

## Rapporto tecnico n. 426 anno 2019

---

# Implementazione con scheda STM32 NUCLEO® di un algoritmo per la modulazione della larghezza d'impulso di tipo casuale (Random PWM)

*Autori:*

**G. Marsala<sup>1</sup>, V. Presti<sup>1</sup>, A. Sauro<sup>1</sup>, S. G. Scordato<sup>1</sup>, G. Vitale<sup>2</sup>**

<sup>1</sup>INM – Institute of Marine Engineering  
National Research Council of Italy

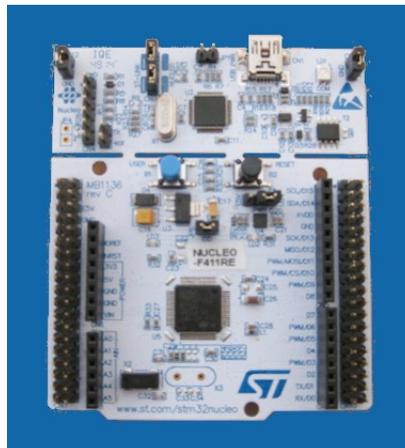
<sup>2</sup>ICAR – Institute for High Performance  
Computing and Networking  
National Research Council of Italy

***“Esemplare fuori commercio per il deposito legale agli effetti della Legge 15 aprile 2004, n. 106”***

*Gli autori sono i soli responsabili del contenuto di questo rapporto tecnico*

Data di creazione: 30 maggio 2019

Prima distribuzione al pubblico e deposito legale: 30 maggio 2019



---

Sede principale: Via di Vallerano, 139 – 00128 Roma  
P. IVA 02118311006 – C.F. 80054330586

E-mail: [segreteria.inm@cnr.it](mailto:segreteria.inm@cnr.it)

PEC: [protocollo.inm@pec.cnr.it](mailto:protocollo.inm@pec.cnr.it)

Tel. 06-50299222 - Fax 06-5070619

<http://www.inm.cnr.it>

---

Sede di Roma "Sezione di Acustica e Sensoristica O.M. Corbino"  
Area della Ricerca di Tor Vergata  
Via del Fosso del Cavaliere, 100 - 00133 Roma

Sede di Genova  
Area della ricerca di Genova  
Via De Marini, 6 –16149 Genova

Sede di Palermo  
Area della ricerca di Palermo  
Via Ugo La Malfa, 153 –90146 Palermo

## INDICE

1	INTRODUZIONE.....	1
2	DESCRIZIONE DEL CONTROLLO.....	2
2.1	SIMULAZIONE DEI TRACCIATI.....	3
3	CARATTERISTICHE DEL SEGNALE.....	5
3.1	SEGNALE PWM A FREQUENZA FISSA.....	5
3.2	SEGNALE PWM A FREQUENZA VARIABILE.....	5
4	CRITERIO DI SCELTA DEL CONTROLLO – IL FENOMENO DI ONDULAZIONE...7	
4.1	ANALISI SPETTRALE.....	7
4.2	METODO E STRUMENTO ADOTTATO PER L'ACQUISIZIONE DEI DATI.....	8
5	PSEUDO-CODIFICA DELL'ALGORITMO.....	11
6	INDIVIDUAZIONE DEL SEGNALE PWM E DEGLI INGRESSI.....	13
	APPENDICE.....	14

## 1 INTRODUZIONE

La tecnica di modulazione dei convertitori statici di potenza è fondamentale per garantire le prestazioni richieste e per minimizzare le emissioni elettromagnetiche. Infatti utilizzando una frequenza portante fissa si generano armoniche multiple della fondamentale sovrapposte alla tensione di uscita; se invece si fa variare la frequenza della portante le emissioni sono distribuite più uniformemente nello spettro di frequenze e si limita il contributo di singole frequenze.

In commercio esistono vari circuiti integrati in grado di generare un segnale PWM con portante a frequenza fissa, tuttavia se si vuole studiare il comportamento facendo variare la frequenza in modo casuale è necessario implementare un algoritmo possibilmente su un sistema a basso costo.

In questo rapporto tecnico sono descritti i metodi di simulazione e sperimentali adottati per implementare il controllo della commutazione di un convertitore DC-DC, di tipo *elevatore (Boost)* realizzato in laboratorio. La prassi adottata in tali metodi ha previsto la programmazione di un micro controllore ( $\mu C$ ), per la generazione del segnale di controllo. In particolare saranno confrontate due tipologie di controllo, denominate rispettivamente "*tecnica di controllo con segnale PWM classico*" e "*tecnica di controllo con segnale Random PWM*". Tale confronto viene effettuato sull'analisi del fenomeno della ondulazione (*Ripple*), presente nella tensione e nella corrente di uscita ai convertitori.

Il codice è stato implementato in modo da poter variare quando funziona in modalità random, sia la frequenza di scostamento rispetto alla frequenza centrale fissata pari a 20 kHz, sia il numero di impulsi di pilotaggio da applicare al *gate dell'IGBT* prima di aggiornare il valore della frequenza di switching secondo un numero random. La disponibilità di questi due gradi di libertà sarà utilizzata per minimizzare la densità spettrale della potenza, all'uscita del convertitore nel campo delle emissioni condotte.

## 2 DESCRIZIONE DEL CONTROLLO

Tecnicamente i convertitori DC-DC sono dispositivi in grado di convertire energia elettrica in tensione a regime continuo, proporzionale alla tensione pure a regime continuo applicata al suo ingresso, la quale fornisce energia al convertitore stesso. Il loro principio di funzionamento prevede infatti che, attraverso un dispositivo di commutazione, le fasi di carica e scarica dell'energia si alternano: quando vengono raggiunte le condizioni di regime la tensione all'uscita ed anche la corrente in valore medio risulta nulla, quindi la tensione di uscita può essere considerata costante. In genere la funzione di immagazzinamento dell'energia è affidata ad un induttore.

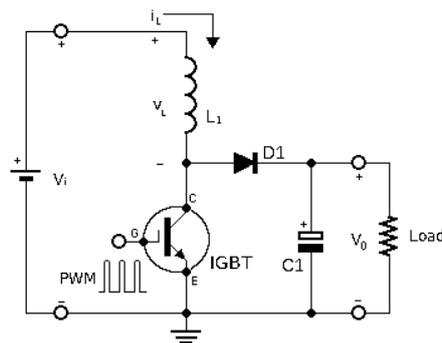


Figura 1: Schema circuitale di un convertitore di tipo *Boost*

Con riferimento al convertitore di tipo Boost o step-up classico mostrato in figura 1, dove il segnale PWM è inviato al *gate* di un *transistor a giunzione con gate isolato (IGBT)*, il valore medio della corrente che circola sull'induttore  $L_1$  è nulla; risultando quindi nullo il seguente integrale nell'intervallo temporale  $T$ , in cui avviene una commutazione (corrispondente all'inverso della frequenza di *switching*).

$$\frac{1}{T} \int_0^T v_{L(t)} dt = \frac{1}{T} \int_0^{t_{ON}} v_{i(t)} dt + \frac{1}{T} \int_{t_{ON}}^T (v_{i(t)} - v_{o(t)}) dt = 0$$

$$T = t_{ON} + t_{OFF}$$

$t_{ON}$  e  $t_{OFF}$  sono i tempi di permanenza nello stato *ON* ed *OFF* della commutazione; mentre  $v_{i(t)}$  e  $v_{o(t)}$  sono le tensioni di ingresso e di uscita al convertitore.

Nelle l'ipotesi che le tensioni sia d'ingresso che d'uscita siano costanti ( $v_{i(t)} = V_i$ ,  $v_{o(t)} = V_o$ ), sono soddisfatte le seguenti equazioni.

$$V_i \frac{t_{ON}}{T} + V_i \frac{t_{OFF}}{T} - V_o \frac{t_{OFF}}{T} = V_i D + V_i(1-D) - V_o(1-D) = 0$$

$$\frac{t_{ON}}{T} = D; \quad \frac{t_{OFF}}{T} = \frac{T-t_{ON}}{T} = 1-D$$

Il parametro  $D$  è il *Duty Cycle* dei segnali *modulati in larghezza d'impulso (Pulse Width Modulation)*.

Da tale risultato è quindi provata la proporzionalità tra la tensione  $V_i$  e la tensione  $V_o$ , espressa con la seguente equazione.

$$V_o = V_i \frac{1}{1-D}$$

La commutazione controllata con questi segnali è definita quindi "*tecnica di controllo PWM*".

## 2.1 SIMULAZIONE DEI TRACCIATI

Sulla base delle equazioni sopra descritte è possibile ricavare, a partire dal livello di tensione sul gate dello switch, la corrente sull'induttore e la tensione di uscita.

Nella seguente figura 2 sono illustrate rispettivamente le tracce della differenza di potenziale  $v_L$  e della corrente  $i_L$  che circola nell'induttore  $L_1$ ; le tracce sono ottenute simulando una tensione di ingresso  $V_i$  pari a 20 V ed un induttore da 1 mH, in un intervallo temporale di 150  $\mu$ s. In figura 3 è illustrata la traccia del ripple, sovrapposta alla tensione continua di uscita  $V_o$ <sup>1</sup>; anch'essa è ottenuta in simulazione, con un carico ohmico in uscita di 92  $\Omega$  ed una capacità di 100  $\mu$ F, per ridurre sensibilmente il ripple filtrando la frequenza di switching.

<sup>1</sup> Le tracce delle rispettive figure sono ottenute con l'ambiente di simulazione *simulink*, sviluppato per Matlab®. Si noti l'effetto della discretizzazione delle tracce stesse, dove il tempo di campionamento è stato impostato a 4 $\mu$ s.

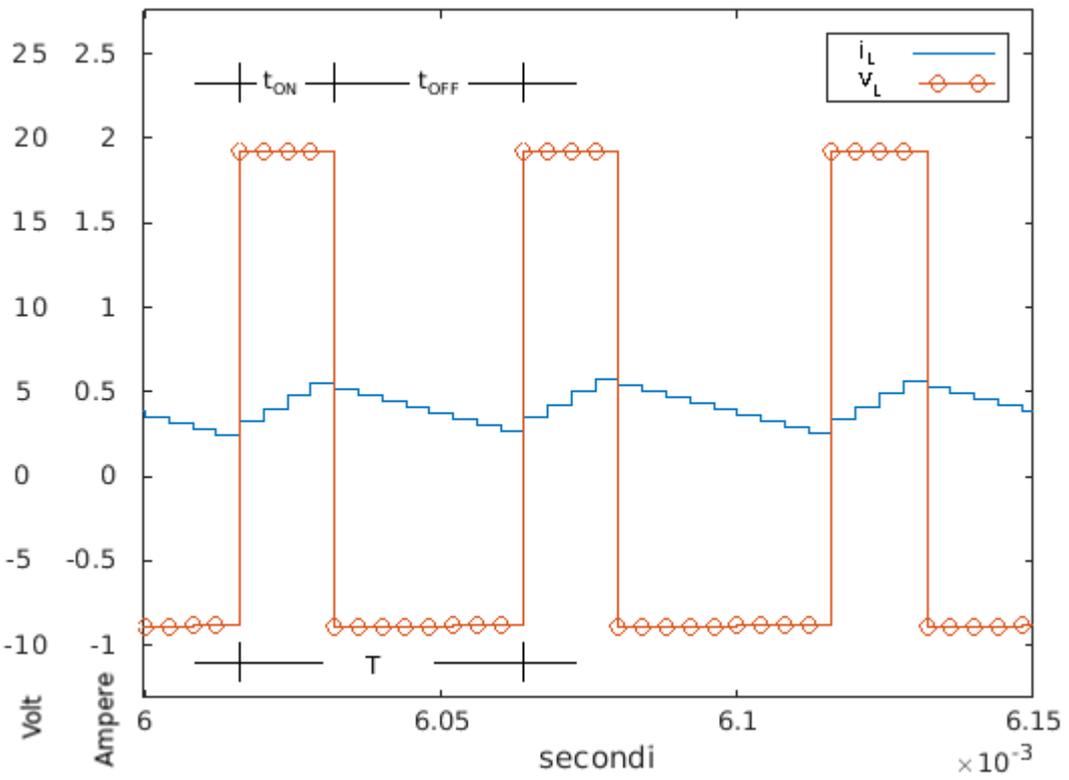


Figura 2

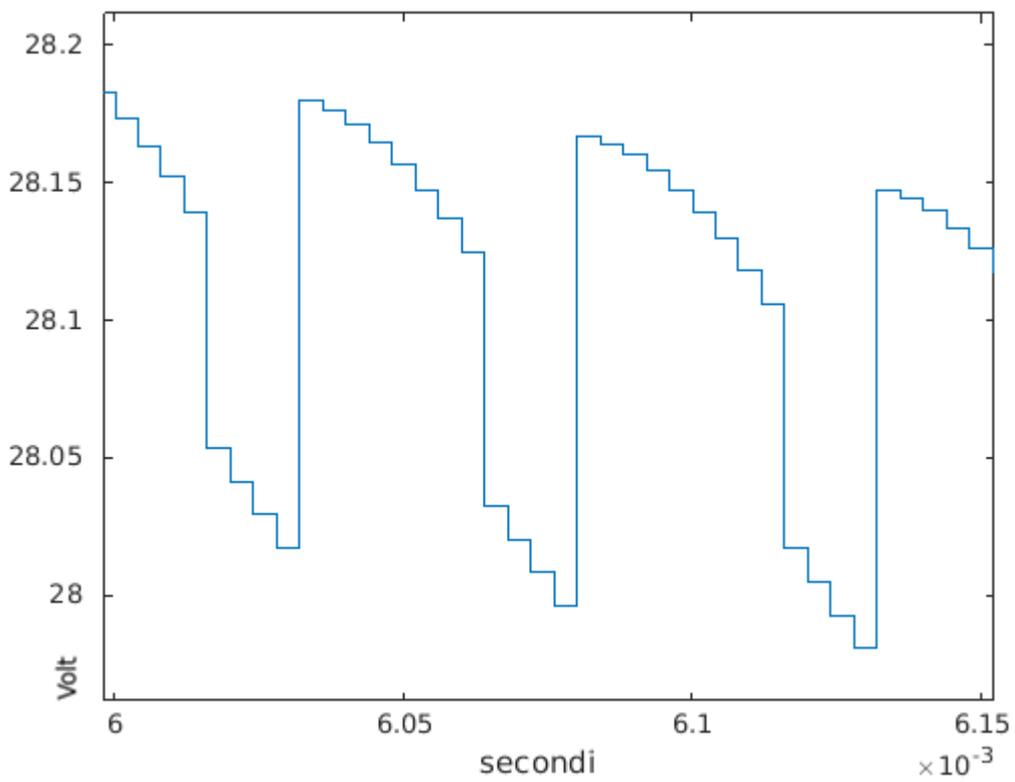


Figura 3

### 3 CARATTERISTICHE DEL SEGNALE

Per l'implementazione del controllo con segnale PWM a frequenza fissa oppure variabile, è stato programmato un micro controllore ( $\mu\text{C}$ ) in modo da generare un segnale con specifiche caratteristiche, di seguito descritte.

#### 3.1 SEGNALE PWM A FREQUENZA FISSA

La tipologia di controllo con segnale PWM classico prevede la generazione di impulsi con frequenza fissa a 20KHz, e larghezza degli impulsi con fattore  $D$  pari a 0,30 (Figura 4).

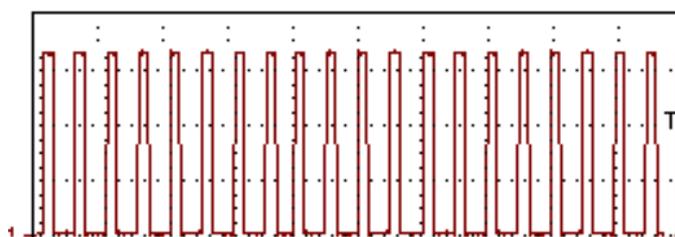


Figura 4

#### 3.2 SEGNALE PWM A FREQUENZA VARIABILE

Per la tipologia di controllo con segnale PWM a frequenza variabile viene generato un treno di impulsi dove, mantenendo fisso il fattore di larghezza degli impulsi  $D$  pari a 0,30 la, frequenza varia in modo casuale con larghezza di banda  $B$ , centrata nella frequenza di 20KHz con scostamento di  $\pm 10\text{KHz}$ . In sostanza viene generato un treno di impulsi, caratterizzato dai seguenti fattori:

- Fissando la larghezza degli impulsi mantenendo costante il parametro  $D$ , viene stabilito che il periodo  $T$  dovrà *variare in modo casuale (Random)*, fissando i limiti della banda  $B$  in frequenza.

- Essendo variabile il periodo  $T^2$ , il suo valore viene aggiornato ad ogni ripetizione di un *numero fissato di impulsi* (come in figura 5, il periodo varia ogni 8 impulsi).

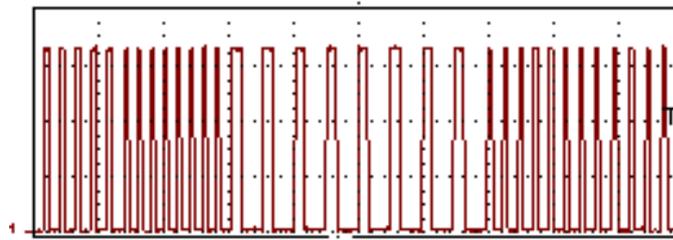


Figura 5

---

2 L'ordine di grandezza del periodo è determinato dalla banda in frequenza del treno di impulsi.

## 4 CRITERIO DI SCELTA DEL CONTROLLO – IL FENOMENO DI ONDULAZIONE

L'alternanza carica - scarica dell'energia, causata dalla componente alternata della corrente sulla capacità di uscita del convertitore, crea il fenomeno di *ondulazione (ripple)* alle alte frequenze sulla tensione. A causa di ciò si ha la presenza delle *armoniche della fondamentale* nello spettro in frequenza. Tali armoniche aggiungono il proprio contributo in termini di potenza dissipata dal carico, collegato all'uscita del convertitore: tramite l'analisi della *trasformata di Fourier* della tensione di uscita si ottiene quindi la misura della potenza, su cui si basa il criterio di scelta del controllo.

### 4.1 ANALISI SPETTRALE

In figura 6 è rappresentato lo spettro di un segnale in onda quadra con frequenza di 20KHz e coefficiente  $D$  pari a 0,30, con banda di 250MHz in scala logaritmica. L'armonica con modulo massimo è di frequenza pari a 20,8KHz ed il modulo espresso in  $\text{dB}_{\text{Vrms}}$  è pari a 1,18dB. In termini di energia, nell'ipotesi che il segnale con tale spettro è applicato ad un dispositivo dotato di un carico ohmico di  $50\Omega$ , l'armonica con modulo massimo contribuisce con una potenza di  $P_{\text{dBmW}} = \text{dB}_{\text{Vrms}} + 10\text{Lg}_{10}(20) = 1,18 + 13,01 = 14,19\text{dB}_{\text{mW}}$ ; e la potenza sarà pari a  $P_{\text{mW}} = 10^{\text{dBmW}/10} = 10^{1,419} = 26,24\text{mW}$ .

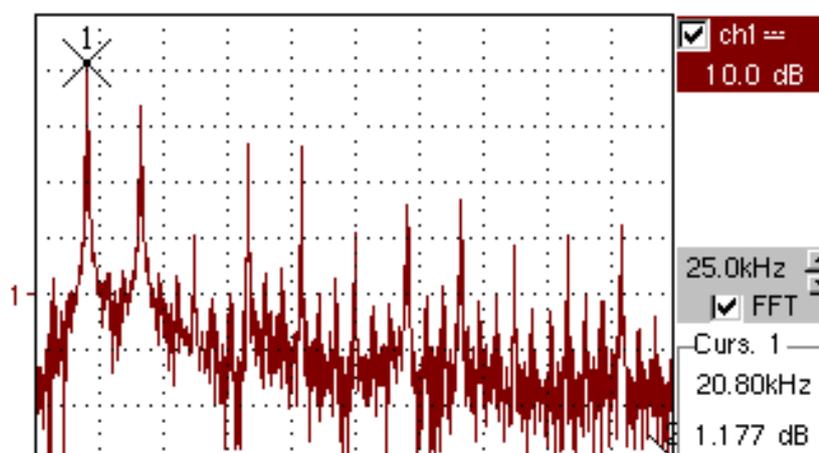


Figura 6

In figura 7 è rappresentato lo spettro di un segnale generato con modalità casuale, pure con banda di 250MHz in scala logaritmica; l'armonica con modulo massimo è di frequenza pari a 10KHz e modulo pari a -8,18dB. Applicando quindi lo stesso metodo di calcolo e ipotizzando pari a 50Ω il carico, il contributo energetico dell'armonica è pari a  $\text{dB}_{\text{mW}} = -8,18 + 13,01 = 4,83\text{dB}_{\text{mW}}$ ; ed in questo caso la potenza è pari a  $P_{\text{mW}} = 10^{4,83/10} = 3,04\text{mW}$ .

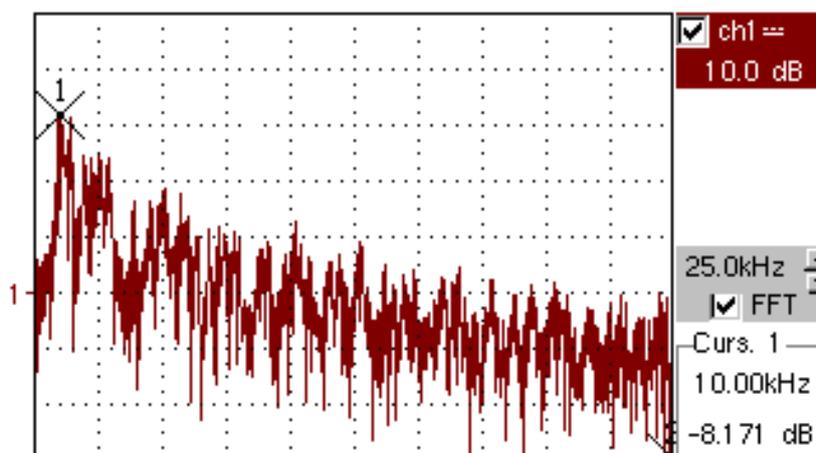


Figura 7

Questa variazione dello spettro si ottiene modificando il periodo del segnale in onda quadra ogni 10 impulsi.

In base ai precedenti valori *quantitativi* di potenza, il criterio di scelta *individua* il treno di impulsi (in questo caso 10 impulsi), come segnale di controllo che riduce sensibilmente il contributo alla potenza del ripple, presente nella tensione di uscita.

## 4.2 METODO E STRUMENTO ADOTTATO PER L'ACQUISIZIONE DEI DATI

L'acquisizione dei segnali e degli spettri è stata eseguita con l'oscilloscopio OX 7062 della Metrix®, il quale è in grado di eseguire comandi e rispondere alle interrogazioni *ricevute dai terminali a cui è connesso*. Strumenti di misura e di acquisizione dotati di tali funzionalità sono detti *strumenti programmabili*.

Nello specifico delle misure e delle acquisizioni eseguite, lo strumento è stato gestito dal terminale di un calcolatore remoto con sistema operativo basato su UNIX, trasmettendo i comandi tramite la porta seriale con il comando *echo*.

I comandi e le interrogazioni sono stringhe alfanumeriche, formulate in codice ASCII e trasmesse tramite porta seriale; oppure attraverso l'interfaccia di rete, secondo il protocollo TCP-IP (*Transmission Control Protocol - Internet Protocol*). Nelle seguenti figure 8 e 9 sono rappresentati rispettivamente il menù a tendina e la corrispondente scheda grafica, per l'assegnazione dell'indirizzo IP all'interfaccia di rete dello strumento.

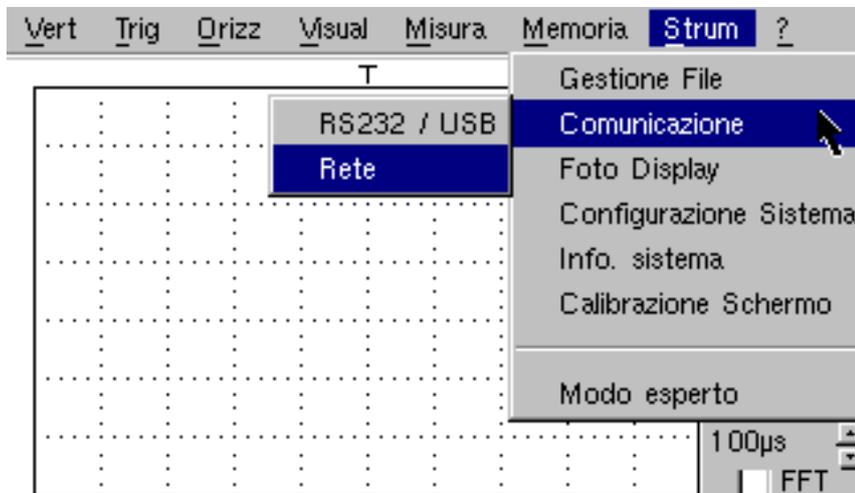


Figura 8

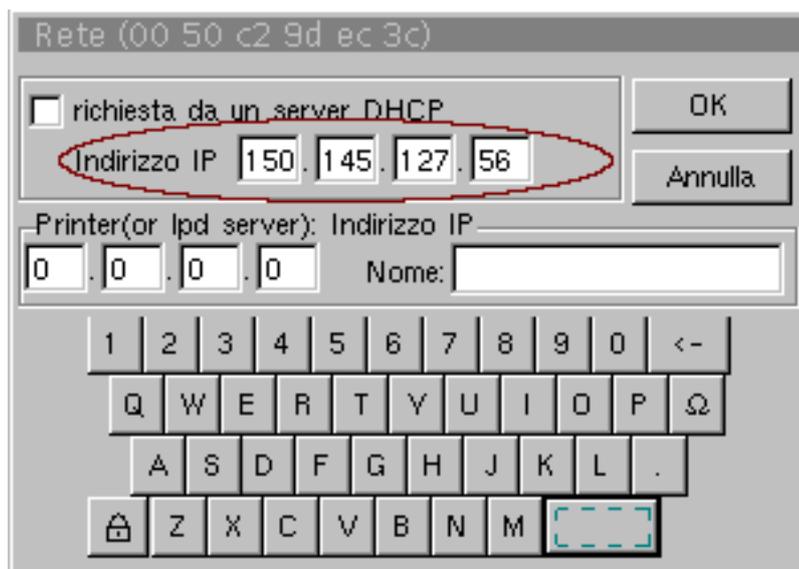


Figura 9

Supponendo che lo strumento sia dunque connesso in rete, le stringhe possono essere trasmesse attivando una *sessione di comunicazione* con il comando *telnet* secondo il protocollo applicativo dei *terminali remoti* (figura 10); attivando invece una sessione di comunicazione secondo il protocollo applicativo per il trasferimento dei

files (*File Transfer Protocol*), con il comando *ftp* è possibile accedere da remoto alla memoria dello strumento, dove vengono immagazzinati i dati delle misure (si veda la stessa figura 10).

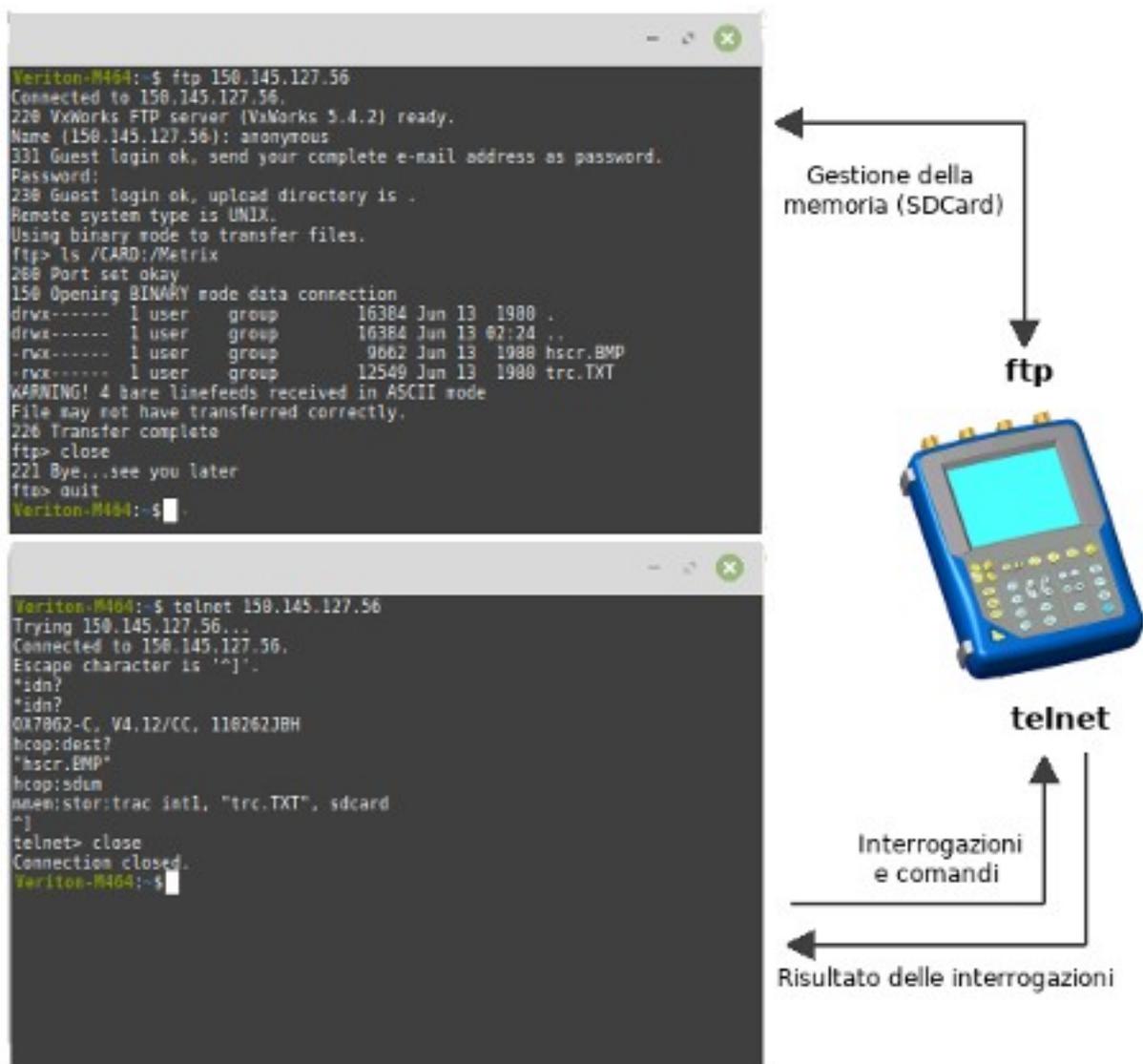


Figura 10

Il sistema di comunicazione appena descritto, come si può chiaramente intuire, permette all'operatore di eseguire in sicurezza le misure di corrente e tensione, anche nel caso in cui ad esempio i livelli delle grandezze elettriche risultino pericolosi per l'operatore stesso.

## 5 PSEUDO-CODIFICA DELL'ALGORITMO

Con un ciclo illimitato il controllore genera una sequenza con un numero finito di impulsi e frequenza variabile, tale da implementare il controllo PWM tramite due *interrupts*, che gestiscono rispettivamente un contatore ed una variabile booleana.

Per la gestione del periodo e del numero di impulsi, tramite il primo interrupt viene incrementato un contatore ad ogni notifica inviata al controllore, che segnala la commutazione del segnale sul fronte di discesa. Ogni volta che il contatore raggiunge il limite stabilito dalle specifiche del controllo, viene ripristinata quindi la condizione iniziale del contatore, ed aggiornato il periodo  $T$  con i limiti stabiliti dalla banda  $B$ .

$$B = f_{max} - f_{min};$$
$$T_{min} = \frac{1}{f_{max}}, \quad T_{max} = \frac{1}{f_{min}}.$$
$$T_{min} \leq T \leq T_{max};$$
$$T_{min} \leq T \leq T_{min} + (T_{max} - T_{min})$$

In base al valore booleano di una variabile gestita dal secondo *interrupt*, il quale segnala l'evento della pressione di un bottone, sarà generato un segnale a frequenza fissa per il controllo con *PWM classico*; oppure a frequenza variabile per il controllo con *Random PWM*. In figura 11 sono schematizzate in linguaggio UML le relative attività.

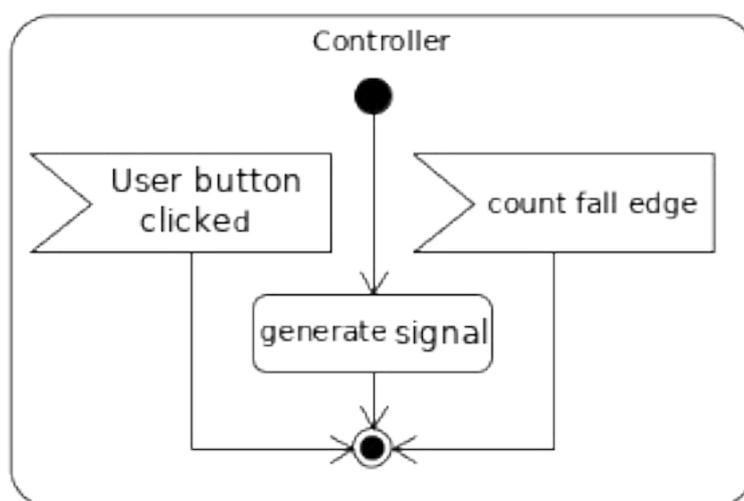


Figura 11

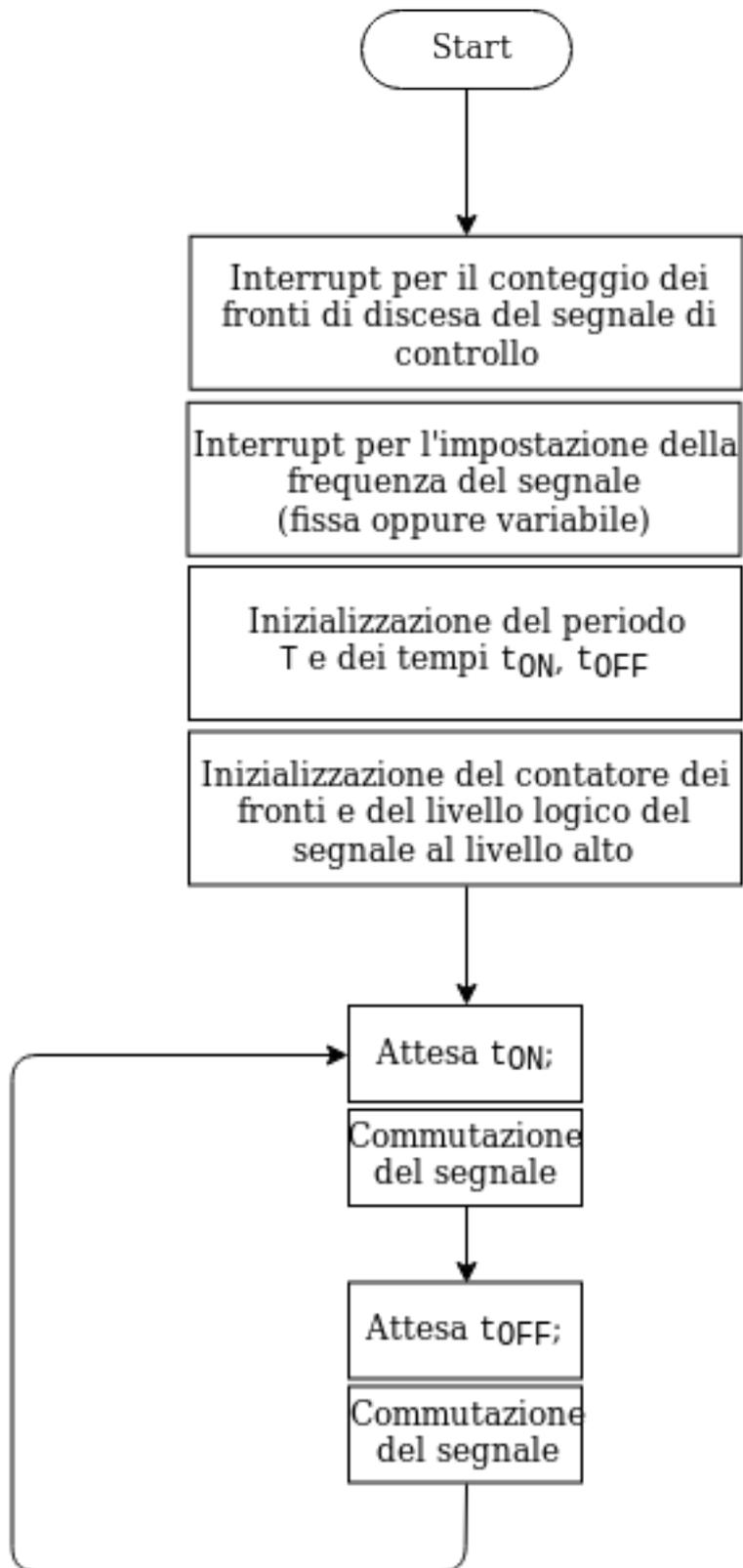


Figura 12

Il diagramma di flusso rappresentato in figura 12 descrive l'implementazione delle attività.

## 6 INDIVIDUAZIONE DEL SEGNALE PWM E DEGLI INGRESSI

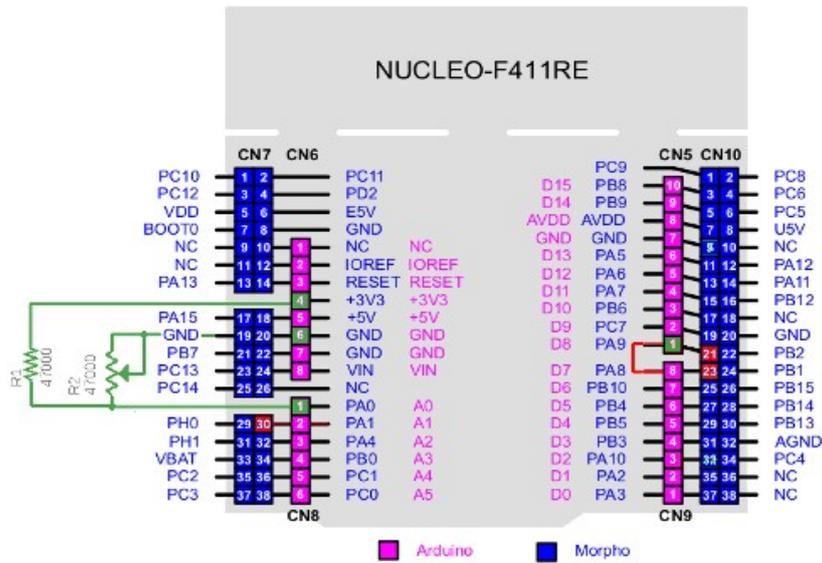


Figura 13

Come è rappresentato in figura 13, lo schema degli ingressi e delle uscite della scheda con  $\mu\text{C}$ , con la relativa interconnessione adottata per la generazione del treno di impulsi. Come schema generale, nella scheda vi sono due file di connettori, dotati di rispettivi piolini con la relativa nomenclatura: i connettori propri della scheda (*Morpho*), ed i connettori specifici della scheda *Arduino*<sup>®</sup>.

In dettaglio, l'interconnessione tra gli ingressi e le uscite della scheda è stata realizzata per ottenere il segnale PWM rispetto a massa all'uscita digitale in *D8* (connettore *CN5*, piolino 1). Il segnale PWM viene inviato dall'uscita di *D8* all'ingresso pure digitale in *D7*, dal piolino 21 al piolino 23 di *CN10*, per il conteggio dei fronti di discesa dello stesso PWM tramite il relativo interrupt. Con un partitore di tensione realizzato con un potenziometro lineare da  $47\text{K}\Omega$  e con una resistenza anch'essa da  $47\text{K}\Omega$ , alimentato con una tensione di  $3,3\text{V}$  rispetto a massa dal piolino 4 del connettore *CN6*, viene regolato un livello di tensione direttamente proporzionale al numero di impulsi che devono essere ripetuti con periodo costante. Tale livello di tensione viene convertito da un convertitore analogico-digitale, inviando il relativo segnale al suo ingresso in *A0* (piolino 1 di *CN8*).

## APPENDICE

In questa sezione è riportato il codice sorgente, con cui è implementato l'algoritmo di generazione del segnale PWM.

```
#include "mbed.h"
#define TIME_T(t) ((time_t *)t)
#define POINTER(p) (&p)

Serial console (SERIAL_TX, SERIAL_RX);

InterruptIn fwait (D7);
// Segnale da cui vengono intercettati
// i fronti di discesa del PWM

InterruptIn btn (USER_BUTTON);
// Segnale da cui vengono intercettati
// i fronti di discesa del segnale
// generato dal tasto USER_BUTTON

AnalogIn setFalls (A0);
// Segnale per l'impostazione del contatore
// dei fronti di discesa generati dal PWM

DigitalOut dOut (D8);
// Segnale con modulazione d'impulso (PWM)
int PwmFall, falls;
// Contatore dei fronti di discesa del PWM

bool fixed;
void doSetState (void)
{
    console.printf ("%d\r\n", (fixed)? 0: falls);
    fixed = !fixed;
    // Aggiornamento della variabile booleana
    // ai fronti di discesa del segnale generato
    // da USER_BUTTON per impostare la generazione
    // del PWM a frequenza fissa oppure variabile
    return;
}

float DC;
int T, deltaT, Tmin, Tmax;
int tON, tOFF;
// Tempi di commutazione tON <-> tOFF
// nel periodo T espressi in microsecondi
// con Duty Cycle pari a DC
```

```

void doSetPwm (void)
{
    for (PwmFall++; PwmFall == falls; PwmFall = 0)
        if (!fixed)
            {
                T = Tmin + rand()%deltaT;
                tON = (int) (DC*T); tOFF = (int)((1 - DC)*T);
                falls = 10 + (int) (184.0f*setFalls.read());
            }
// Aggiornamento sincronizzato del periodo T (Tmin<T<Tmax),
// dei tempi di commutazione e del numero degli impulsi
// ai fronti di discesa del PWM con banda in frequenza
// variabile tra 10KHz e 30KHz in modo casuale
    return;
}

int main (void)
{
    btn.disable_irq ();
    btn.fall (POINTER (doSetState));
    btn.enable_irq ();

    fWait.disable_irq ();
    fWait.fall (POINTER (doSetPwm));
    fWait.enable_irq ();

//Gestione degli interrupts che intercettano
//i fronti di discesa generati dal PWM e
//da USER_BUTTON con il metodo fall ();

    float f0 = 20.0e+3f;
    float deltaf = 10.0e+3f;
    float fmin = f0 - deltaf;
    float fmax = f0 + deltaf;

    Tmin = (int)(1.0e+6f/fmax);
    Tmax = (int)(1.0e+6f/fmin);
    deltaT = Tmax - Tmin;

    DC = 0.32f; T = (int)(0.92e+6f/f0);
    tON = (int) (DC*T); tOFF = (int)((1 - DC)*T);
    PwmFall = 0; falls = 6;
// Impostazione del Duty Cycle per la
// modulazione ed il numero degli impulsi

    srand (time (TIME_T(0)));
// inizializzazione del generatore dei
// numeri casuali con distribuzione uniforme
    bool doQuit = false;

```

```
// Implementazione del ciclo infinito
// per la generazione del PWM
for (dOut = false, fixed = true; !doQuit; )
{
    wait_us (tOFF); dOut = !dOut;
    wait_us (tON); dOut = !dOut;
}
}
```