

Firmware

it.emcelettronica.com

EMG superficiale per sport performance con Arduino MO Pro



- Progetto Power Line Monitor Open Source
- Progetto di una libreria per LCD 16x2 compatibile con Arduino MO PRO
- Implementare il Galvanic Skin Response a microcontrollore nei dispositivi indossabili
- Le tecnologie per l'energy harvesting si diffondono sempre di più
- Automatizzazione del processo di taratura di un motore a corrente continua
- STM32F4 Discovery: Tasti, PWM, ADC



Electronica Super Source

Solo da pochi mesi abbiamo acquisito la rivista [Firmware](#), ma con EOS-Book distribuiamo i PDF da ormai più di 3 anni. In questo periodo abbiamo raggiunto traguardi importanti, con migliaia di abbonati.

Ogni tanto però riceviamo qualche lamentela sull'usabilità del blog. Una percentuale bassissima rispetto al totale, zero virgola zero zero per cento.

“ MA giustamente, e ripeto, giustamente, abbiamo il dovere di aiutare anche l'utente smarrito nel nostro sito.

Del resto abbiamo più di 6500 articoli pubblicati nei quali è facile perdersi. Le richieste che abbiamo ricevuto sono state tutte prontamente risolte via email, ma ho comunque deciso di fare un articolo dedicato all'usabilità per una migliore fruizione dei contenuti.

Tramite il menu nella barra in alto è possibile accedere alla sezione [EOS-Book](#).

Sezione dedicata agli ABBONATI con l'elenco di tutti gli articoli PREMIUM che comprende anche gli EOS-Book pubblicati con i link ai download dei PDF. Nella stessa lista sono presenti anche gli articoli PLATINUM (archivio Premium).

Di seguito i link diretti agli articoli per i download dei PDF ePub e mobi: [EOS-Book](#) (per gli abbonati Premium mensili o annuali) [EOS-Book archivio](#) e [Firmware](#) (per gli abbonati Platinum).

Di seguito i link diretti agli articoli di [Firmware](#) ed agli articoli [Platinum](#).

E' inoltre disponibile anche un sito dedicato per una ulteriore fruibilità dei contenuti:

www.emcelettronica.com.

Emanuele Bonanni



SETTEMBRE 2015

Firmware

ANALOG

- 4 • Automatizzazione del processo di taratura di un motore a corrente continua**
- 33 • Le tecnologie per l'energy harvesting si diffondono sempre di più**

DESIGN

- 17 • Implementare il GSR (Galvanic Skin Response) a microcontrollore nei dispositivi indossabili**
- 24 • Progetto di una libreria per LCD 16×2 compatibile con Arduino MO PRO**
- 37 • EMG superficiale per sport performance con Arduino MO Pro**
- 47 • Power Line Monitor Open Source [Progetto Completo]**

TOOLS

- 56 • STM32F4 Discovery: Tasti, PWM, ADC**



Automatizzazione del processo di taratura di un motore a corrente continua



di **Giuseppe Silano**
03 Novembre 2015

L'articolo evidenzia l'importanza del processo di taratura, fondamentale affinché più misure di uno stesso oggetto siano confrontabili tra loro. Tale importanza passa attraverso l'automatizzazione del processo di taratura di un motore a corrente continua. L'obiettivo è realizzare un tachimetro digitale per misurare la velocità di rotazione di un motore, utilizzando lo stesso come dinamo tachimetrica, collegato attraverso un giunto metallico ad un secondo motore controllato utilizzando un full bridge, e la tecnica PWM. Allo scopo vengono descritti i componenti utilizzati, le fasi di lavoro, e i due pannelli frontali sviluppati utilizzando il software LabVIEW. Il primo consente di interfacciarsi alla struttura di controllo, settare i parametri, monitorare la tensione ai capi del motore, ed accedere ai diagrammi di taratura; il secondo permette ad un qualsiasi utente di variare il valore del duty cycle e conoscere la velocità di rotazione corrispondente del motore.

DESCRIZIONE DELLA STRUTTURA

In Figura 1.0.1 è mostrato lo schema a blocchi rappresentativo della struttura in esame dal quale si evince: l'alimentatore stabilizzato, il segnale di PWM, il circuito di controllo, il blocco motore, e l'encoder.

Il circuito di controllo (Figura 1.0.2) è alimentato mediante il generatore di tensione, Agilent E3631A, settato ad una tensione di 11 V e limitato in corrente ad 1 A.

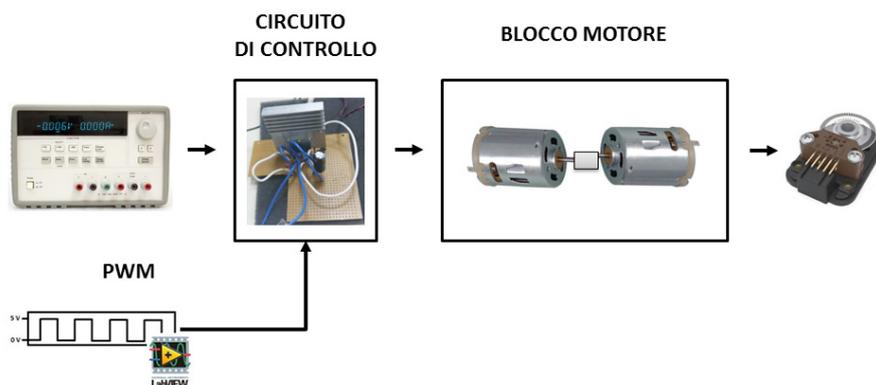


Figura 1.0.1: Schema a blocchi della struttura

L'elemento fondamentale di tale circuito è il convertitore **DC/DC Full Bridge LMD 18200**, il quale, come si deduce dallo schema circuitale (Figura 1.0.2), riceve in ingresso sul pin 5 il segnale di PWM, uscita digitale della DAQ NI PCI - 6036E (Figura 1.0.5), generato appositamente in **LabVIEW**. I pin 6 e 7 sono collegati rispettivamente all'alimentazione, fornita dal generatore, ed alla massa. Mentre, al fine di ridurre il ripple sull'alimentazione fornita in ingresso al full bridge, si è utilizzato un apposito condensatore elettrolitico del valore di $220 \mu\text{F}$. Infine, i pin 1 e 11 destinati al **bootstrap** sono connessi mediante due condensatori ceramici di 10 nF ai pin 2 e 10, collegati a loro volta ai capi del motore.

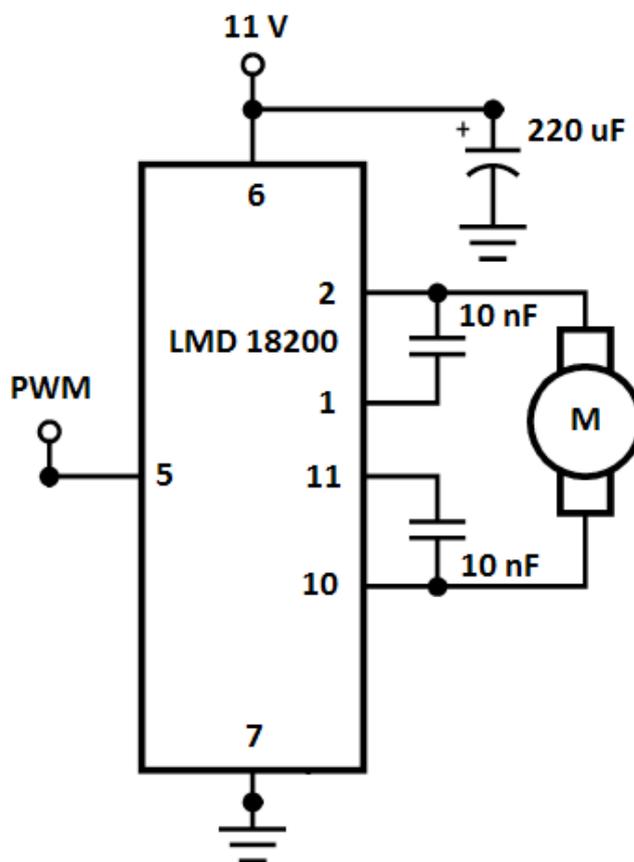


Figura 1.0.2: Convertitore DC/DC Full Bridge, LMD18200

La parte principale dello schema è rappresentata dal "blocco motore" (Figura 1.0.1), costituito da un primo motore comandato attraverso il circuito di controllo e collegato mediante un giunto metallico ad un secondo motore (Figura 1.0.3), che funge da dinamo tachimetrica.



Figura 1.0.3: Blocco motore: Gebr Bühler Nachf. Motor

Osservata la tensione d'uscita ai capi del secondo motore, si vuole misurare la sua velocità di rotazione in giri/minuto (rpm, rounds per minute). A tal fine, si dispone di un encoder ottico incrementale Hp HEDS 9500 (Figura 1.0.4), ossia, un trasduttore analogico - digitale che, sulla base della rotazione dell'asse del motore, fornisce in uscita due segnali elettrici in quadratura sottoforma di treni d'impulsi rettangolari.

I due segnali sono disponibili in uscita ai pin channel A e channel B (Figura 1.0.4). Il channel B è la sorgente del contatore ed è connesso al pin DEV 1CTR1SRC del blocco connettore NI CB - 68LP, a sua volta collegato tramite l'adattatore PCI 1200 alla DAQ; invece, il channel A fornisce la direzione del conteggio ed è collegato al pin P.07. Inoltre, ai pin 8 e 35, sempre della morsettiera, sono collegati i due ingressi dell'encoder relativi all'alimentazione (VCC) ed alla massa (GND). Infine, il terminale N.C. (not connected) dell'encoder è collegato al rispettivo pin N.C. del blocco connettore.

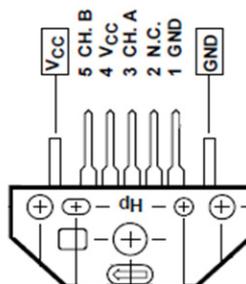


Figura 1.0.4: HP HEDS 9500, package pinouts

Il funzionamento di tutta la struttura è possibile grazie alla scheda DAQ di acquisizione dati, NI PCI - 6036E (Figura 1.0.5) che funge da interfaccia tra il computer e i segnali del mondo esterno. Essa include sedici ingressi analogici a 16 bit e due uscite analogiche a 16 bit, presenta poi otto linee I/O digitali e due counter/timer a 24 bit, a 20MHz.



Figura 1.0.5: NI PCI - 6036E

In Figura 1.0.6 è mostrata la struttura nel suo insieme.

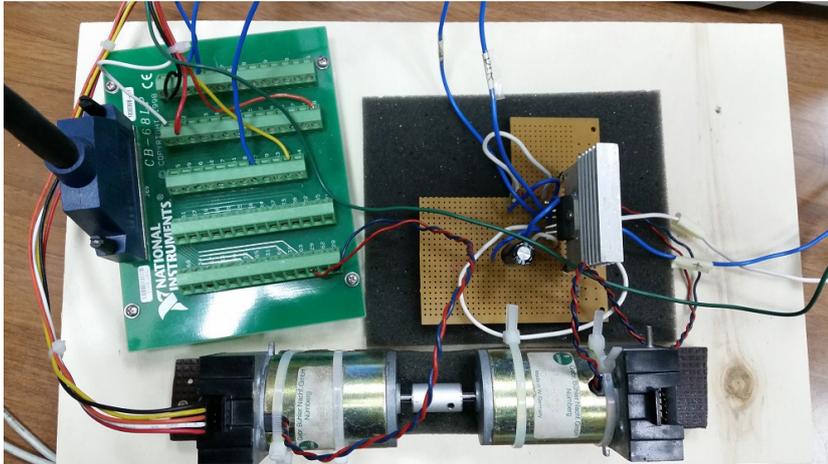


Figura 1.0.6: Struttura completa

TARATURA AUTOMATICA IN LABVIEW

L'obiettivo di questo progetto, ripercorrendo quanto già detto all'interno del paragrafo introduttivo, è quello di realizzare un tachimetro in modo che l'utente finale, inserendo un valore di duty cycle, possa conoscere la corrispettiva velocità in giri al minuto del motore. **La relazione costitutiva che lega la tensione ai capi del motore, funzione del duty cycle, alla sua velocità di rotazione è stata ottenuta attraverso un processo di taratura automatico.** Il processo di taratura, così come il tachimetro, sono stati realizzati nell'ambiente di sviluppo LabVIEW. **LabVIEW** è uno strumento di programmazione grafico anziché testuale, sviluppato dalla *National Instruments (NI)*. Tale linguaggio, basato sul modello di programmazione a flusso di dati, è anche detto "Linguaggio G". Un programma o un sottoprogramma scritto in questo ambiente viene detto VI (*Virtual Instruments*) ed è composto dai seguenti elementi:

- **il Pannello Frontale**, con il quale si interfaccia l'utilizzatore, che contiene gli input,

detti *controlli* e gli output, chiamati *indicatori*;

- **un Block Diagram**, ossia il codice dell'applicazione, costituito da nodi, terminali e connessioni;
- **un'icona di connessione** per poter richiamare il VI da un altro VI (in questo caso prende il nome di SubVI).

LabVIEW trova impiego principalmente per l'acquisizione e l'analisi dei dati, il controllo di processi, la generazione di rapporti, o più in generale, per tutto ciò che riguarda l'automazione industriale. Ai fini del progetto, sono stati realizzati due diversi VI, uno per il processo di taratura automatico, l'altro per la realizzazione del tachimetro.

In questo paragrafo, verrà analizzato esclusivamente il VI relativo al primo. Vedremo come l'operatore selezionando diversi parametri, quali valore iniziale e passo del duty cycle, frequenza della generazione della PWM e canali fisici, dai quali la DAQ Pannello Frontale della taratura preleva e fornisce segnali, potrà monitorare la tensione in uscita ai capi del motore e visualizzare le caratteristiche statiche tra cui diagrammi di taratura, isteresi, accuratezza, etc.

PANNELLO FRONTALE DELLA TARATURA

In Figura 2.1.1 è mostrato il pannello frontale relativo alla taratura. A sinistra è possibile notare il pulsante di *Start/Stop* del programma con il relativo LED, i parametri relativi alla PWM ed



all'encoder.

BLOCK DIAGRAM DELLA TARATURA

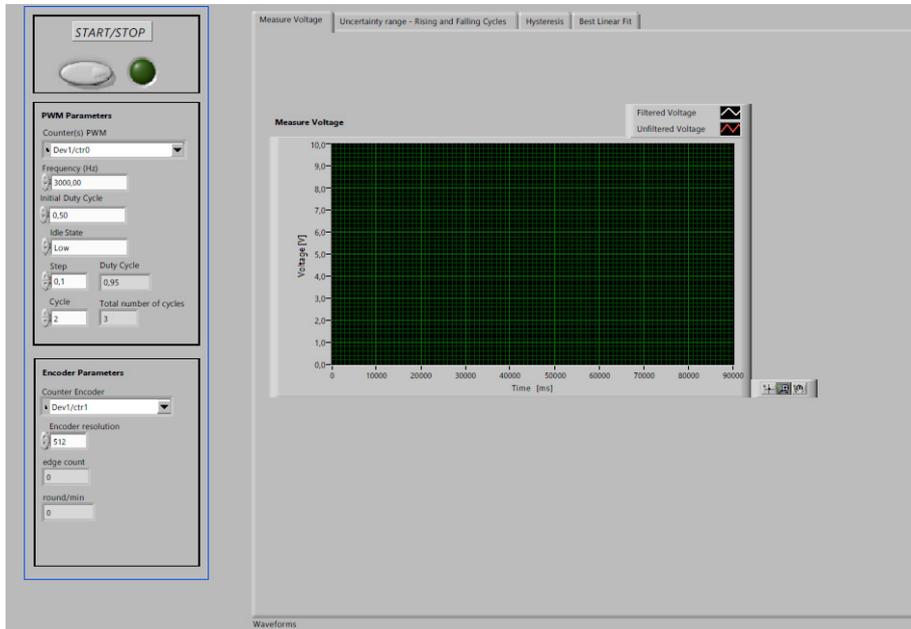


Figura 2.1.1: Pannello frontale della taratura

Nella Figura 2.1.1 sono mostrati i campi della PWM che l'operatore può settare: il canale, la frequenza, l'idle state (cioè lo stato a riposo del segnale, alto o basso), il valore iniziale ed il passo del duty cycle. E' possibile, inoltre, visualizzare in tempo reale, tramite appositi indicatori, il valore del duty cycle ed il numero totale di cicli corrente.

Nella stessa figura sono mostrati i controlli dell'encoder da settare, il canale e la risoluzione, gli indicatori corrispondenti al numero di fronti del segnale di counter (*edge count*), ed il numero di giri al minuto relativo al valore del duty cycle corrente.

Infine, è presente la *TabControl* che permette all'operatore di selezionare la pagina desiderata tra *Measure Voltage*, *Uncertainty range - Rising and Falling Cycles*, *Hysteresis* e *Best Linear Fit*.

Il Block diagram del VI taratura è stato strutturato in due parti. Nella prima si effettua l'acquisizione dei dati necessari al processo di taratura, mentre, nella seconda si effettua l'analisi di regressione sulla base delle informazioni acquisite, in modo da ottenere i corrispondenti diagrammi di taratura.

L'acquisizione dei dati viene gestita attraverso un ciclo *for*, in modo che vengano realiz-

zati N cicli di misura in salita ed N cicli di misura in discesa, dove N è inserito dall'operatore attraverso il pannello frontale. Ciascun ciclo di misura prevede M valori di duty cycle, dove M è dato da:

$$M = \left\lceil \frac{duty_cycle_{max} - duty_cycle_{min}}{passo} \right\rceil + 1$$

Per ogni valore corrente del duty cycle, posto che non si sia verificata una condizione di terminazione (stop del programma o raggiungimento dei valori limiti di duty cycle), si entra nel *true* di una struttura *if - else*, dove il processo di acquisizione delle informazioni è gestito attraverso una *flat sequence* (Figura 2.2.1).

Nel primo frame della *flat sequence* viene settato il verso della direzione del motore, vengono configurati la PWM, l'encoder e la lettura della tensione in uscita ai capi del motore. Il secondo



frame riguarda, invece, la gestione del transitorio. Il terzo frame è il nucleo della *flat sequence*, in quanto è qui che avviene l'acquisizione dei campioni relativi al segnale di tensione opportunamente filtrato, e la memorizzazione sia del loro valore medio che del valore di velocità di rotazione del motore in giri al minuto, per il valore corrente di duty cycle. Nell'ultimo frame viene gestito il *clear dei task* creati nel primo, inoltre, viene visualizzata la forma d'onda della tensione.

Al termine dei $2N$ cicli di misura, in uscita dal ciclo for, sono disponibili due matrici di dimensione $2N \times M$, una per la velocità di rotazione (*giri al minuto*) e l'altra per la tensione (V). In particolare, le prime N righe sono relative ai cicli di misura in salita, mentre, le restanti N righe sono relative a quelle in discesa. Inoltre, ciascuna colonna corrisponde ad un determinato valore di duty cycle tra gli M possibili.

Nella Figura 2.2.2 è mostrato il Block Diagram relativo all'elaborazione dei dati, come si può vedere vengono calcolati i valori della deviazione standard delle misure in salita, in discesa, e quelle totali. Successivamente vengono utilizzati appositi blocchetti per ricavarne le funzioni di regressione. All'interno della figura volutamente non sono state rappresentate le linee di interconnessione, con il fine di rendere più agevole la lettura dell'intero schema.

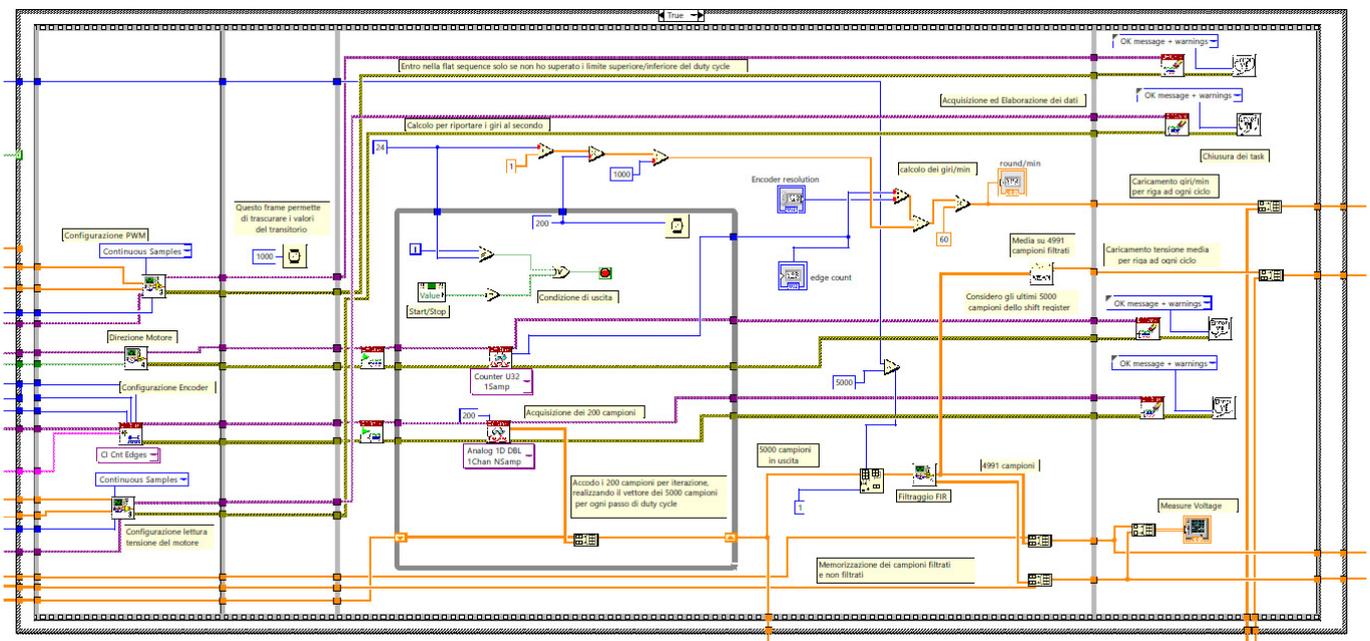


Figura 2.2.1: Struttura if - else con all'interno la flat sequence

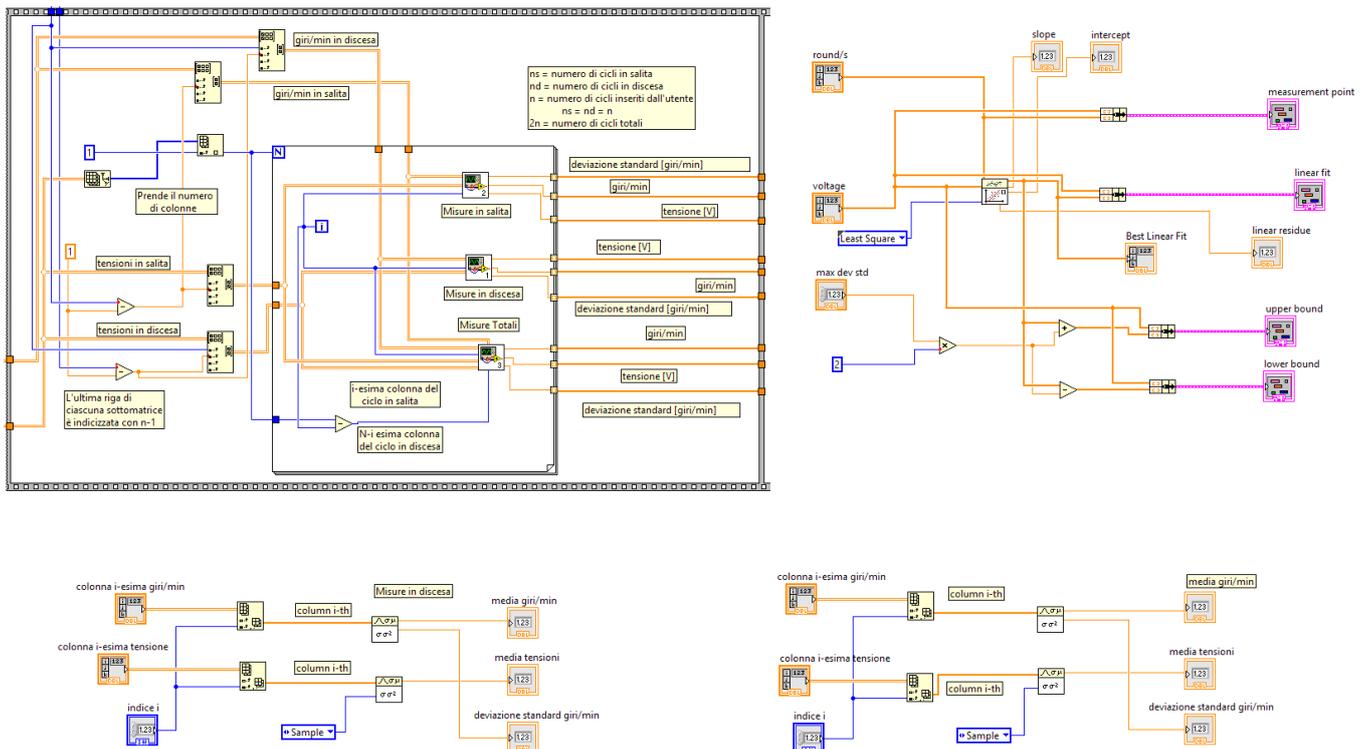


Figura 2.2.2: Seconda parte Block Diagram, elaborazione dati.

ANALISI DEI DATI

In questo paragrafo, verranno mostrati i risultati delle prove sperimentali effettuate, verificando il corretto processo di taratura e traendo le relative conclusioni sull'analisi di regressione realizzata. I parametri utilizzati sono riassunti nella tabella seguente (Tabella 2.1):

Simbolo	Descrizione	Valore
N	numero di cicli di misurazione in salita o in discesa	2
$2N$	numero totale dei cicli di misurazione	4
$duty\ cycle - iniziale$	valore minimo del duty cycle	0.15
$passo - duty\ cycle$	passo del duty cycle	0.05
M	numero di duty cycle applicati	17
$frequenza\ PWM [Hz]$	frequenza di generazione della PWM	3000

Tabella 2.1: Tabella dei parametri

Ai fini della taratura, è stato necessario determinare il valore del passo minimo in modo da garantire la variabilità della misura a fronte

di un incremento del valore del duty cycle. In particolare, avendo fissato un determinato valore di duty cycle, sono state raccolte K misure sia di giri al minuto che di tensione. Sulla base di queste popolazioni, calcolate medie e deviazioni standard, si è scelto di utilizzare come valore di passo minimo quello che assicurava la non sovrapposizione degli intervalli di misura, ossia 0.03.

Il processo di taratura dello strumento è stato svolto in condizioni ambientali il più possibile controllate per limitare la variazione dei dati. Allo

scopo di conoscere la velocità di rotazione del motore, si è variato il duty cycle su tutto il possibile range di applicazione, dal valore minimo



di 0.15 al valore massimo di 0.95, secondo il passo selezionato.

Come si evince dal grafico (Figura 2.3.1) dell'andamento della tensione nel tempo, vi sono, per un singolo ciclo di misurazione, 17 diverse acquisizioni del segnale di tensione in corrispondenza degli altrettanti valori di duty cycle applicati con un passo di 0.05. La tensione, inoltre, è stata opportunamente filtrata utilizzando un filtro FIR a media mobile in modo da eliminare le fastidiose fluttuazioni.

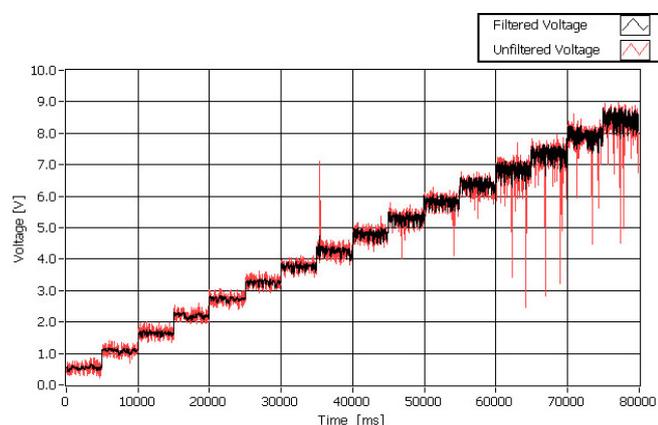


Figura 2.3.1: Caratteristica tempo - tensione

Fatta l'ipotesi di gaussianità, considerando dapprima solo i cicli in salita, poi solo quelli in discesa ed infine quelli totali, è stata effettuata l'operazione di media aritmetica sulle misure ottenute di giri al minuto e di tensione.

La curva che approssima in modo adeguato i valori medi delle misure effettuate, in corrispondenza di ogni valore di duty cycle applicato, è la *curva di taratura*. Tra le tipologie di regressione sono state valutate quella di tipo lineare,

quadratica e polinomiale. Si è scelta come funzione di interpolazione quella lineare, sulla base dell'errore quadratico medio, o residuo, e del coefficiente di correlazione lineare R^2 , tenendo anche conto della complessità computazionale (Tabella 2.2).

Tipologia di regressione	Residuo
Lineare	1.942
Quadratica	1.816
Polinomiale	1.866

Tabella 2.2: Tabella dei residui

Di seguito sono mostrati i grafici relativi all'analisi di regressione lineare effettuata, per i cicli in salita, in discesa e quelli totali. In Figura 2.3.2 è mostrata la retta di regressione relativa ai cicli di misura in salita. Ciascun punto di misura ha come coordinate il valore medio di tensione e quelli di giri al minuto relativi allo stesso valore di duty cycle. Tutti i punti di misura ricadono all'interno delle fasce d'incertezza $\pm 2\sigma$.

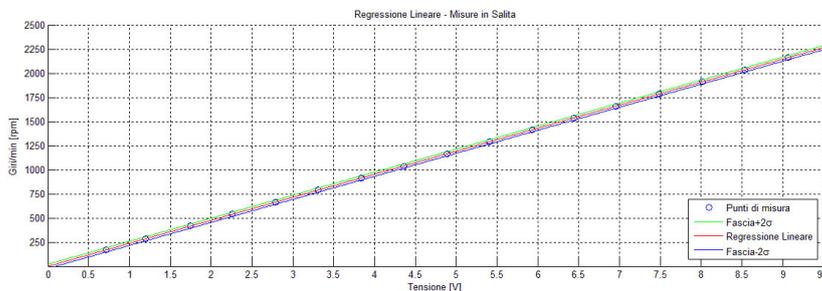


Figura 2.3.2: Regressione lineare dei cicli in salita

In Figura 2.3.3, analogamente a quella precedente, è mostrata la retta di regressione relativa ai cicli di misura in discesa, con le relative fasce d'incertezza.

ANALOG

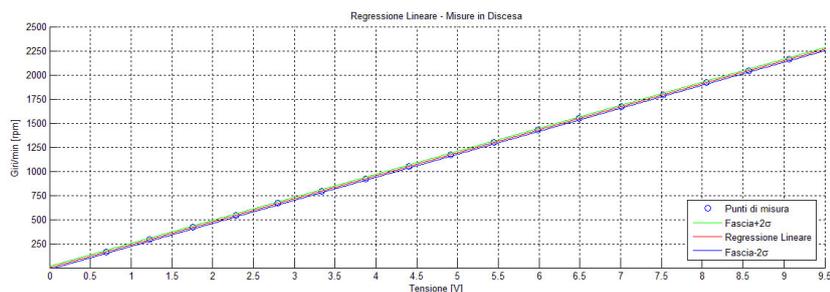


Figura 2.3.3: Regressione lineare dei cicli in discesa

In Figura 2.3.4 sono mostrate, per ogni valore di duty cycle applicato, le differenze in valore assoluto tra i corrispondenti valori di giri al minuto in salita e in discesa per il relativo valore di tensione. Il massimo valore tra queste differenze rappresenta l'isteresi, ed è pari a 15.293 giri al minuto.

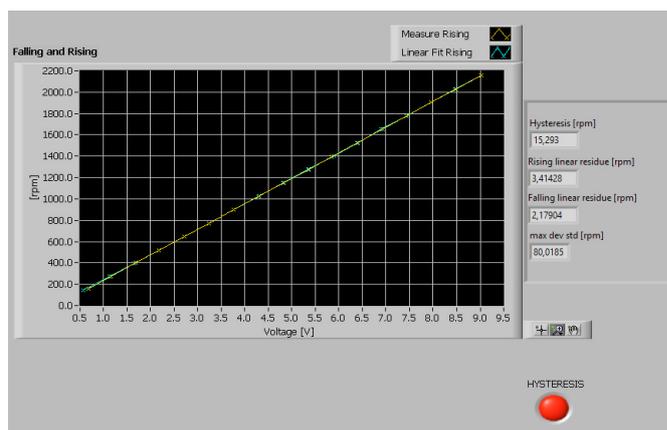


Figura 2.3.5: Tab Control, Hysteresis

Infine, in Figura 2.3.6 è mostrata la retta di regressione relativa a tutti i cicli di misura effettuati, sia in salita che in discesa. Ancora una volta i punti di misura ricadono all'interno delle fasce d'incertezza.

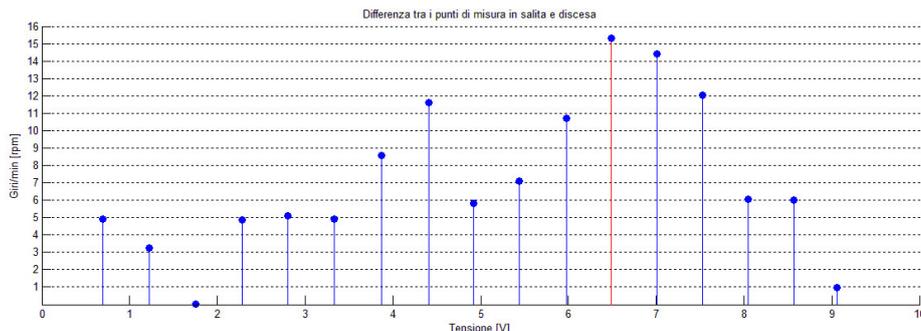


Figura 2.3.4: Isteresi: cicli in salita ed in discesa

Come si evince dalla Figura 2.3.5, il LED relativo all'isteresi è spento (rosso), infatti, il valore dell'isteresi calcolato è inferiore al valore massimo della deviazione standard (rpm) che invece è pari a 80.018.

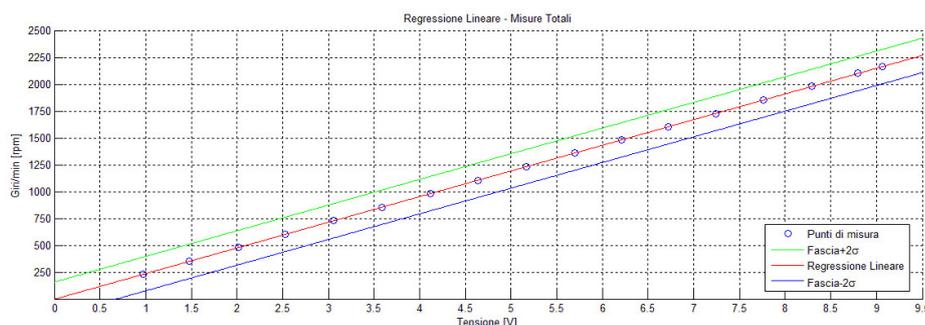


Figura 2.3.6: Regressione lineare sul numero totale di cicli



Il grafico in Figura 2.3.7 conferma la scelta di utilizzare una curva di regressione lineare anziché quadratica, in quanto i loro andamenti sono simili.

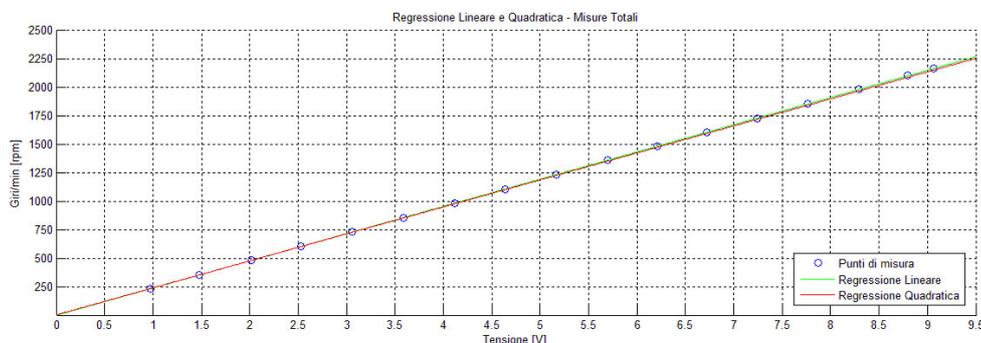


Figura 2.3.7: Confronto tra curve di regressione sul numero totale di cicli

Al fine di testare la **precisione** delle misure effettuate, il processo di taratura è stato ripetuto più volte *in condizioni di ripetibilità*, quali stessa temperatura di funzionamento (26 °C), stesso piano di appoggio della strumentazione, assenza di urti accidentali, ecc. La Tabella 2.3 mostra come i parametri di ciascuna retta di regressione ottenuti in queste diverse prove sperimentali sono simili tra loro.

Slope	Intercept
239.058	-1.448
238.676	-1.603
239.010	-2.776
239.153	-1.857
238.801	-2.303

Media	238.940	-1.997
Dev. Std.	0.175	0.485

Tabella 2.3: Tabella: misure in condizioni di ripetibilità

IL TACHIMETRO DIGITALE

Terminato il processo di taratura, si può passare alla fase finale del progetto: **la realizzazione del tachimetro**. Analizzeremo, ora, dapprima il

Block diagram ed in seguito il Pannello frontale del VI, attraverso il quale un qualunque operatore, settando il solo valore del duty cycle, è in grado di conoscere la velocità di rotazione in

giri al minuto del motore.

BLOCK DIAGRAM DEL TACHIMETRO

La parte del Block Diagram del tachimetro relativa alla configurazione della PWM, alla misura della tensione ed alla direzione del motore, è analoga a quella della taratura; in aggiunta, vi è la lettura da file dei parametri (i risultati del processo di taratura sono stati salvati in un file .txt), il coefficiente angolare e l'intercetta, della retta di regressione lineare ottenuta nel VI taratura considerando tutti i cicli di misura, sia in salita che in discesa.

L'operazione di lettura è stata realizzata attraverso il blocchetto *Read from Spreadsheet* Figura 3.1.1

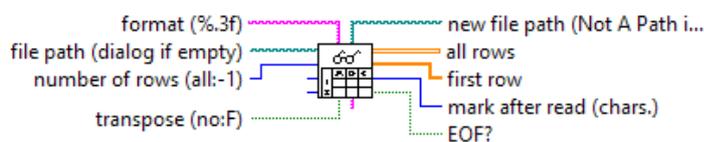


Figura 3.1.1: Lettura da file

Il funzionamento effettivo del tachimetro con-



siste in un ciclo *while* (Figura 3.1.2), nel quale, avviene la lettura del valore del duty cycle e l'acquisizione di 500 campioni del segnale di tensione ai capi del motore.

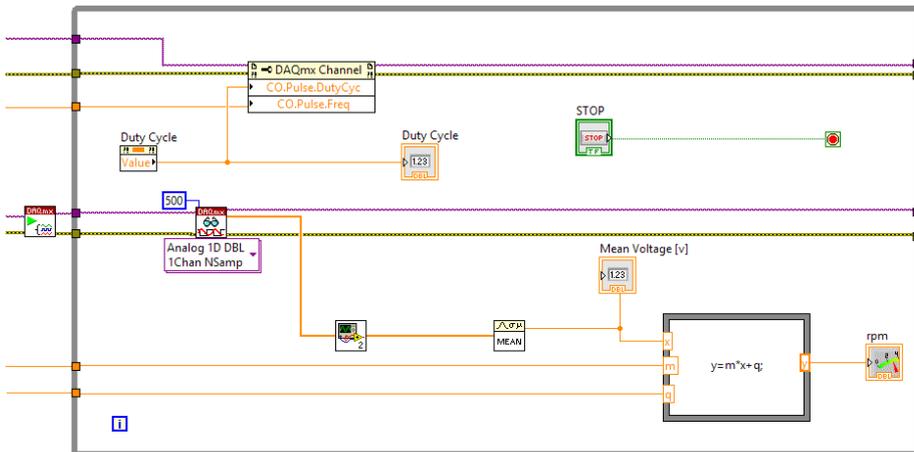


Figura 3.1.2: Ciclo while del tachimetro

L'utilizzo del blocchetto *DAQmx Channel property node*, avente la proprietà "change to write", consente di osservare istantaneamente gli effetti della variazione del valore del duty cycle senza la necessità di effettuare la chiusura del task corrente per poter generare la nuova PWM. I campioni di tensione letti vengono poi opportunamente filtrati, attraverso il filtro FIR a media mobile (lo stesso utilizzato nel processo di taratura). Sui campioni filtrati, eliminati quelli sporchi, viene effettuata l'operazione di media aritmetica. Il valore di tensione x così ottenuto, insieme a quelli di coefficiente angolare m ed intercetta q , letti precedentemente (Figura 3.1.1), entrano nel blocco *Formula-Node* in cui è implementata l'equazione della retta $y = mx + q$. In uscita dal blocco si rende disponibile all'utente il valore della velocità di ro-

tazione del motore (rpm), corrispondente al duty cycle corrente.

La condizione di uscita dal ciclo è dettata dall'azionamento del pulsante di *STOP* da parte dell'operatore. Infine, vi è la parte del Block Diagram relativa alla chiusura dei task.

PANNELLO FRONTALE DEL TACHIMETRO

Nella Figura 3.2.1 è mostrato il Pannello frontale del tachimetro destinato all'utente finale. A sinistra vi è il *knob*, con il quale l'utente può se-

lezionare il valore del duty cycle in un range compreso tra 0.15 e 0.95, con un passo di 0.05, e leggerlo nell'apposito indicatore. Vi è poi il pulsante di *STOP*. A destra vi è l'effettivo tachimetro che mostra all'utente, attraverso il posizionamento di una lancetta, il valore della velocità di rotazione in giri al minuto per quel valore di duty cycle selezionato. Allo stesso tempo l'utente può conoscere anche il valore medio della tensione, ai capi del motore, per quella velocità.

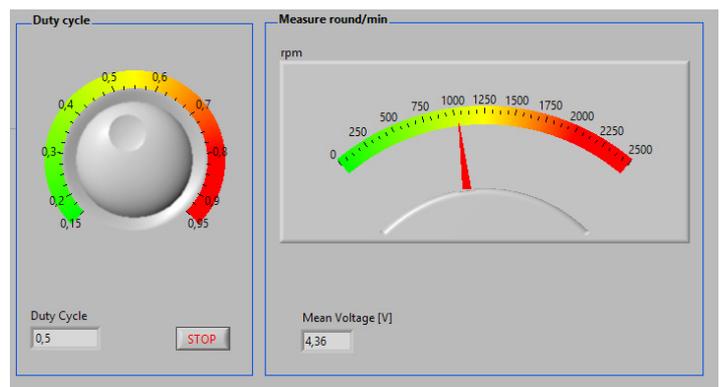


Figura 3.2.1: Pannello frontale del Tachimetro



CONCLUSIONI

L'articolo ha consentito di illustrare, passando attraverso la realizzazione di un progetto, quanto sia importante il processo di taratura, e della sua automatizzazione (l'operatore potrebbe introdurre errori di misura), le accortezze da tenere a mente durante la sua realizzazione, e quanto sia fondamentale la conoscenza di **misure elettroniche**. Infatti, un'errata valutazione dei dati a disposizione può comportare un'errato processo di taratura, conseguentemente valori di giri/minuto (nel caso in esame) che possono provare danni all'interno di un processo industriale, cosa non da poco per chi vuole fare di questa passione una professione.

NOTE

Bootstrap: Quando il carico è collegato tra source e massa (high side), applicando il segnale di pilotaggio, il mosfet non entra in conduzione, in quanto, il source non è a massa, bensì segue la tensione sul carico. Per assicurare la conduzione di tale mosfet si rende, pertanto, necessario un apposito circuito di bootstrap che innalza la tensione sulla gate del transistor ad un valore superiore a quello di alimentazione.

Taratura: Definizione di taratura dal **VIM** (Vocabolario Internazionale di Metrologia): operazione eseguita in condizioni specificate, che in una

prima fase stabilisce una relazione tra i valori di una grandezza, con le rispettive incertezze di misura, forniti da campioni di misura, e le corrispondenti indicazioni, comprensive delle incertezze di misura associate, e in una seconda fase usa queste informazioni per stabilire una relazione che consente di ottenere un risultato di misura a partire da un'indicazione. L'esito di una taratura può essere espresso mediante una dichiarazione, una funzione di taratura, un diagramma di taratura, una curva di taratura, o una tabella di taratura. In alcuni casi esso può consistere in un semplice fattore additivo o moltiplicativo, utilizzabile per la correzione, accompagnato dall'incertezza di misura associata.

Precisione: Definizione di Precisione di misura dal **VIM**: grado di concordanza tra indicazioni o valori misurati ottenuti da un certo numero di misurazioni ripetute dello stesso oggetto o di oggetti simili, eseguite in condizioni specificate.

ALLEGATI

[Pannello_Utente_Project with flat sequence e Project with flat sequence](#)

L'autore è a disposizione nei commenti per eventuali approfondimenti sul tema dell'Articolo.
Di seguito il link per accedere direttamente all'articolo sul Blog e partecipare alla discussione:
<http://it.emcelettronica.com/automatizzazione-del-processo-di-taratura-di-un-motore-a-corrente-continua>

Quello che hai appena letto è solo un estratto, Firmware n. 118 è composto da ben 68 pagine ed è riservato agli abbonati PLATINUM!

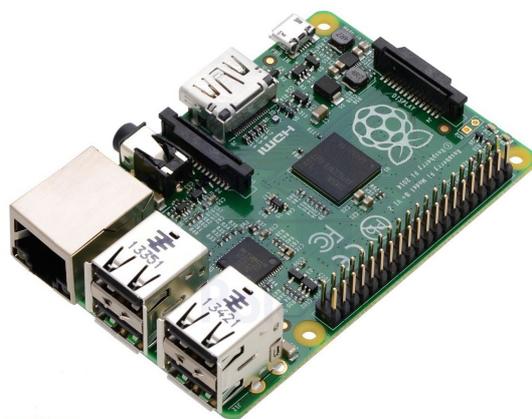


Sottoscrivi subito l'abbonamento PLATINUM, accedi agli articoli Firmware pubblicati in anteprima sul nostro blog ed effettua il download della rivista dedicata all'Elettronica Embedded ed ai Microcontrollori (in PDF, ePub e mobi) per un anno!

Inoltre, con l'abbonamento Platinum potrai accedere e scaricare un anno di EOS-Book, la rivista dedicata alla Scienza e alle Tecnologie, l'intero Archivio storico e gli Speciali monotematici (ben 43 numeri in PDF, ePub e mobi!!!)



E se ti abboni entro il 31 dicembre ricevi in omaggio (spese di spedizione a carico nostro) la nuova Raspberry Pi B+ , il mini-pc per realizzare tutti i tuoi progetti!!!



Che aspetti? ABBONATI ORA con PAYPAL è semplice e sicuro

SCOPRI DI PIU'



Implementare il GSR (Galvanic Skin Response) a microcontrollore nei dispositivi indossabili



di **Maurizio**
05 Novembre 2015

La risposta galvanica della pelle (GSR) è definita come un cambiamento delle proprietà elettriche in seguito ad eventi quali lo stress o sudorazione da esercizi fisici. Le misurazioni GSR rilevano l'impedenza della pelle umana e la temperatura corporea in situazioni diverse e possono essere utilizzate nel trattamento medico, benessere e nella rivelazione del falso (macchina della verità). GSR è anche conosciuto come EDA (Electrodermal Activity) che è la proprietà del corpo umano che provoca una variazione continua delle caratteristiche elettriche della pelle. Un sistema di questo tipo può essere implementato con un sistema mobile, al fine di provvedere al monitoring in real time delle condizioni di salute attraverso l'interfaccia bluetooth a basso consumo energetico (BLE). In questo articolo analizzeremo un design con il microcontrollore MAX32600 della Maxim evidenziando i suoi utilizzi in vari campi applicativi.

1. INTRODUZIONE

La risposta galvanica della pelle (GSR) è una variazione in termini di energia cinetica trasmessa dai nervi e dal sudore della pelle, definita come una delle **diverse risposte elettrodermiche (EDR)** relative a cambiamenti nelle proprietà elettriche della pelle di una persona, causati da un'interazione tra gli eventi ambien-

tali e dallo stato psicologico dell'individuo. La pelle umana è un buon conduttore di elettricità e quando una debole corrente elettrica viene erogata alla pelle, possono essere misurati cambiamenti nella conduzione. La variabile che si misura è la resistenza della pelle o il suo reciproco, la conduttanza cutanea. Secondo la legge di Ohm, la resistenza della pelle (R) è uguale



alla tensione (V) applicata tra due elettrodi sulla pelle divisa per la corrente che la attraversa (I). La legge può essere espressa come $R = V / I$. Un amplificatore GSR applica una tensione costante sulla pelle attraverso dei mini-elettrodi. La tensione è così piccola che non può essere percepita. La corrente che fluisce attraverso la pelle può essere rilevata e visualizzata per mezzo di eventuali display. Poiché la tensione costante applicata alla pelle è nota e il flusso di corrente può essere misurato, la conduttanza della pelle, espressa in microSiemens (μS), può essere determinata dalla legge di ohm. **La conduttanza è caratterizzata da due tipi: tonica e fasica.** La conduttanza cutanea tonica è il livello di base in assenza di qualsiasi particolare evento ambientale, ed è generalmente indicato come Skin Conductance Level (SCL). La conduttanza cutanea fasica, invece, cambia in funzione degli eventi. Eventuali stimoli discreti ambientali (suoni, odori, ecc) evocano cambiamenti connessi alla conduttanza cutanea. Questi sono generalmente indicati come risposte di conduttanza cutanea (SCR). Le variazioni di SCR possono durare circa 10-20 secondi e sono spesso denominate GSR.

La frequenza tipica di **GSR spontanei è compreso tra uno e tre per minuto.** Alcune persone sono molto reattive con notevole generazione spontanea di GSR, mentre altri hanno un livello tonico relativamente costante di conduttanza cutanea, senza segnali GSR spontanei. I parametri che possono essere definiti per la misura sono: l'ampiezza, in **microSiemens (μS)**, e la latenza. L'ampiezza di un evento GSR è la

differenza tra il livello di conduttanza della pelle al momento della risposta, e quella al picco della risposta stessa. La latenza è il tempo che intercorre tra lo stimolo e l'inizio del GSR; valori di latenza dovrebbe essere di circa tre secondi o meno. Un altro parametro è il tempo di salita, ovvero il tempo tra l'inizio dell'evento GSR e il picco della risposta; valori tipici per il tempo di salita sono da uno a tre secondi.

2. IL SISTEMA ELETTRODERMICO

La pelle è essenzialmente formata da tre strati (epidermide, derma, subdermis) che aiutano nella manutenzione del bilancio idrico e nella regolazione della temperatura del corpo. Queste funzioni sono svolte da vasi sanguigni e ghiandole sudoripare eccrine. **I vasi sanguigni si restringono e si dilatano per fornire il flusso di sangue sulla superficie del corpo per la conservazione o rilascio di calore.** Le ghiandole eccrine raffreddano il corpo depositando umidità sulla superficie della pelle attraverso dei condotti. Le ghiandole, che si trovano nelle subdermis su gran parte del corpo, sono circa dieci volte più dense sulla superficie palmare delle mani e delle piante dei piedi. Il modello più ampiamente accettato di conduttanza cutanea è quello proposto da Edelberg nel 1972. Nel modello di Edelberg, i condotti di sudore si comportano come resistenze variabili; la loro resistenza diminuisce (aumenta conduttanza) in quanto si riempiono di sudore. L'ampiezza della variazione della conduttanza dipende dalla quantità di sudore consegnata ai condotti e dal numero di ghiandole sudoripare che sono state attivate.



L'attivazione della ghiandola di sudore è un semplice meccanismo di sopravvivenza fisiologica che è di interesse per la psicologia perché è una risposta neurale. Questa attivazione è controllata dal cervello attraverso la divisione simpatica del sistema nervoso autonomo.

3. IL DESIGN

L'elemento principale è il **microcontrollore MAX32600** che integra le periferiche front-end analogiche per le misure di impedenza, in particolare un ADC a 16 bit con un MUX in ingresso e quattro DAC da 12 bit e 8 bit. La figura 1 visualizza un esempio di schema a blocchi per l'implementazione di un rivelatore GSR a basso costo, flessibile per lo sviluppo di dispositivi indossabili. Lo schema a blocchi di misura dell'impedenza è riportato in figura 2.

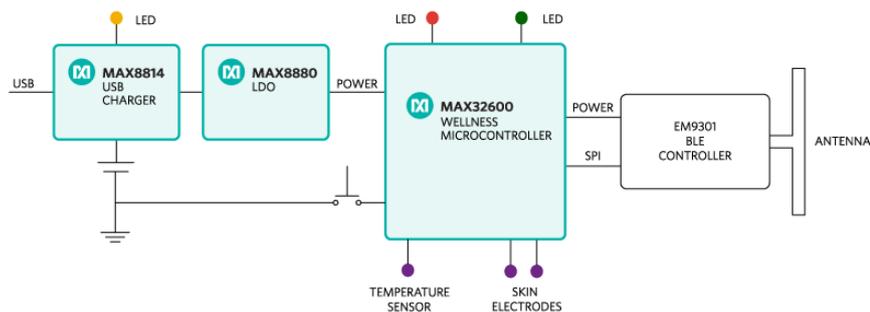


Figura 1: Schema a blocchi per l'implementazione di un rivelatore GSR

La zona di misura interessata (la pelle del polso) è l'input della impedenza di un amplificatore invertente opportunamente designato con gli operazionali disponibili nel microcontrollore. Un segnale sinusoidale stimola la risposta dell'impedenza ad una frequenza desiderata F_c che viene rilevata in termini di ampiezza. Esprimiamo la tensione di prova come:

$$x(t) = \cos(2\pi \times F_c \times t)$$

Il segnale in uscita dell'ADC può essere espresso mediante la seguente relazione:

$$y(t) = V_L \times \cos(2\pi F_c \times t + \theta)$$

scalata e shiftata di fase rispetto all'input. Per estrarre la fase del segnale ricevuto, è necessaria una **rivelazione coerente**.

I sistemi coerenti hanno bisogno delle informazioni di fase e usano filtri adattati per rilevare e decidere quali dati sono stati inviati. Considerando che:

$$T_S = 1/(4 \times F_c)$$

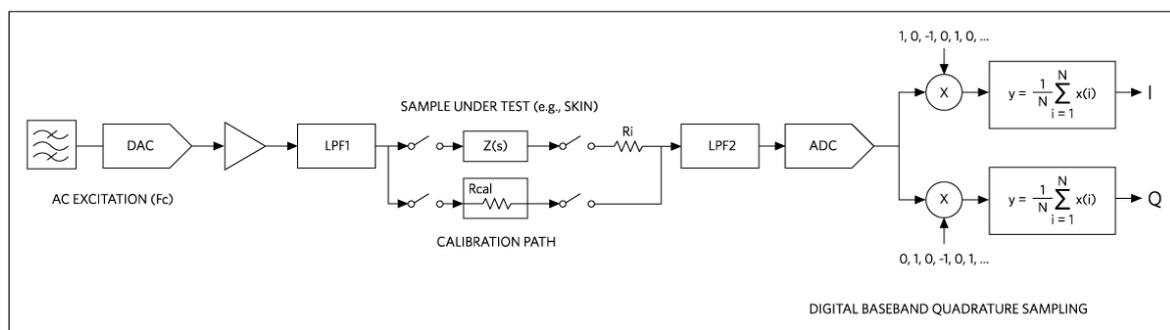


Figura 2: Misura dell'Impedenza



Allora:

$$y(k) = V_L \times \cos(\pi/2 \times k + \theta)$$

con $k = \{0, 1, 2, \dots, N - 1\}$ ed $N =$ campioni ADC

Utilizzando le formule della trigonometria possiamo esprimere la relazione sopra nel seguente modo:

$$y(k) = V_L [\cos\theta, \cos(\pi/2 + \theta), \cos(\pi + \theta), \cos(3\pi/2 + \theta), \dots]$$

$$y(k) = V_L [\cos\theta, -\sin\theta, -\cos\theta, \sin\theta, \dots]$$

Ricordando che:

$$V_L e^{j\theta} = V_L \cos(\theta) + jV_L \sin(\theta) = I + jQ$$

Ci calcoliamo i fattori I e Q analizzando i primi due campioni dell'ADC:

$$I = V_L \cos(\theta) = y(0)$$

$$Q = V_L \sin(\theta) = y(1)$$

Splittiamo i campioni (N) ADC in pari e dispari. Poichè stiamo elaborando una singola frequenza, possiamo mediare i valori per aumentare il rapporto segnale rumore. E quindi possiamo esprimere I e Q nel seguente modo andando ad osservare lo schema di figura 2:

$$I = \frac{2}{N} \sum_{i=0}^{\frac{N}{2}-1} y(2i)(-1)^i$$

$$Q = \frac{2}{N} \sum_{i=0}^{\frac{N}{2}-1} y(2i+1)(-1)^{i+1}$$

La fase e ampiezza possono essere espressi nelle seguenti equazioni:

$$\text{Fase} = \theta = \text{atan2}(Q, I)$$

$$\text{Ampiezza} = V_L = \sqrt{I^2 + Q^2}$$

Il carico di impedenza complessa ($Z(s)$) è misurata con un carico di calibrazione ($Y_{cal}(s)$) e con il campione di prova ($Y_{sys}(s)$) ad una frequenza di interesse. Sulla base del rapporto tra le risposte e le due resistenze esterne R_i e R_{cal} , è calcolata l'impedenza di prova:

$$(R_i + Z(s)) / R_{cal} = Y_{cal}(s) / Y_{sys}(s)$$

L'ampiezza e la fase impedenza di carico possono essere espresse nel seguente modo:

$$Z(s)_{\text{ampiezza}} = R_{cal} (Y_{cal}(s)_{\text{ampiezza}} / Y_{sys}(s)_{\text{ampiezza}}) - R_i$$

$$Z(s)_{\text{fase}} = Y_{cal}(s)_{\text{fase}} - Y_{sys}(s)_{\text{fase}}$$

Il progetto prevede una batteria ricaricabile agli ioni di litio con una tensione nominale di 3,6 V e capacità massima di 45mAH. La batteria è re-



golata dal LDO MAX8880 che offre una combinazione ottimale di bassa tensione di dropout, bassa corrente di riposo, rapida risposta ai transienti, basso rumore e buona reiezione ai ripple. In generale un **LDO è costituito da un riferimento di tensione, un amplificatore di errore e un partitore di tensione di retroazione**. La sua tensione di gate del transistor che fornisce la corrente in uscita è controllato dall'amplificatore che confronta la tensione di riferimento con la tensione di retroazione e amplifica la differenza in modo da ridurre la tensione di errore. Se la tensione di reazione è inferiore alla tensione di riferimento, il gate del transistor si comporta opportunamente per consentire una maggiore corrente con un aumento della tensione di uscita. Viceversa, se la tensione di retroazione è superiore alla tensione di riferimento. Inoltre, gli **LDO** richiedono condensatori di ingresso e uscita per filtrare transitori di carico di rumore e di controllo. **La batteria può essere comodamente caricata da una connessione USB di un PC attraverso il controller MAX8814** che integra un MOSFET ed elimina il diodo Schottky-reverse di blocco per una semplice soluzione offrendo una rapida carica con una corrente impostata a 32 mA. Un sensore di temperatura è utilizzato per valutare l'escursione termica della pelle, il microcontrollore dispone di un sensore interno e di pin per la disposizione di uno esterno. Nel design è utilizzato un termistore NTC con resistenza nominale di 30.15 kΩ a 37 °C. I Termistori sono elementi temperatura di rilevamento di materiale semiconduttore adatti a visualizzare grandi variazioni di resistenza in proporzione

alle piccole variazioni di temperatura. I parametri Steinhart-Hart del sensore sono i seguenti:

$$a = 824.970719, b = 222.454876, c = 0,095600.$$

Sulla base dell'approssimazione al terzo ordine dell'equazione Steinhart-Hart, la temperatura espressa in kelvin è data da

$$T = 1 / (a + b \times \ln (R_{NTC}) + c \times (\ln (R_{NTC}))^3)$$

Dove R_{NTC} è la resistenza del sensore misurata dall'ADC. Attraverso il controller Bluetooth Low Energy EM9301 della EM Microelectronics, il circuito è in grado di funzionare attraverso un dispositivo mobile (App. Android) per controllare e comandare le operazioni di lettura. **L'EM9301 è dotato di un Host Controller Interface (HCI) e una potente gestione dell'alimentazione**. Questa caratteristica è ottenuta grazie all'elevata efficienza di un convertitore up DC / DC che fornisce la tensione richiesta per il circuito. **Bluetooth Low Energy (BLE)**, a volte indicato come "Bluetooth intelligenti", è un sottoinsieme del classico Bluetooth, introdotto come parte della specifica Bluetooth 4.0. Mentre vi è una certa sovrapposizione con il classico Bluetooth, BLE ha in realtà una stirpe completamente diversa ed è stato avviato da Nokia come un progetto in-house chiamata 'Wibree' prima di essere adottato dalla Bluetooth SIG. Ciò che rende BLE interessante e applicabile in molti progetti è la semplice modalità di progettare sistemi che possono interloquiere con qualsiasi piattaforma



mobile moderna (telefoni iOS, Android, Windows, ecc.).

4. FIRMWARE

Il firmware del progetto è abbastanza semplice, basato essenzialmente su un modello interrupt-driven. Ogni volta che un trasferimento di dati da o verso l'hardware deve essere ritardato per qualsiasi motivo, è necessario attuare dei buffering. I Buffer di dati aiutano a staccare la trasmissione e ricezione dati dalla scrittura e lettura di sistema. Un buon meccanismo di buffer ci conduce verso il modello **interrupt-driven I/O**, in cui un buffer di ingresso viene riempito durante un interrupt e viene svuotato da processi che leggono il dispositivo; un buffer di uscita viene riempito da processi che scrivono al dispositivo e viene svuotato per mezzo di interrupt. In particolare, dopo aver definito le impostazioni quali alimentazione e clock, il dispositivo si connette con il primo mobile BLE che trova nella zona, passando nella modalità di sospensione qualora nessun dispositivo è presente (Figura 3). Appena la connessione è stata stabilita, il circuito misura l'impedenza ogni secondo e trasferisce i dati all'host (Figura 4). Il firmware è sviluppato utilizzando l'**ARM Cortex Toolchain per il pacchetto di microcontrollori medici Maxim Integrated**.

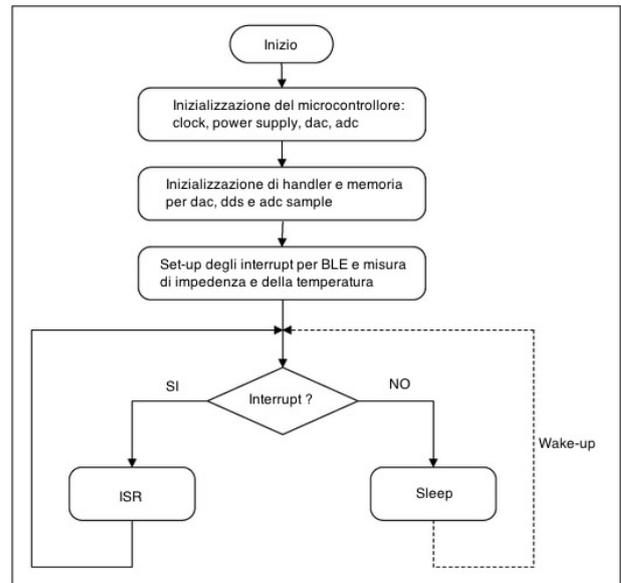


Figura 3: Diagramma a blocchi del firmware

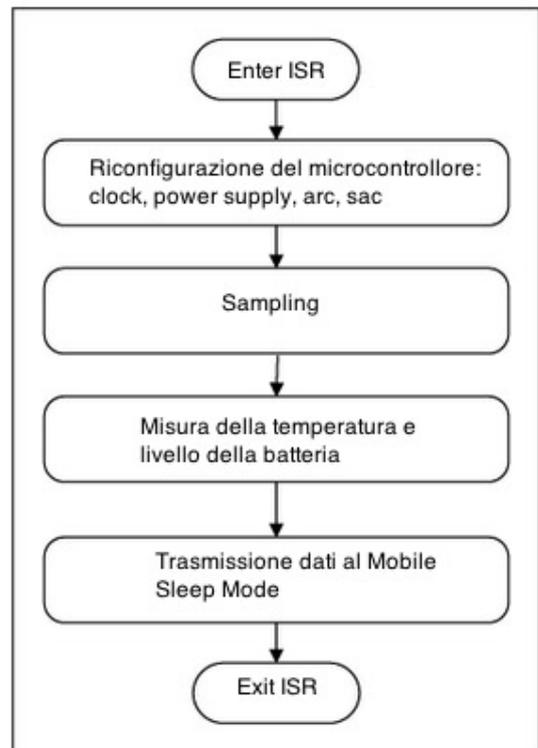


Figura 4: Diagramma a blocchi del Interrupt Service Routine (ISR)

5. CONCLUSIONI

Il progetto qui descritto (MAXREFDES73#) di-



Disponibile dalla **Maxim Integrated**, si rivolge a qualsiasi sviluppatore che abbia interesse a sviluppare dispositivi di misura di impedenza e temperatura con il microcontrollore MAX32600. EDA è ampiamente utilizzato nella ricerca psicologica a causa del suo basso costo e l'alta utilità. Molti dispositivi di terapia di biofeedback utilizzano **EDA come indicatori della risposta allo stress** con l'obiettivo di aiutare l'utente a controllare l'ansia. Spesso, il monitoraggio EDA è combinato con la registrazione della frequenza cardiaca, la frequenza respiratoria e la pressione sanguigna. Inoltre, la misurazione EDA è un componente dei moderni dispositivi utilizzati come macchine della verità (poligrafici). La conducibilità della nostra pelle cambia a seconda degli effetti di emozione inconsci come il rumore improvviso, lo stress, il tatto, dolore o vista. **I dispositivi GSR consentono di individuare emozioni forti dal semplice collegamento di sensori sulla nostra pelle.** Semplicemente misurando la conduttanza cutanea (che è controllata da una parte speciale del nostro cervello) possiamo individuare e rilevare le emozioni umane e lo stress...non vi pare molto interessante ?

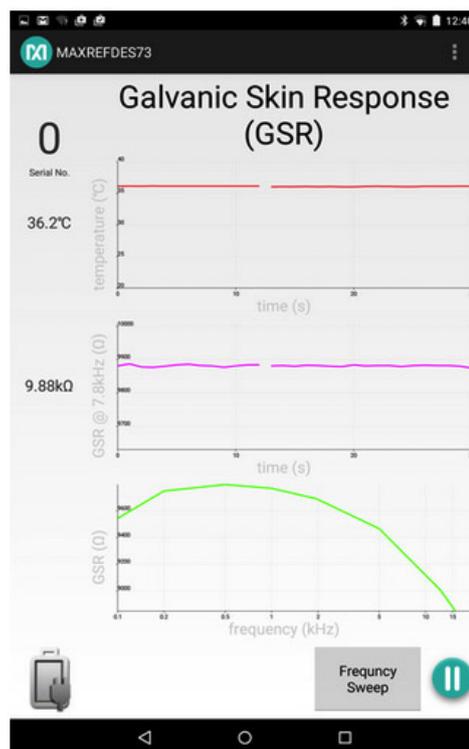


Figura 5: Screenshot della App android della Maxim per il progetto MAXREFDES73#

La misura della conducibilità della pelle è una sfida, perché i progettisti hanno bisogno di manipolare diversi chip discreti e software di calibrazione prima di completare il design del sensore. **Il MAXREFDES73# integra una serie di periferiche e una facile App da usare in Android (figura 5).** E' disponibile sotto forma di un braccialetto ed include misure di superficie della temperatura, le comunicazioni Bluetooth e una batteria ricaricabile che dura fino a una settimana con una singola carica.

L'autore è a disposizione nei commenti per eventuali approfondimenti sul tema dell'Articolo. Di seguito il link per accedere direttamente all'articolo sul Blog e partecipare alla discussione: <http://it.emcelettronica.com/implementare-il-gsr-galvanic-skin-response-a-microcontrollore-nei-dispositivi-indossabili>



Progetto di una libreria per LCD 16×2 compatibile con Arduino M0 PRO



di **Mario Mottula**
10 Novembre 2015

Questo articolo è stato scritto con lo scopo di provare a dare le nozioni fondamentali per la creazione di una libreria Arduino, in particolare analizzeremo una libreria da me creata e non ancora esistente che permette di gestire un display LCD 16x2 con controller Hitachi HD44780 compatibile. Le librerie vengono più volte usate in file sorgente, ed hanno lo scopo di semplificare di molto il listato di un programma rendendolo più snello e comprensibile, queste sono costituite da funzioni o strutture di dati ed hanno la caratteristica fondamentale di essere riutilizzabili evitando così al programmatore di riscrivere il codice ogni qualvolta si ha la necessità di eseguire quelle determinate operazioni.

L'idea di sviluppare la libreria è nata da una mia esigenza di progettazione, visto che quella messa a disposizione dall'IDE di Arduino non era ancora stata resa compatibile con il controller di **Arduino M0 Pro**, avendo la necessità di controllare un display di questo tipo ho deciso di crearla da zero.

Componenti utilizzati

- **Board Arduino M0 Pro**
- **LCD Data Vision DV-16275** compatibile con i segnali di dato a 3.3 Volt provenienti dalla M0 Pro (da notare che il display va comunque alimentato a 5 Volt, ma riconosce perfettamente i segnali d'ingresso a 3.3 Volt come

visibile a pagina 21 del datasheet scaricabile dal link seguente [Datasheet dv-12765](#).

- **Resistenza 220 Ohm 1/4 Watt** per impostazione contrasto standard LCD tra pin 3 e GND

PREPARAZIONE DI UNA NUOVA LIBRERIA

Una libreria per Arduino è costituita da due file fondamentali che sono un file .h ed un file .cpp, ed un file opzionale che è il keywords.txt.

Il file .h o header file contiene la dichiarazione della classe, tutte le funzioni usate nella stessa e le definizioni delle variabili necessarie, il file .cpp contiene il codice vero e proprio di ogni



funzione , mentre il file keywords.txt, che ripeto è opzionale, contiene le parole chiave e permette di visualizzare le chiamate alla libreria di colore diverso ogni qualvolta si invoca una singola funzione nello sketch.

Di fondamentale importanza è che sia la cartella che il file .h ed il file .cpp abbiano lo stesso nome, come vedremo di seguito, quindi dovremo innanzitutto creare una cartella con un nome che descriva intuitivamente per questione di comodità il nome della libreria, che contenga sia il file .h che il file .cpp ed eventualmente anche il keyword.txt **tutti con lo stesso nome**. Nel mio caso ho nominato libreria “LCD_m0” e di conseguenza, come accennato precedentemente, ho creato una cartella con lo stesso nome e con all’interno i tre file di cui abbiamo discusso prima, ottenendo il risultato seguente (Fig.1):

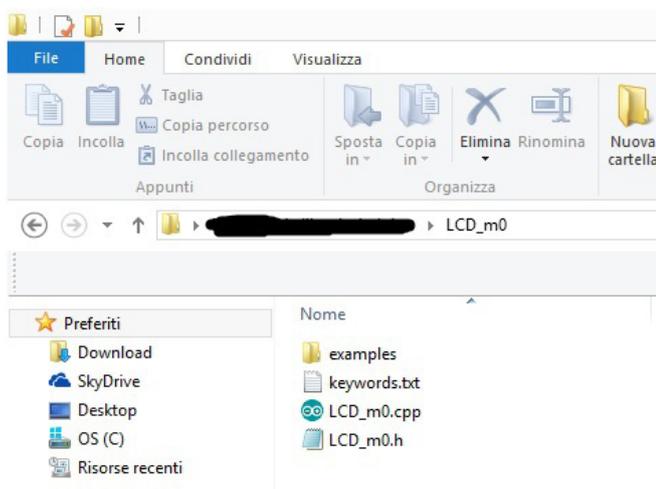


Fig.1

Nella cartella si nota una sottocartella con nome examples, della quale parleremo in seguito e che sostanzialmente contiene gli esempi della

libreria sviluppata che vengono messi a disposizione da chi la progetta, per facilitare la comprensione d’utilizzo della stessa.

FILE LCD_M0.H

Entriamo a questo punto in merito al file LCD_m0.h e vediamo come è strutturato (Fig.2).

```
#ifndef LCD_m0_h
#define LCD_m0_h

#include "Arduino.h"

class LCD_m0
{
public:
    .
    .
    .
private:
    .
    .
};
#endif
```

Fig.2

Come si può vedere ogni file header che definisce una classe, è delimitato dalle direttive **#ifndef** (Con al seguito il nome della libreria e l’identificativo **_h**) e **#endif** che devono sempre essere presenti.

Va inoltre usato sempre il **#define** seguito dal nome della libreria, in questo caso LCD_m0_h, che definisce la costante LCD_m0_h.

Detto questo vediamo cosa bisogna inserire all’interno della classe, che chiameremo sempre con il nome assegnato alla libreria, in que-

Quello che hai appena letto è solo un estratto, Firmware n. 118 è composto da ben 68 pagine ed è riservato agli abbonati PLATINUM!

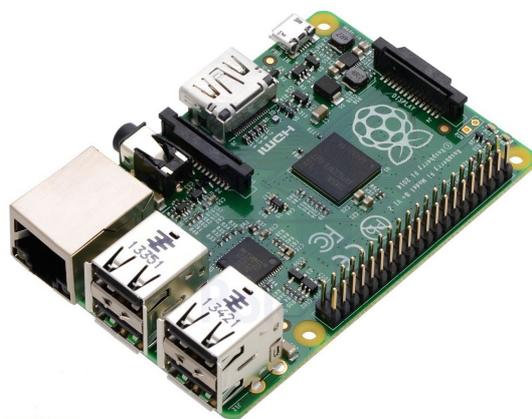


Sottoscrivi subito l'abbonamento PLATINUM, accedi agli articoli Firmware pubblicati in anteprima sul nostro blog ed effettua il download della rivista dedicata all'Elettronica Embedded ed ai Microcontrollori (in PDF, ePub e mobi) per un anno!

Inoltre, con l'abbonamento Platinum potrai accedere e scaricare un anno di EOS-Book, la rivista dedicata alla Scienza e alle Tecnologie, l'intero Archivio storico e gli Speciali monotematici (ben 43 numeri in PDF, ePub e mobi!!!)



E se ti abboni entro il 31 dicembre ricevi in omaggio (spese di spedizione a carico nostro) la nuova Raspberry Pi B+ , il mini-pc per realizzare tutti i tuoi progetti!!!



Che aspetti? ABBONATI ORA con PAYPAL è semplice e sicuro

SCOPRI DI PIU'