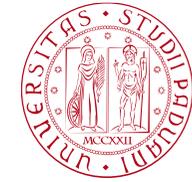
Em_1 = Em0_1 - Kost_1 * (273+theta) * (1 - SOC_1); //tensione del generatore della cella

Res0_1 = Resb_1 * (1 + A0_1 * (1 - SOC_1)); //resistenza R0

Vdin_1 = a_din_1*Vdin_1 + b_din_1*Res1_1*(I+Ibalance_1); //tensione del parallelo tra RC

V_1 = Em_1 - Vdin_1 - (Res2_1 + Res0_1)*(I+Ibalance_1); //tensione della cella
```

Dal file di simulazione *simu\_celle*



### 3. Modello termico

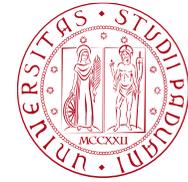
Discretizzazione dell'espressione

$$\theta(t) = \theta_{init} + \int_0^t \frac{P_s - \frac{(\theta - \theta_a)}{R_\theta}}{C_\theta} d\tau \quad \xrightarrow[\text{Laplace}]{\text{Trasformata di}} \quad \theta(s) = \theta_{init} + \frac{1}{s} \frac{P_s - \frac{(\theta - \theta_a)}{R_\theta}}{C_\theta}$$

$$s = \frac{1 - z^{-1}}{T_c} \quad \xrightarrow[\text{di Eulero}]{\text{Approssimazione}} \quad \theta(z) = \theta_{init} + \frac{T_s P_s(z)}{(1 - z^{-1}) C_\theta} - \frac{T_s}{1 - z^{-1}} \left( \frac{\theta(z) - \theta_a}{\tau_{th}} \right) \quad \tau_{th} = R_\theta C_\theta$$

$$\theta \left( 1 + \frac{T_s}{\tau_{th}} - z^{-1} \right) = \theta_{init} (1 - z^{-1}) + \frac{T_s P_s}{C_\theta} - \frac{T_s}{\tau_{th}} \theta_a \quad \xrightarrow[\text{Zeta}]{\text{Antitrasformata}} \quad \theta(k) \left( 1 + \frac{T_s}{\tau_{th}} \right) = \theta(k - 1) + \frac{T_s}{C_\theta} P_s(k) - \frac{T_s}{\tau_{th}} \theta_a$$

$$\theta(k) = a\theta(k - 1) + bR_{th}P_s(k) - b\theta_a \quad a = \frac{\tau_{th}}{T_s + \tau_{th}} \quad b = \frac{\tau_{th}}{T_s + \tau_{th}} \quad P_s = \sum_{i=1}^4 V_i I$$



### 3. Modello termico

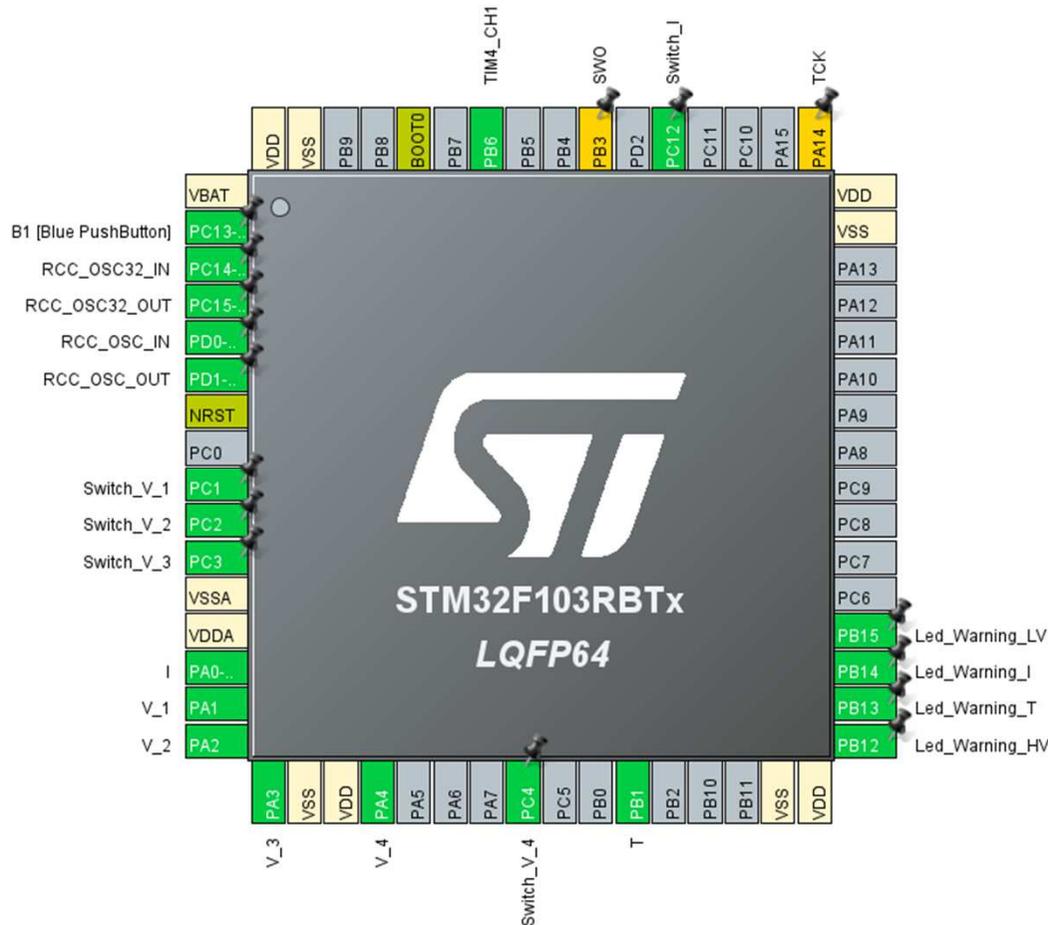
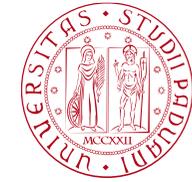
---

Valori dei parametri ed espressione delle grandezze

```
Rth = 0.2 ;      Pel = 0;      tau_th = 10800;           //parametri termici  
  
a_th = tau_th / (tau_th+Tc);  b_th = 1 - a_th;           //parametri dinamici  
  
theta_amb = 25;      theta = theta_amb,           //temperatura  
  
I_pot = I_set - I_gen;  
  
Pel = V_1*(I_pot+Ibalance_1) + V_2*(I_pot+Ibalance_2)           //potenza elettrica  
      + V_3*(I_pot+Ibalance_3) + V_4*(I_pot+Ibalance_4);  
  
theta = a_th*theta + b_th*Rth*Pel + b_th*theta_amb ;           //temperatura celle
```

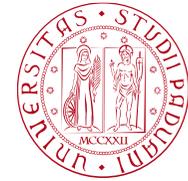
Dal file di simulazione *simu\_celle*

# 4. Configurazione delle periferiche



Le periferiche utilizzate per il progetto sono:

- ADC1 : Utilizzato per l'acquisizione delle tensioni e della corrente dai circuiti di condizionamento.
- ADC2 : Riservato all'acquisizione del sensore di temperatura dai circuiti di condizionamento.
- TIM2 : Per la regolazione della frequenza di campionamento dell'ADC2.
- TIM3 : Per la regolazione della frequenza di campionamento dell'ADC1.
- TIM4 : Utilizzato per la generazione di un segnale DPWM per l'invio del segnale analogico per il generatore di corrente di carica.
- Output GPIO



## 4. Configurazione delle periferiche

### Acquisizione della corrente di carico

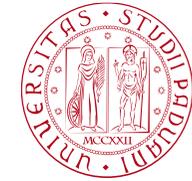
Il TIM3 è impostato per far acquisire il segnale di corrente dall'ADC1 a frequenza:

$$f_{TIM3} = \frac{f_{APB1}}{(counter\_period+1) \cdot (prescaler+1)} = 1kHz \quad \text{con} \quad \begin{cases} f_{APB1} = 8 MHz \\ counter\_period = 7999 \\ prescaler = 0 \end{cases}$$

L'avvio del campionamento è eseguito da codice nell'ISR del TIM3:

```
void TIM3_IRQHandler(void) { ...  
  
    hadc1.Instance->SQR3 = ADC_CHANNEL_0;           // Selezione canale 0  
    HAL_ADC_Start(&hadc1);                          // Avvio conversione  
    HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY); // Attesa conversione  
    ADC1_I = HAL_ADC_GetValue(&hadc1);             // Prelevamento valore di corrente  
  
    ... }
```

Dal file stm32f1xx\_IT

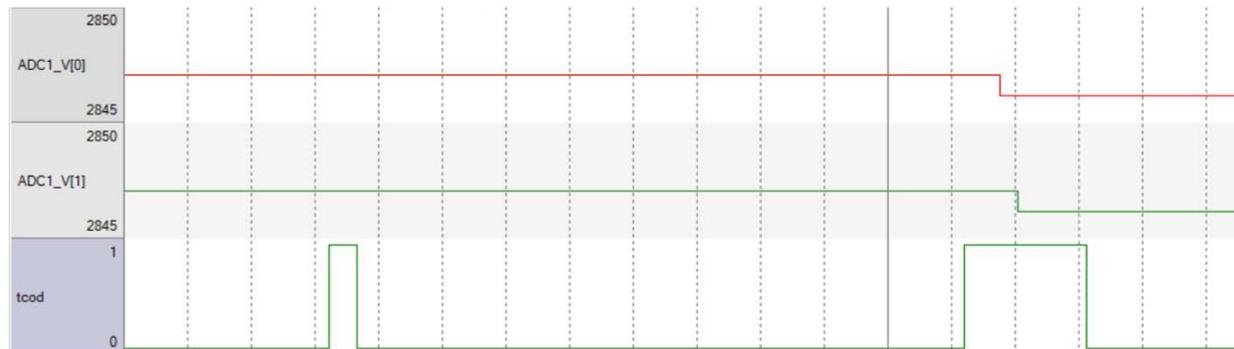


## 4. Configurazione delle periferiche

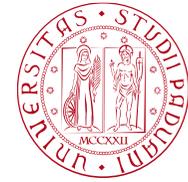
### Acquisizione della tensione delle celle

La frequenza di campionamento delle tensioni delle celle è stata impostata a  $f_c = 100 \text{ Hz}$ , frequenza consona rispetto alla costante di tempo elettrica :  $f_c \gg 1/2\pi\tau_{el} \approx 32 \mu\text{Hz}$  con  $\tau_{el} = 5000\text{s}$

```
void TIM3_IRQHandler(void) { ...  
    if(count100Hz == 10) {  
        hadc1.Instance->SQR3 = ADC_CHANNEL_1;           // Selezione canale 1  
        HAL_ADC_Start(&hadc1);                          // Avvio conversione  
        HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY); // Attesa conversione  
        ADC1_V[0] = HAL_ADC_GetValue(&hadc1);           // Prelevamento valore di tensione cella 1  
    }  
... }
```



Dal file stm32f1xx\_IT



## 4. Configurazione delle periferiche

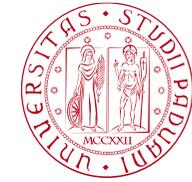
### Acquisizione della temperatura

L'acquisizione della temperatura viene eseguita ad una frequenza  $f_{TIM2} = 10Hz$  in modo da assicurare il vincolo

$$f_{TIM2} \gg \frac{1}{2\pi\tau_{th}} \approx 15\mu Hz \quad \text{dove } \tau_{th} = 10800 s$$

La conversione viene effettuata da codice all'interno della ISR del TIM2:

```
void TIM2_IRQHandler(void) {  
  
    HAL_ADC_Start_IT(&hadc2);           // Avvio conversione  
    HAL_ADC_PollForConversion(&hadc2, HAL_MAX_DELAY); // Attesa conversione  
    ADC2_T = HAL_ADC_GetValue(&hadc2); // Prelevamento valore di temperatura  
    ...  
}
```



## 4. Configurazione delle periferiche

### Uscite digitali GPIO

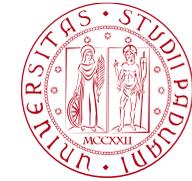
- Comando degli switch di scarica passiva delle celle
- Segnalazioni di stati critici del sistema BMS (sovratensione, sovracorrente, sovratemperatura)
- Comando dello switch master di isolamento carico

Pin Name	Signal on Pin	GPIO output ...	GPIO mode	GPIO Pull-u...	Maxi...	User Label	Modified
PB12	n/a	Low	Output Push Pull	No pull-up a...	Low	Led_Warning_V	✓
PB13	n/a	Low	Output Push Pull	No pull-up a...	Low	Led_Warning_T	✓
PB14	n/a	Low	Output Push Pull	No pull-up a...	Low	Led_Warning_I	✓
PB15	n/a	Low	Output Push Pull	No pull-up a...	Low	Safety_Button_I	✓
PC1	n/a	Low	Output Push Pull	No pull-up a...	Low	Switch_V_1	✓
PC2	n/a	Low	Output Push Pull	No pull-up a...	Low	Switch_V_2	✓
PC3	n/a	Low	Output Push Pull	No pull-up a...	Low	Switch_V_3	✓
PC4	n/a	Low	Output Push Pull	No pull-up a...	Low	Switch_V_4	✓
PC12	n/a	Low	Output Push Pull	No pull-up a...	Low	Switch_I	✓
PC13-TAMP...	n/a	n/a	External Interrupt ...	No pull-up a...	n/a	B1 [Blue Push...	✓

PB12 Configuration :-

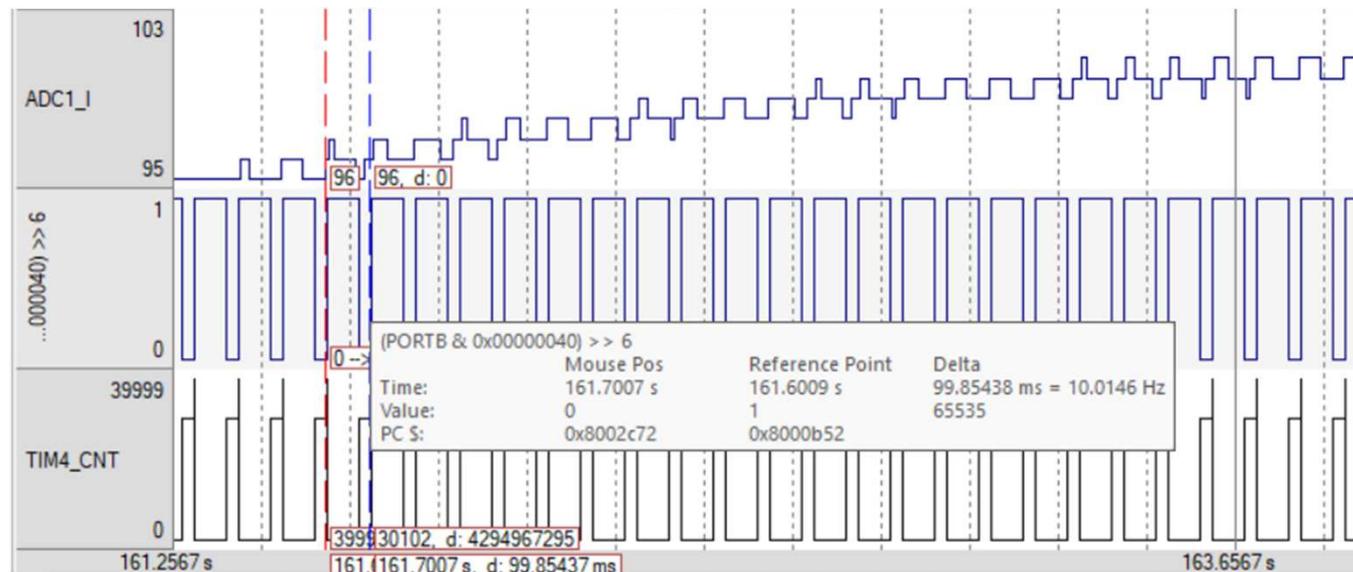
GPIO output level	Low
GPIO mode	Output Push Pull
GPIO Pull-up/Pull-down	No pull-up and no pull-down
Maximum output speed	Low
User Label	Led_Warning_V

# 4. Configurazione delle periferiche



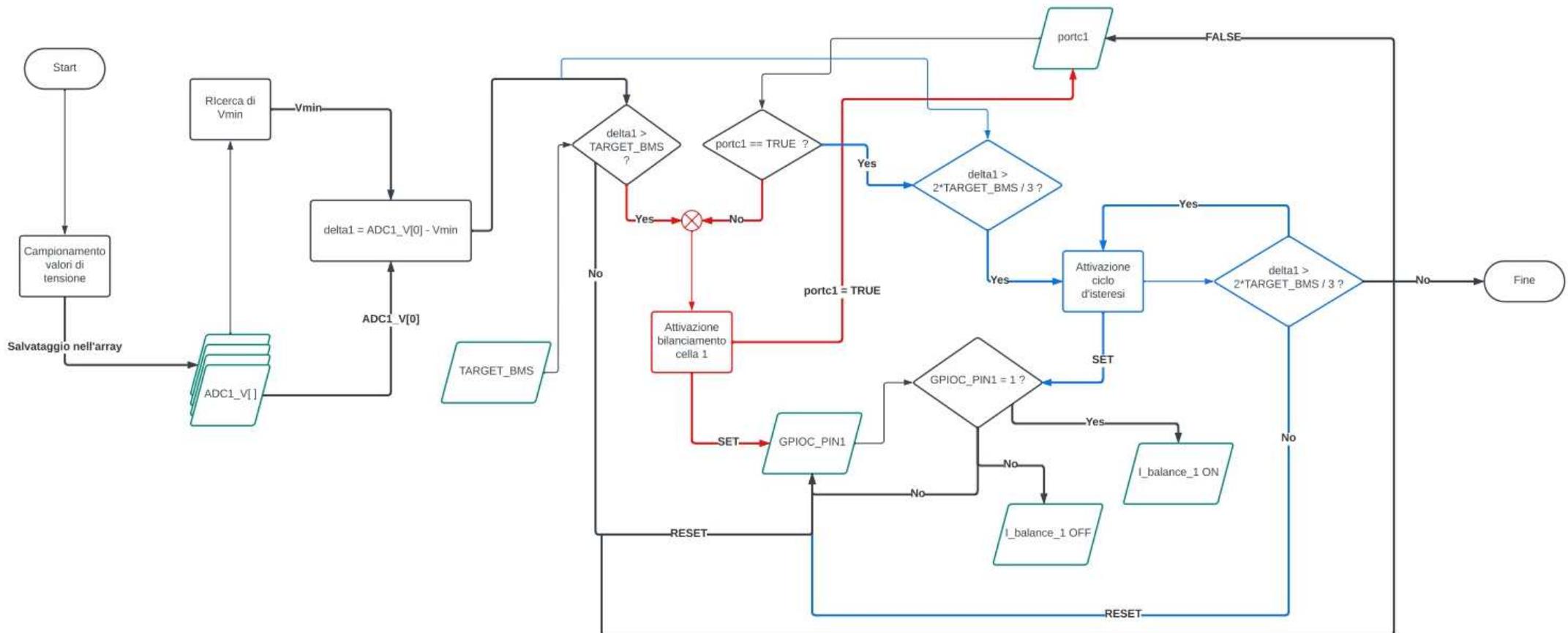
## Generazione del segnale PWM

Attraverso il TIM4 viene generato un segnale DPWM di tipo edge-aligned a frequenza  $f_{TIM4} = 10Hz$  usato per il comando di corrente nell'algoritmo di ricarica. L'informazione della funzione di carica viene gestita dal duty-cycle che viene poi aggiornato da programma all'interno dell'ISR del TIM4, modificando il valore del registro di MATCH.

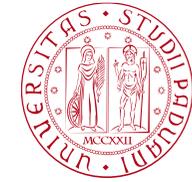


# 5. Algoritmo di bilanciamento passivo

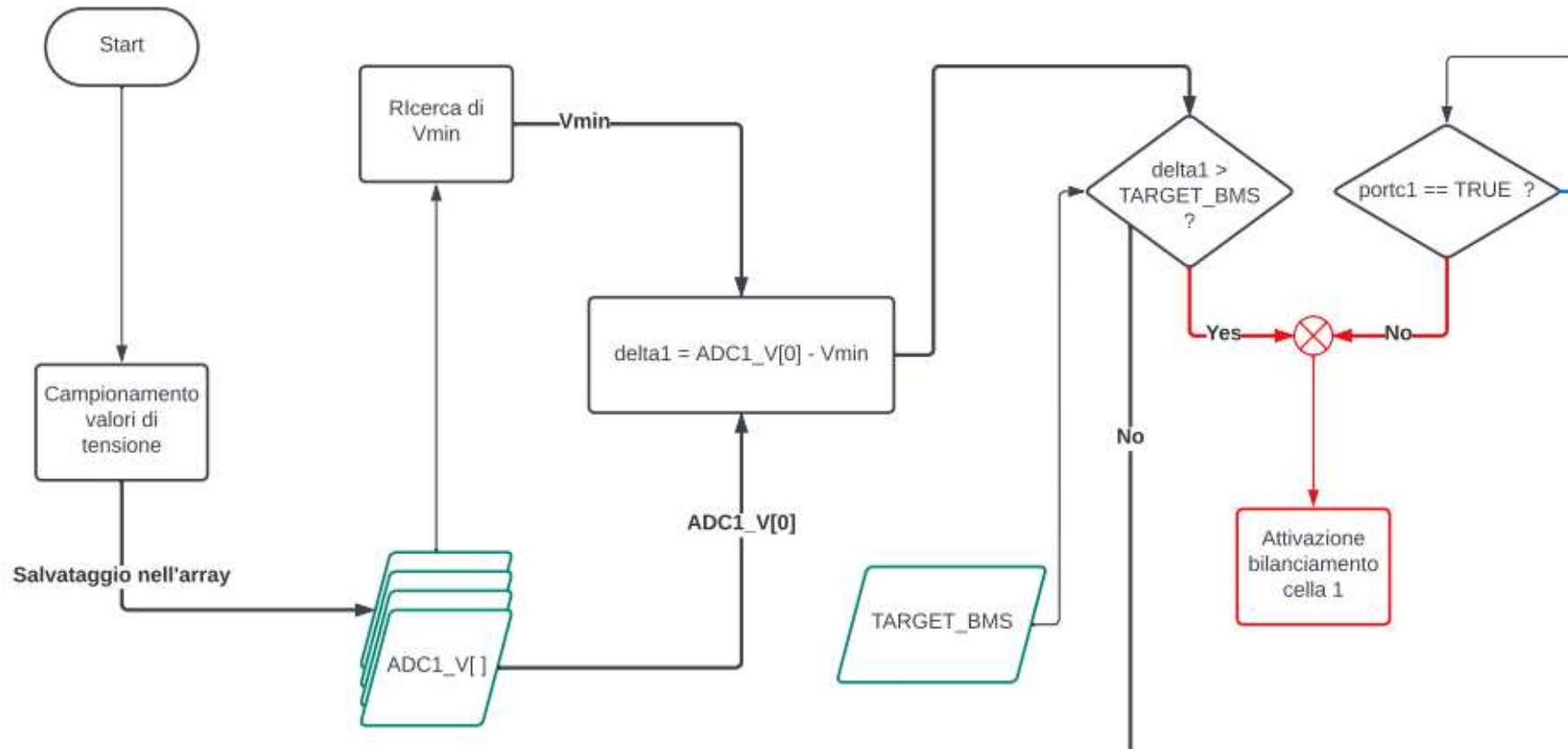
## Schema generale



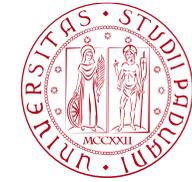
# 5. Algoritmo di bilanciamento passivo



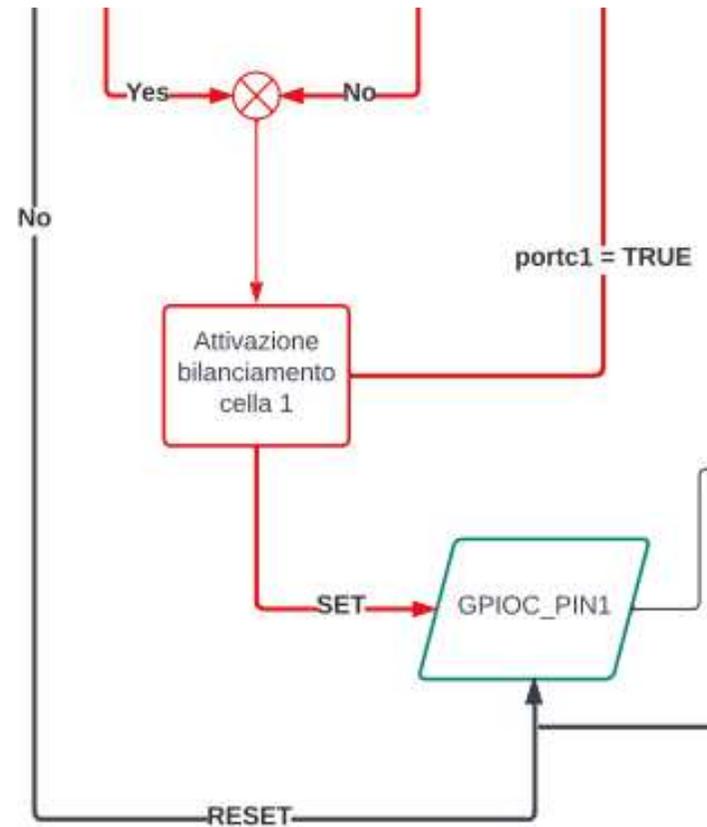
Ricerca celle a tensione massima e minima



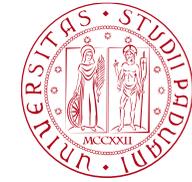
# 5. Algoritmo di bilanciamento passivo



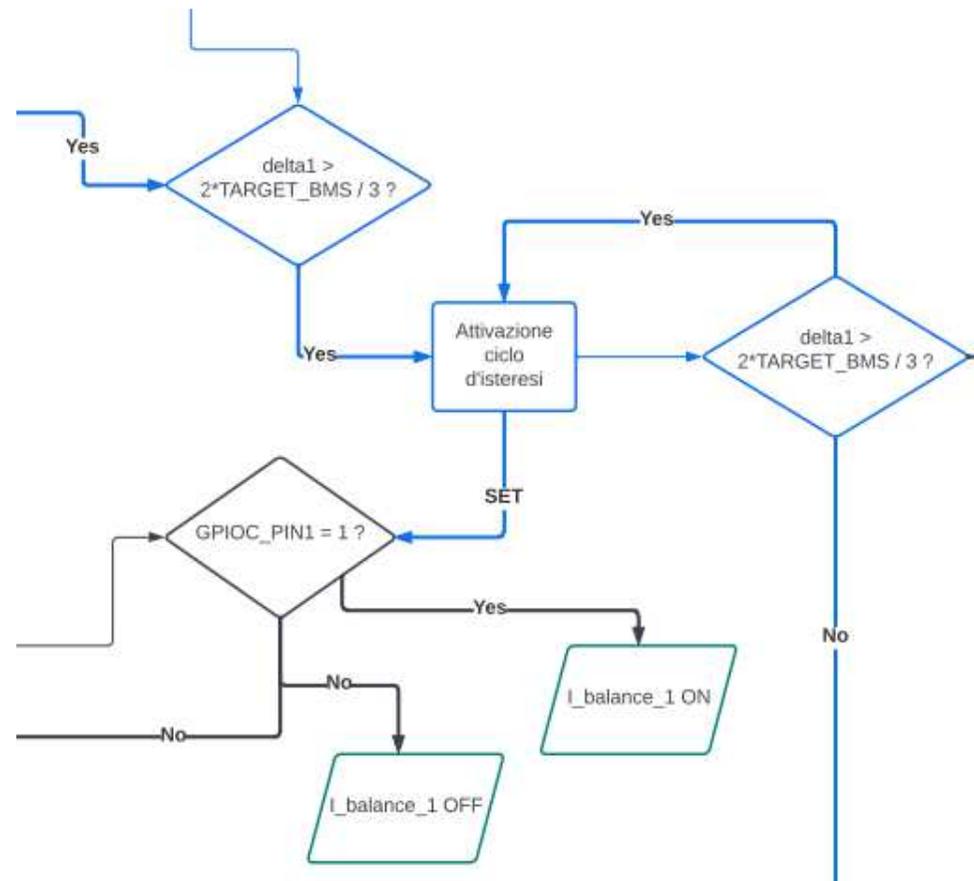
Attivazione circuito di bilanciamento

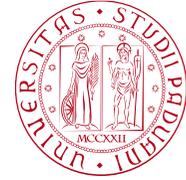


# 5. Algoritmo di bilanciamento passivo



Attivazione ciclo d'isteresi





## 5. Algoritmo di bilanciamento passivo

Algoritmo di ricerca massimo e minimo

```
Vmin = ADC1_V[0];
```

```
Vmax = ADC1_V[0];
```

```
for(int i = 0; i < 4; i++)
```

```
{
```

```
    if(Vmax <= ADC1_V[i]) {
```

```
        Vmax = ADC1_V[i];
```

```
        indexmax = i+1;
```

```
    }
```

```
    if(Vmin >= ADC1_V[i]) {
```

```
        Vmin = ADC1_V[i];
```

```
        indexmin = i+1;
```

```
    }
```

```
}
```

```
//delta maggiore fra le tensioni delle celle
```

```
deltaV = Vmax - Vmin;
```

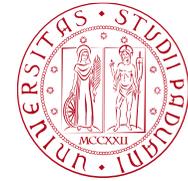
```
//delta delle singole celle rispetto a Vmin
```

```
delta1 = ADC1_V[0] - Vmin;
```

```
delta2 = ADC1_V[1] - Vmin;
```

```
delta3 = ADC1_V[2] - Vmin;
```

```
delta4 = ADC1_V[3] - Vmin;
```



## 5. Algoritmo di bilanciamento passivo

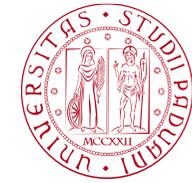
Codice per attivazione bilanciamento e ciclo d'isteresi

```
//Condizione di attivazione bilanciamento cella 1
if (!portc1 && (delta1 > TARGET_BMS) ) {
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_1, GPIO_PIN_SET);
    portc1 = true;    //flag pin PC1: se portc1 = true allora il BMS è in esecuzione
}

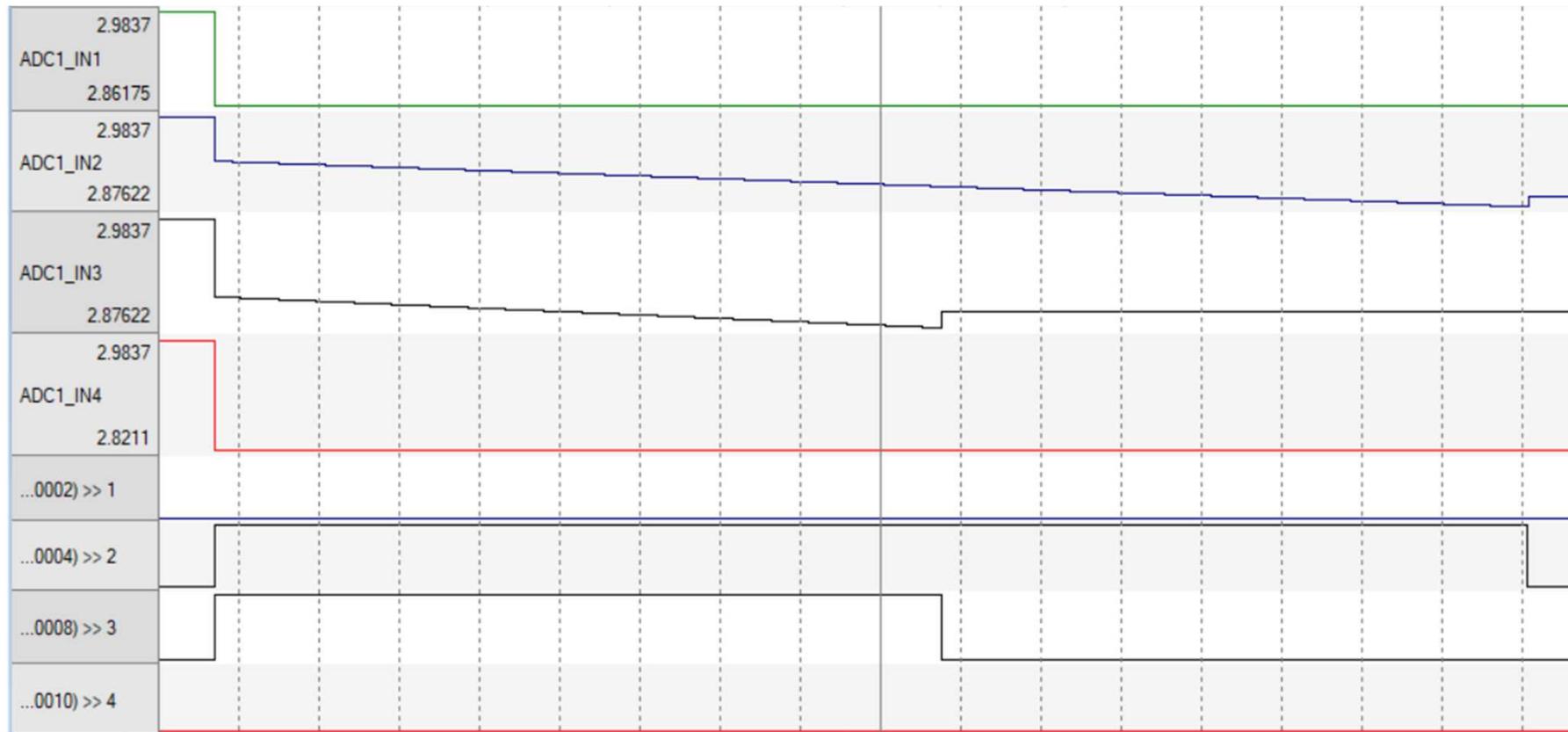
//isteresi attivazione BMS
else if (portc1 && (delta1 > 2*TARGET_BMS/3)) {
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_1, GPIO_PIN_SET);    //set pin PC1
}

else {
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_1, GPIO_PIN_RESET);    //reset pin PC1
    portc1 = false;
}
```

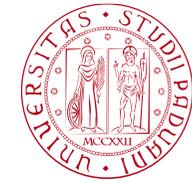
# 5. Algoritmo di bilanciamento passivo



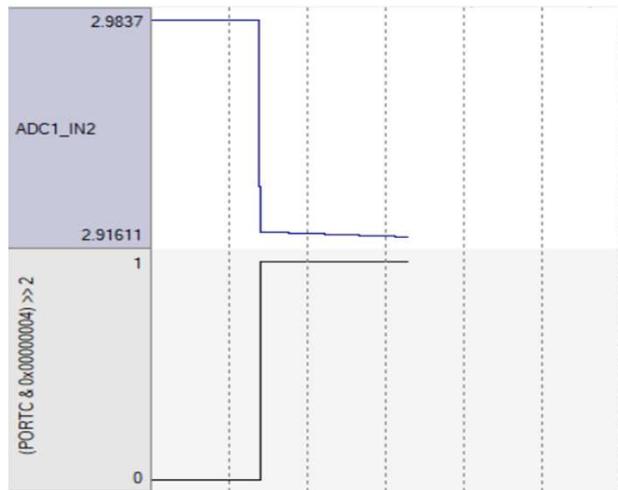
## Simulazione funzionamento BMS



# 5. Algoritmo di bilanciamento passivo

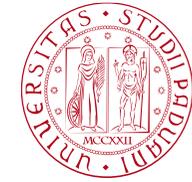


## Attivazione e disattivazione del pin PC2



Watch 1		
Name	Value	Type
deltaV	174	uint
delta1	64	uint
delta2	174	uint
delta3	142	uint
delta4	0	uint

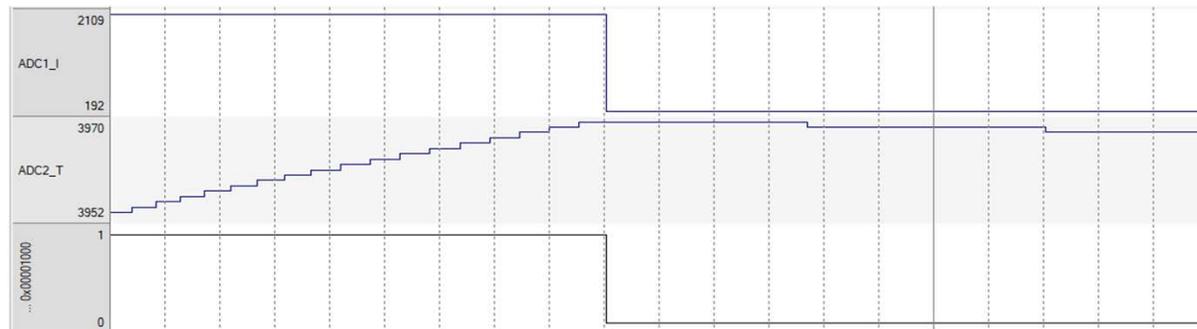
Watch 1		
Name	Value	Type
deltaV	86	uint
delta1	64	uint
delta2	86	uint
delta3	86	uint
delta4	0	uint



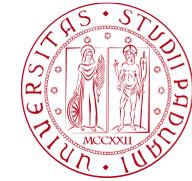
## 6. Gestione allarmi

### Allarme overtemperature

```
#define OVERTEMPERATURE_DISCHARGE 4012  
....  
if(ADC2_T >= OVERTEMPERATURE_DISCHARGE){  
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_12, GPIO_PIN_RESET); //Valore di temperatura corrispondente a 50 (°C)  
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_13, GPIO_PIN_SET); //Aprò l'interruttore che porta la corrente a 0  
    reset= true; //Accensione led allarme temperatura  
} //Attivazione pulsante di reset  
else if(!reset){  
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_13, GPIO_PIN_RESET); //Spegnimento led segnalazione allarme  
}  
if (!reset)  
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_12, GPIO_PIN_SET); //Attivazione pin se non si presenta alcun allarme
```



Dal file stm32f1xx\_IT



## 6. Gestione allarmi

### Allarme undervoltage

```
#define UNDERVOLTAGE 2896
```

```
....
```

```
if(Vmin < UNDERVOLTAGE && Vmin > 0 && ADC1_I == 192) {  
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_12, GPIO_PIN_RESET);  
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_15, GPIO_PIN_SET);  
    reset = true;  
}
```

```
//Valore di tensione corrispondente alla tensione  
//effettiva interna delle celle di 1.540 (V)
```

```
//Apro l'interruttore che porta la corrente a 0  
//Accensione led segnalazione allarme temperatura  
//Attivazione pulsante di reset
```

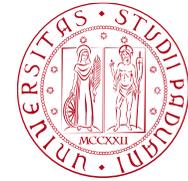
```
else if(!reset) {
```

```
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_15, GPIO_PIN_RESET);  
}
```

```
//Spegnimento led segnalazione allarme
```



Dal file stm32f1xx\_IT



## 6. Gestione allarmi

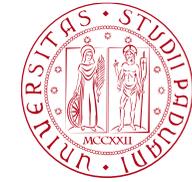
### Allarme overvoltage in scarica

```
#define OVERVOLTAGE_D 3715 //Valore di tensione corrispondente = 2.135 (V)
....
if((Vmax > OVERVOLTAGE_D && state == 0) || (overvolt_d == true && Vmax > OVERVOLTAGE_D - 20)){
    overvolt_d = true;
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_12, GPIO_PIN_RESET); //Apertura interruttore che porta la corrente a 0
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_SET); //Accensione led segnalazione allarme di tensione
    ....
}
else if (!reset) HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_RESET); //Spegnimento led segnalazione allarme di tensione
else overvolt_d = false;
```

### Allarme overvoltage in carica

```
#define OVERVOLTAGE_C 3803 //valore di tensione corrispondente = 2.15 (V)
....
if((Vmax > OVERVOLTAGE_C && state != 0) || (Vmax > OVERVOLTAGE_C - 20)){
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_12, GPIO_PIN_RESET); //Apertura interruttore che porta la corrente a 0
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_SET); //Accensione led segnalazione allarme di tensione
    ....
}
else if (!reset) HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_RESET); //Spegnimento led segnalazione allarme di tensione
```

Dal file stm32f1xx\_IT



## 6. Gestione allarmi

### Allarme overvoltage

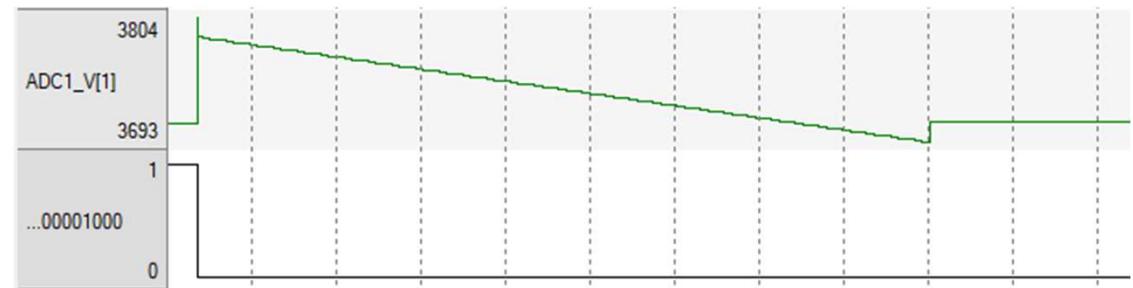
```
if ( !portc1 && ADC1_V[0] >= OVERVOLTAGE_D ) { //Attivazione scarica della cella
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_1, GPIO_PIN_SET);
    portc1 = true;
}

else if ( portc1 && ( ( OVERVOLTAGE_D - ADC1_V[0] < 22 ) || ( ADC1_V[0] >= OVERVOLTAGE_D ) ) ) //Ciclo d'isteresi scarica cella
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_1, GPIO_PIN_SET);

else { //Condizione di uscita dalla scarica della cella
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_1, GPIO_PIN_RESET);
    portc1 = false;
}

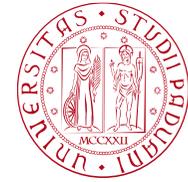
....

//Uscita dall'allarme di overvoltage
if (!portc1 && !portc2 && !portc3 && !portc4) {
    overvolt_d = false;
    reset= true;
}
```



Dal file stm32f1xx\_IT

# 6. Gestione allarmi



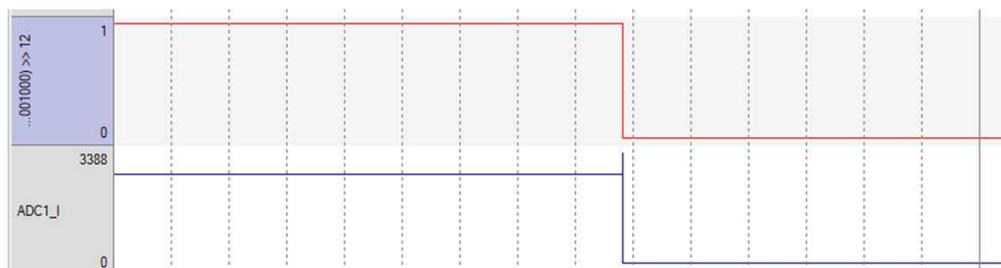
## Allarme overcurrent

```
#define OVERCURRENT 3392
....
if(ADC1_I >= OVERCURRENT) {
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_12, GPIO_PIN_RESET);
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_14, GPIO_PIN_SET);
    reset = true;
}
else if (!reset) {
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_14, GPIO_PIN_RESET);
}
```

//Valore di corrente corrispondente a 49.2 (A)

//Apertura interruttore che porta la corrente a 0  
//Attivazione led allarme  
//Attivazione pulsante di reset

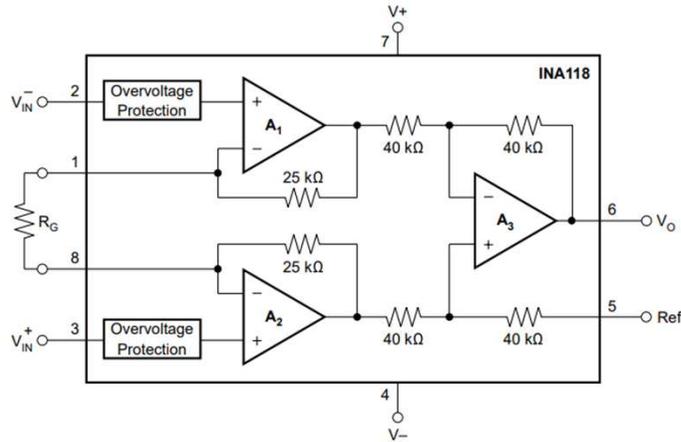
//Spegnimento led allarme



```
if (!reset)
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_12, GPIO_PIN_SET);
//Attivazione pin master se non si presenta alcun allarme
```

# 7. Strumentazione INA

## Circuito di amplificazione

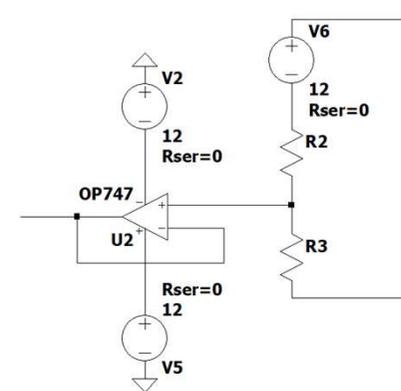


PIN		TYPE	DESCRIPTION
NO.	NAME		
1	R <sub>G</sub>	—	Gain setting pin. For gains greater than 1, place a gain resistor between pin 1 and pin 8.
2	V <sup>-</sup> <sub>IN</sub>	Input	Negative input
3	V <sup>+</sup> <sub>IN</sub>	Input	Positive input
4	V <sup>-</sup>	Power	Negative supply
5	Ref	Input	Reference input. This pin must be driven by low impedance or connected to ground.
6	V <sub>O</sub>	Output	Output
7	V <sup>+</sup>	Power	Positive supply
8	R <sub>G</sub>	—	Gain setting pin. For gains greater than 1, place a gain resistor between pin 1 and pin 8.

$$G = 1 + \frac{50k\Omega}{R_G}$$

$$R_G = \frac{50k\Omega}{G - 1}$$

## Circuito di offset

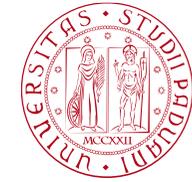


$$V_{offset} = V_{gen} \frac{R_3}{R_3 + R_2}$$

$$R_3 = 10k\Omega$$

$$R_2 = \frac{V_{gen}R_3 - V_{offset}VR_3}{V_{offset}}$$

# 7. Strumentazione INA



## Sensore di temperatura

Modello: PTF Pt100

Transcaratteristica:  $R_T = R_0(1 + aT + bT^2)$

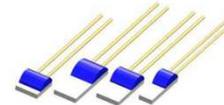
$$R_0 = 100 \quad a = 3,9083 \times 10^{-3} \quad b = -5,775 \times 10^{-7}$$

Range: [-16 51] °C

$$R_{min} = R_0(1 + a(-16) + b(-16)^2) = 93,73\Omega$$

$$R_{max} = R_0(1 + a(51) + b(51)^2) = 119,78\Omega$$

$$I = 1mA \quad V_{min} = 0,09373V \quad V_{max} = 1,1978V$$



Parameter	Symbol	Condition	Min	Typ	Max	Unit
Nominal Resistance at 0 °C	R <sub>0</sub>	Class B (F0.3) Pt100 Class B (F0.3) Pt1000	99.88 998.8	100.00 1000.0	100.12 1001.2	Ω
Tolerance at 25°C		Room temperature calibration	-0.43	0	0.43	°C
Temperature Coefficient of Resistance	TCR	0 °C, 100 °C		3850		ppm/°C
Tolerance Temperature Range*		Class C (F 0.6) Class B (F 0.3) Class A (F 0.15) Class T (F 0.1) Class B/Cryo (F 0.3)	-50 -50 -30 -30 -200		600 600 300 200 200	°C
Self-Heating Coefficient in Air Flow: 1 m/s		PTFC outline PTFD outline PTFF outline PTFM outline		0.5 0.33 0.5 0.5		°C/mW
Response Time Water Flow: 0.4 m/s	T <sub>W0.9</sub>	PTFC outline PTFD outline PTFF outline PTFM outline		0.2 0.35 0.2 0.2		s
Response Time Air Flow: 1 m/s	T <sub>A0.9</sub>	PTFC outline PTFD outline PTFF outline PTFM outline		10 17 10 10		s
Measuring Current R <sub>0</sub> : 100 Ω		PTFC outline (Class B) PTFD outline (Class B) PTFF outline (Class B) PTFM outline (Class B)			1.4 1.7 1.4 1.4	mA

## Resistenza di Shunt

Modello: Yageo PU593

Resistenza nominale:  $R = 1m\Omega$

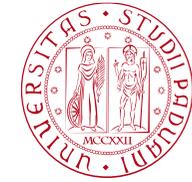
Range: [-3 60] A

$$V_{min} = -0,003V$$

$$V_{max} = 0,06V$$



Attributo del prodotto	Valore dell'attributo
Produttore:	YAGEO
Categoria prodotto:	Resistori rilevatori di corrente - SMD
RoHS::	<a href="#">Dettagli</a>
Serie:	PU
Resistenza:	1 mOhms
Potenza nominale:	3 W
Tolleranza:	1 %
Coefficiente di temperatura:	75 PPM / C
Codice involucro - poll.:	5931
Codice involucro - mm:	15078
Temperatura di lavoro minima:	- 65 C
Temperatura di lavoro massima:	+ 275 C



## 7. Strumentazione INA

---

Valori dei parametri ed espressione delle grandezze

```
Res_T_PT = 0; Res_0_PT = 100; //parametri termoresistenza
a_PT = 0.0039083; b_PT = -0.0000005775;

I_PT = 0.001; V_PT = 0; //corrente e tensione termoresistenza

Res_T_PT = Res_0_PT * (1 + a_PT*theta + b_PT*theta*theta); //termoresistenza

V_PT = Res_T_PT * I_PT; //caduta di tensione sulla termoresistenza
```

Dal file di simulazione *simu\_celle*

# 7. Strumentazione INA

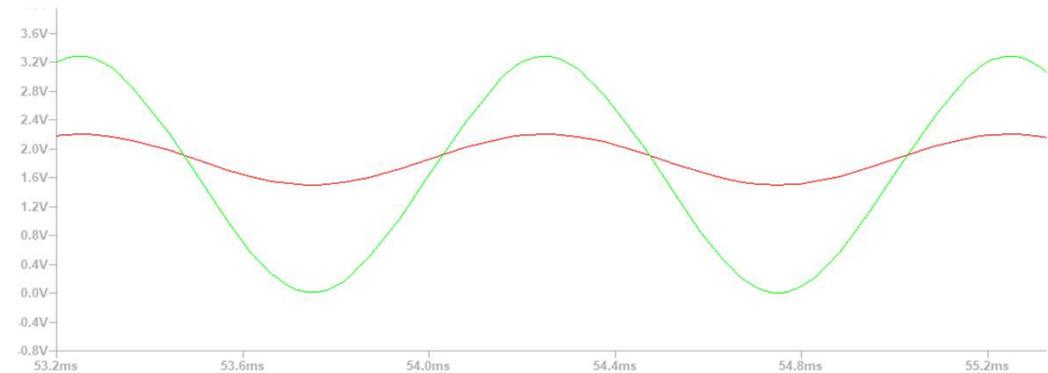
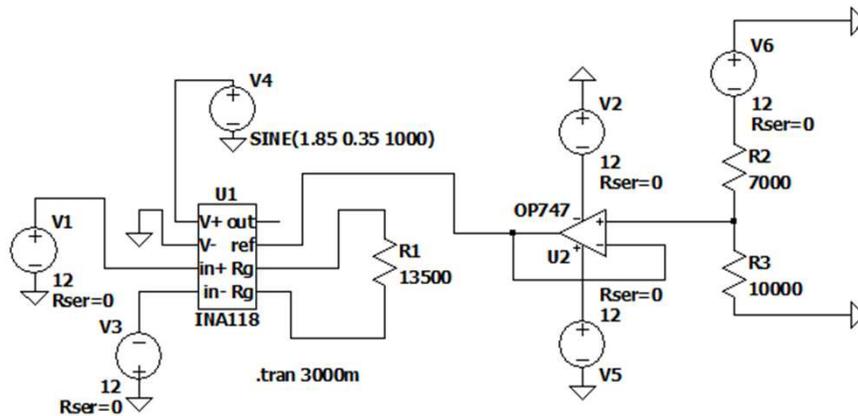
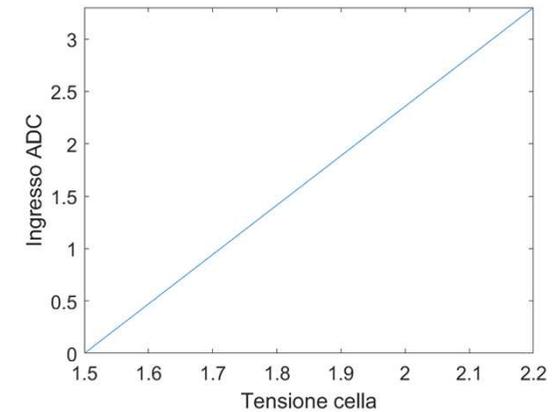
Circuito di adattamento per l'acquisizione della tensione delle celle

$$G = \frac{ADC_{range}}{\Delta V_{celle}} = \frac{3,3V}{2,2V - 1,5V} = 4,7143$$

$$V_{offset} = -GV_{min} = -7,071V$$

$$R_G = \frac{50k\Omega}{4,7143 - 1} = 13461\Omega$$

$$R_2 = \frac{12V10k\Omega - 7,071V10k\Omega}{7,071V} = 6971\Omega$$



Valori sperimentali:  $R_G = 13500\Omega$        $R_2 = 7000\Omega$

Simulazione LTSpice

# 7. Strumentazione INA

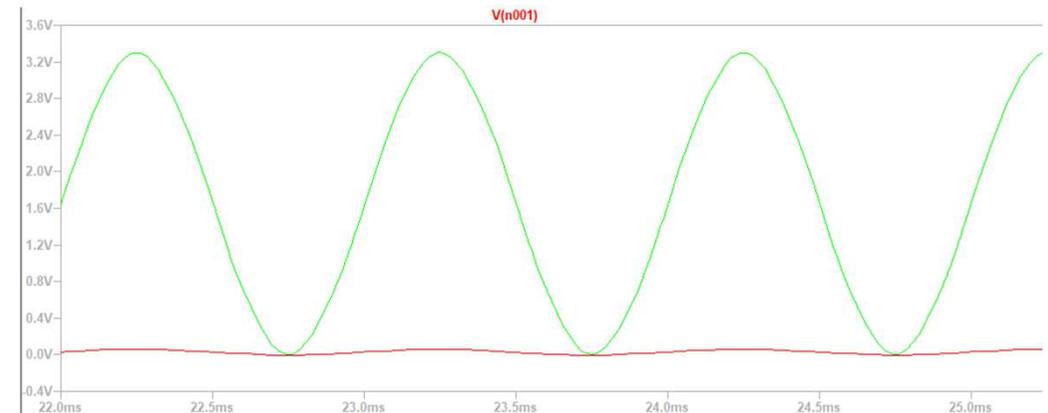
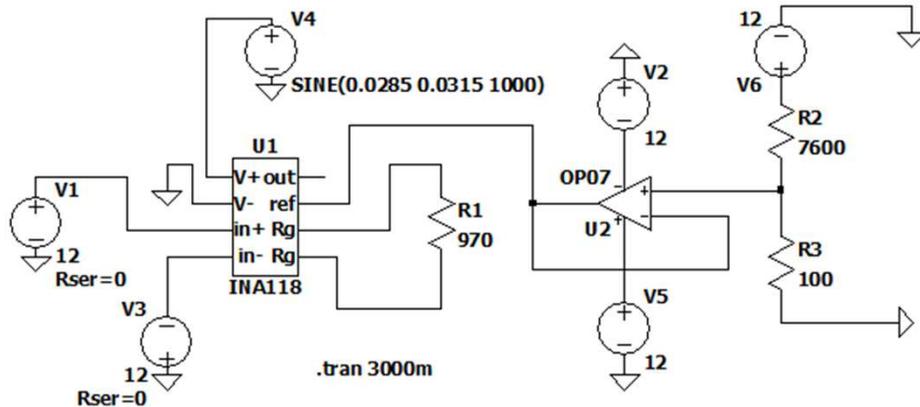
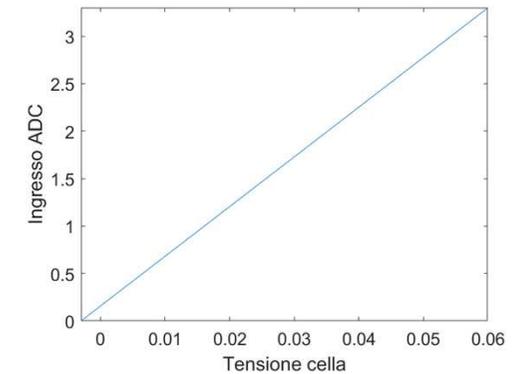
Circuito di adattamento per l'acquisizione della corrente

$$G = \frac{ADC_{range}}{\Delta V_{celle}} = \frac{3,3V}{0,06V + 0,003V} = 52,38$$

$$V_{offset} = -GV_{min} = 0,1571V$$

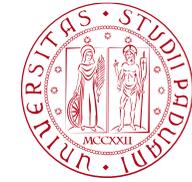
$$R_G = \frac{50k\Omega}{55 - 1} = 973\Omega$$

$$R_2 = \frac{12V100\Omega - 0,1571V100\Omega}{0,1571V} = 7538\Omega$$



Valori sperimentali:  $R_G = 970\Omega$        $R_2 = 7600\Omega$

Simulazione LTSpice



# 7. Strumentazione INA

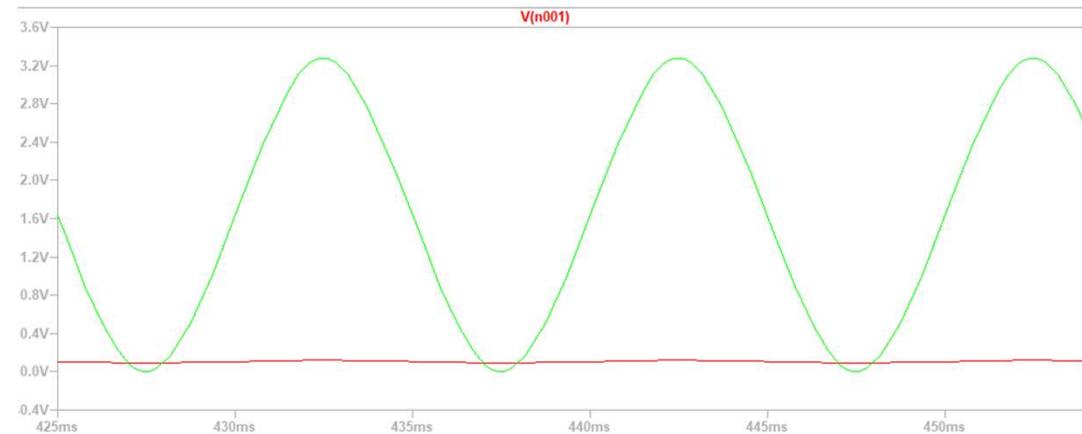
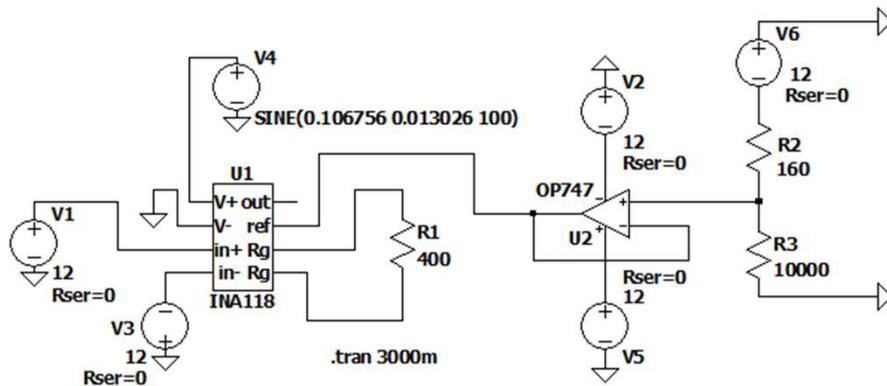
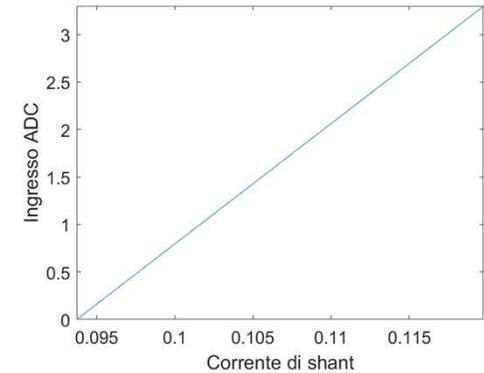
Circuito di adattamento per l'acquisizione della temperatura

$$G = \frac{ADC_{range}}{\Delta V_{celle}} = \frac{3,3V}{0,11978V - 0,09373V} = 126,6795$$

$$V_{offset} = -GV_{min} = -11,8737V$$

$$R_G = \frac{50k\Omega}{126,6795 - 1} = 398\Omega$$

$$R_2 = \frac{12V10k\Omega - 11,8737V10k\Omega}{11,8737V} = 106\Omega$$



Valori sperimentali:  $R_G = 400\Omega$        $R_2 = 160\Omega$

Simulazione LTSpice

# 7. Strumentazione INA



## Adattamento segnali nel modello

$k_{\text{adapt}} = 4.7037$ ;  $q_{\text{adapt}} = -7.0588$ ;

//parametri adattamento tensioni

$k_{\text{PT}} = 126.0$ ;  $q_{\text{PT}} = -11.8110$ ;

//parametri adattamento temperatura

$k_{\text{shunt}} = 52.3875$   $q_{\text{shunt}} = 0.1558$ ;

//parametri adattamento corrente

$\text{ADC1\_IN0} = V_{\text{shunt}} * k_{\text{shunt}} + q_{\text{shunt}}$ ;

//adattamento corrente

$\text{ADC1\_IN1} = V_1 * k_{\text{adapt}} + q_{\text{adapt}}$ ;

//adattamento tensioni

$\text{ADC1\_IN2} = V_2 * k_{\text{adapt}} + q_{\text{adapt}}$ ;

$\text{ADC1\_IN3} = V_3 * k_{\text{adapt}} + q_{\text{adapt}}$ ;

$\text{ADC1\_IN4} = V_4 * k_{\text{adapt}} + q_{\text{adapt}}$ ;

$\text{ADC2\_IN9} = V_{\text{PT}} * k_{\text{PT}} + q_{\text{PT}}$ ;

//adattamento temperatura

Dal file di simulazione *simu\_celle*

## 8. Algoritmo di carica

Tecniche di ricarica delle batterie al piombo [6]

Metodi convenzionali

- ✓ Affidabilità
- ✗ Velocità

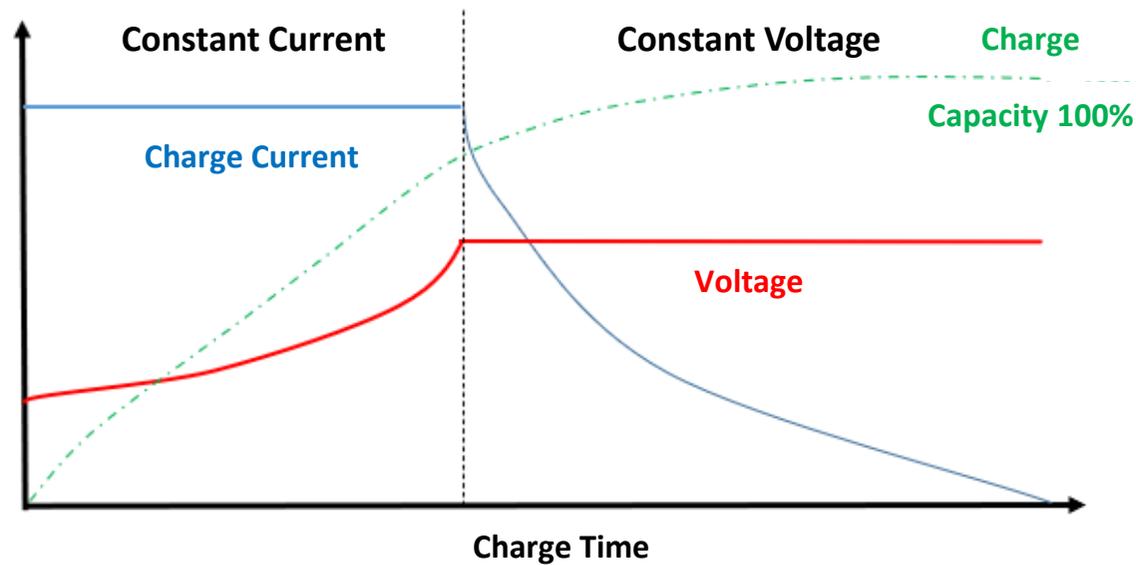
Ricarica ad impulsi

- ✓ Efficienza
- ✗ Complessità

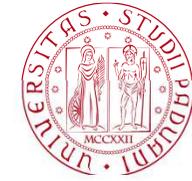
Ricarica intermittente

- ✓ Vita della batteria
- ✗ Complessità

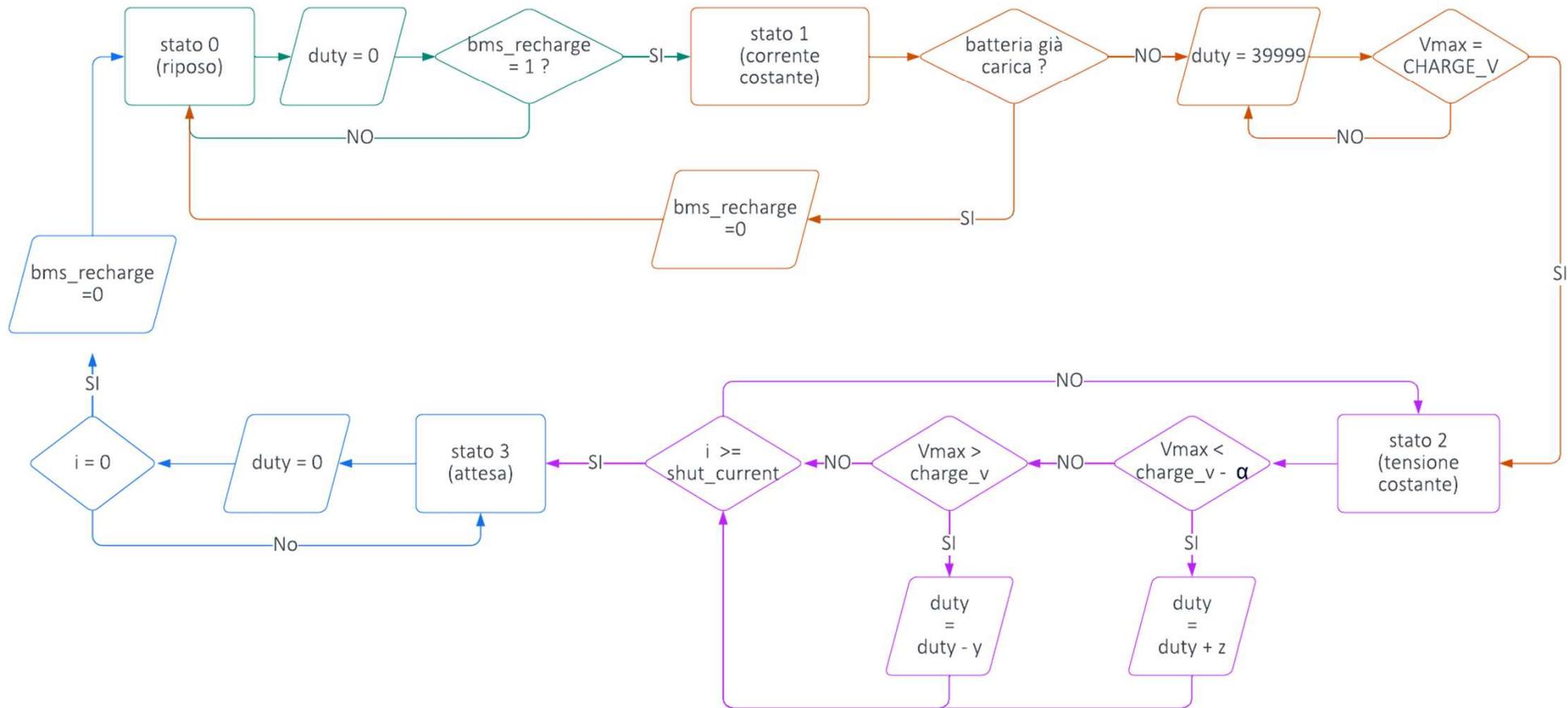
Metodo a due fasi



# 8. Algoritmo di carica



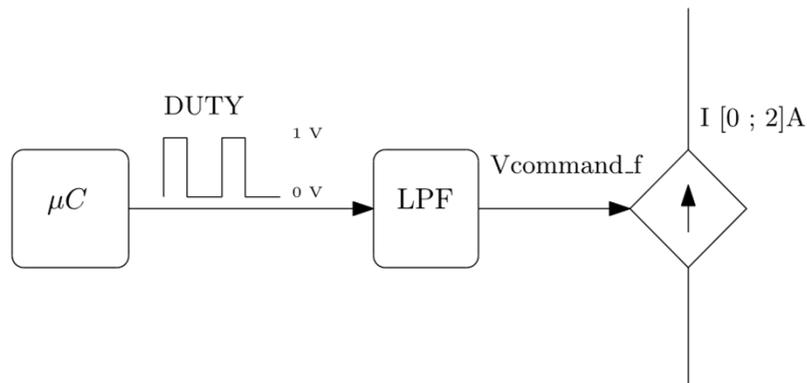
## Implementazione metodo a due fasi (CCCV)



# 8. Algoritmo di carica

## Implementazione generatore di corrente

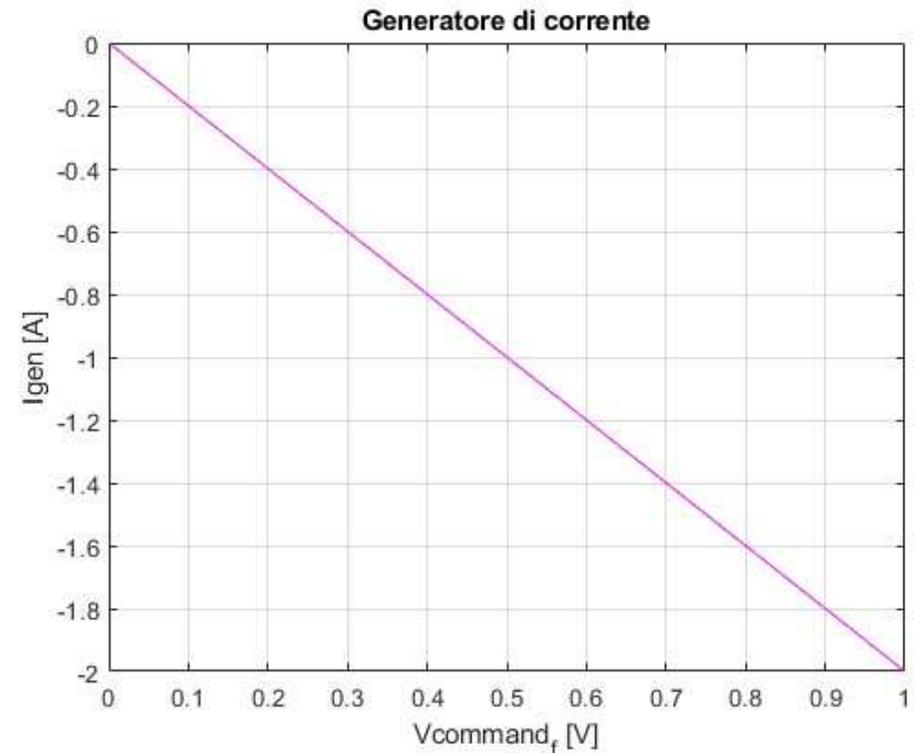
Generatore di corrente pilotato in tensione

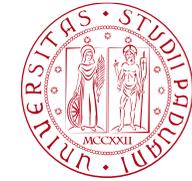


$$V_{command\_f}(k) = \frac{T_{sample}}{\tau + T_{sample}} \cdot DUTY(k) + \frac{T_{sample}}{\tau + T_{sample}} \cdot V_{command\_f}(k-1)$$

$$f_T = \frac{1}{2\pi\tau} = 0.01 \text{ Hz} \rightarrow \tau = RC \cong 15,9 \rightarrow R = 51 \text{ k}\Omega, C = 330 \text{ }\mu\text{F}$$

$$I_{gen} = k_{l\_gen} \cdot V_{command\_f} + q_{gen} \quad k_{l\_gen} = -2 \quad q_{gen} = 0$$





## 8. Algoritmo di carica

### Algoritmo a stati implementato

```
//Costanti
#define OVERVOLTAGE_D 3715 // = 2.135V
#define OVERVOLTAGE_C 3803 // = 2.15V
#define MAX_DUTY 39999
#define CHARGE_V 3773 // = 2.145V
#define SHUT_CURRENT 128 // = -1A

//Riposo
if(state == 0) {
    first_run = true;
    DUTY = 0;

    if(BMS_RECHARGE == 1) {
        state = 1; // Passaggio allo stato 2
    }
}
```

```
//Corrente costante
else if(state == 1) {

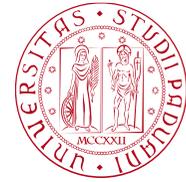
    if(BMS_RECHARGE == 0) // Salto allo stato 3
        state = 3;

    if(Vmax >= OVERVOLTAGE_D && first_run == true) {
        BMS_RECHARGE = 0;
        state = 0; // Ritono allo stato 0
    }
    else {
        DUTY = MAX_DUTY;
        if(Vmax >= CHARGE_V) // Passaggio allo stato 2
            state = 2;
    }
    first_run = false;
}
```

Dal file stm32f1xx\_IT

## 8. Algoritmo di carica

---



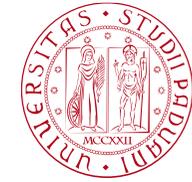
```
//Tensione costante
else if(state == 2) {
    if(BMS_RECHARGE == 0 || ADC1_I >= SHUT_CURRENT)
        state = 3;           // Passaggio allo stato 3

    //Isteresi
    if(Vmax >= (CHARGE_V)) {
        if(DUTY <= 0)         DUTY = 0;
        else                  DUTY = DUTY - 150;
    }
    else if(Vmax <= (CHARGE_V - 7)) {
        if(DUTY >= MAX_DUTY) DUTY = 39999;
        else                  DUTY = DUTY + 20;
    }
}
```

```
//Attesa
else if(state == 3) {
    DUTY = 0;

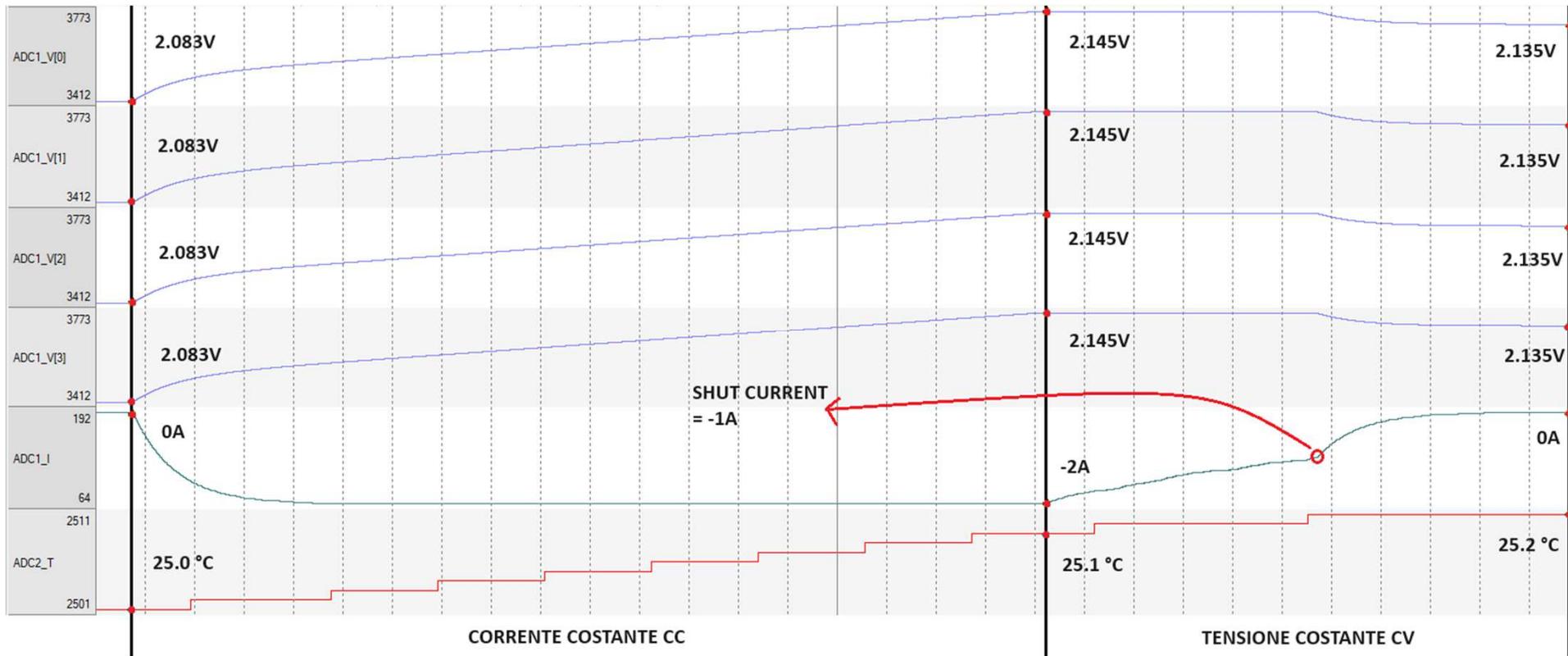
    //Ritorno allo stato 0
    if(ADC1_I == 192) {
        state = 0;
        BMS_RECHARGE = 0;
    }
}
```

# 8. Algoritmo di carica

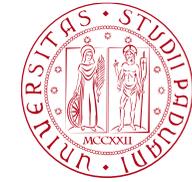


## Simulazione algoritmo

SOC: 70% → 100%

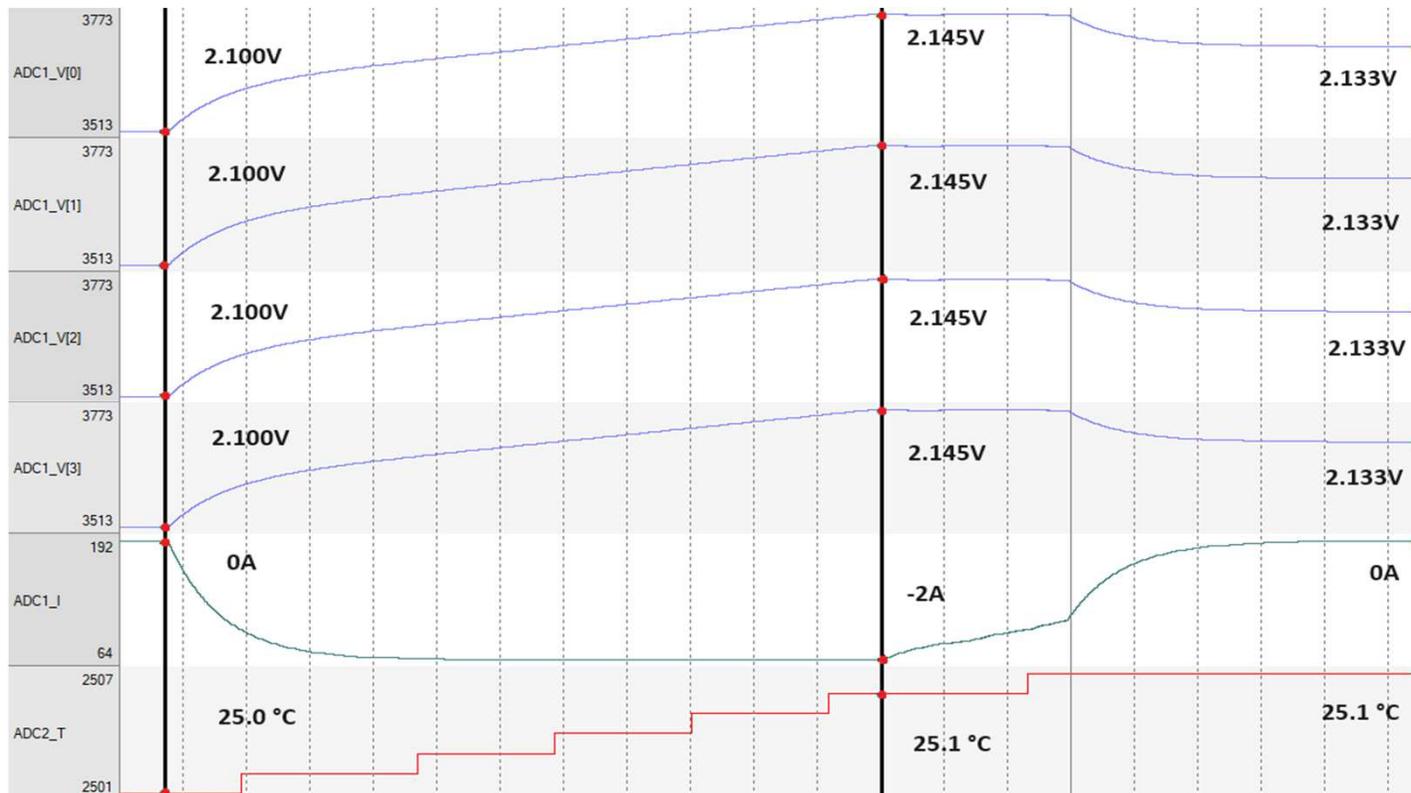


# 8. Algoritmo di carica



## Simulazione algoritmo

Dal 80% del SOC, fino al distacco della ricarica, prima di arrivare al completamento della carica della batteria



## 9. Conclusioni

---



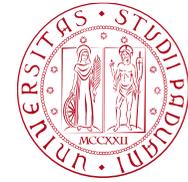
Sfruttando l'ambiente di sviluppo di Mxcube, Keil uVision e il modello elettrico equivalente da noi selezionato siamo stati in grado di implementare un sistema BMS funzionante, il quale non solo è in grado di effettuare un bilanciamento delle celle tramite la tecnica di tipo passivo, ma anche di segnalare eventuali anomalie tramite un sistema di allarmi ed eseguire la ricarica delle celle sfruttando un opportuno algoritmo.

Il progetto è poi completo anche delle indicazioni degli adattamenti dei segnali in ingresso che vengono utilizzati.

Si sottolinea inoltre che data la scelta del modello è stato possibile integrare, con discreta facilità, anche il modello termico della batteria in maniera molto agevole in quanto già fornito dal modello utilizzato.

# 10. Bibliografia

---



- [1] M. Ceraolo, "New dynamical models of lead-acid batteries," in IEEE Transactions on Power Systems.
- [2] H. Wang, G. Li, M. Li, Z. Jiang, X. Wang and Q. Zhao, "Third-order dynamic model of a lead acid battery for use in fuel cell vehicle simulation," 2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC), Jilin, China, 2011.
- [3] S. Dalvi and S. Thale, "Design of DSP Controlled Passive Cell Balancing Network based Battery Management System for EV Application," 2020 IEEE India Council International Subsections Conference (INDISCON), Visakhapatnam, India, 2020.
- [4] M. Tyrpekl, M. Zavrel and V. Kindl, "Comparison of active and passive battery balancing," 2022 20th International Conference on Mechatronics - Mechatronika (ME), Pilsen, Czech Republic, 2022.
- [5] H. Weiss and S. Volgger, "Insular power supply battery manager with balancing and protection unit for Lithium-Ion batteries," 2014 12th International Conference on Actual Problems of Electronics Instrument Engineering (APEIE), Novosibirsk, Russia, 2014.

# 10. Bibliografia

---



- [6] Chih-Chiang Hua and Meng-Yu Lin, “A Study of Charging Control of Lead-Acid Battery for Electric Vehicles” ISIE'2000, Cholula, Puebla, Mexico, ISIE'2000.
- [7] CSB energy technology CO., HR 1221W, “Valve Regulated Lead Acid Battery”, China.
- [8] Texas Instruments, INA118 precision, “Low-Power Instrumentation Amplifier”, Stati Uniti, Dallas, Texas.
- [9] Yageo electronics, PU series, “SHUNT RESISTOR AUTOMOTIVE GRADE”, Italy.
- [10] TE connectivity, “Platinum Thin Film (PTF) Temperature Elements”, Svizzera, Sciaffusa



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA