

LaurTec

Scheda per il controllo Motori 4Ax2

PJ3007

Sviluppa sistemi con motori Stepper Bipolari e DC a spazzole

Autore : *Mauro Laurenti*

ID: PJ3007-IT

INFORMATIVA

Come prescritto dall'art. 1, comma 1, della legge 21 maggio 2004 n.128, l'autore avvisa di aver assolto, per la seguente opera dell'ingegno, a tutti gli obblighi della legge 22 Aprile del 1941 n. 633, sulla tutela del diritto d'autore.

Tutti i diritti di questa opera sono riservati. Ogni riproduzione ed ogni altra forma di diffusione al pubblico dell'opera, o parte di essa, senza un'autorizzazione scritta dell'autore, rappresenta una violazione della legge che tutela il diritto d'autore, in particolare non ne è consentito un utilizzo per trarne profitto.

La mancata osservanza della legge 22 Aprile del 1941 n. 633 è perseguibile con la reclusione o sanzione pecuniaria, come descritto al Titolo III, Capo III, Sezione II.

A norma dell'art. 70 è comunque consentito, per scopi di critica o discussione, il riassunto e la citazione, accompagnati dalla menzione del titolo dell'opera e dal nome dell'autore.

AVVERTENZE

I progetti presentati non hanno la certificazione CE, quindi non possono essere utilizzati per scopi commerciali nella Comunità Economica Europea.

Chiunque decida di far uso delle nozioni riportate nella seguente opera o decida di realizzare i circuiti proposti, è tenuto pertanto a prestare la massima attenzione in osservanza alle normative in vigore sulla sicurezza.

L'autore declina ogni responsabilità per eventuali danni causati a persone, animali o cose derivante dall'utilizzo diretto o indiretto del materiale, dei dispositivi o del software presentati nella seguente opera.

Si fa inoltre presente che quanto riportato viene fornito così com'è, a solo scopo didattico e formativo, senza garanzia alcuna della sua correttezza.

L'autore ringrazia anticipatamente per la segnalazione di ogni errore.

Tutti i marchi citati in quest'opera sono dei rispettivi proprietari.

Indice

Introduzione	4
Specifiche Tecniche	4
Applicazioni	4
Smaltimento	4
Miglioramenti del progetto e compromessi	5
Analisi del progetto	6
Il microcontrollore.....	11
L'alimentatore.....	12
Il Ponte H.....	16
L'interfaccia RS485.....	19
Interfaccia Pulsanti.....	24
Interfaccia LED.....	25
Sensore di Temperatura e ventola di raffreddamento.....	26
Sensore di Corrente.....	28
Segnali di controllo Servo e fine corsa.....	31
Segnali Quadrature Encoder.....	34
Connettore di espansione e segnali di controllo.....	35
Segnali di emergenza.....	37
Stop di Emergenza (linea di ingresso).....	38
Stop di Emergenza (linea di uscita).....	40
Connettore di Programmazione e Debug.....	42
Layout Periferiche.....	44
Istruzioni per il montaggio	45
Collaudo e messa in funzione	48
Prima messa in funzione dopo il montaggio.....	48
Messa in funzione ordinaria.....	48
Software ed esempi	51
Funzioni di Sistema.....	52
Funzioni per Motori DC.....	52
Funzioni per Motore Stepper bipolare.....	53
Funzioni per interfaccia RS485.....	53
Funzioni per ADC.....	53
Funzioni per Emergency Stop.....	54
Funzioni varie.....	54
Esempio 1.....	55
Esempio 2.....	56
Esempio 3.....	57
Bibliografia	61
History	62

Introduzione

La scheda di controllo PJ3007 è pensata per poter supportare lo sviluppo di sistemi embedded per il controllo di motori Stepper bipolari, 4A per fase, due motori DC a spazzole, nonché un singolo motore DC fino a 8A. Il sistema è basato su PIC18F4431 progettato appositamente per il controllo di motori e alimentatori switching. Il progetto supporta applicazioni in cui la sicurezza è di particolare importanza, quali CNC e robot, mettendo in evidenza alcune possibili soluzioni per implementare Stop di emergenza, controlli e protezione del sistema.

Specifiche Tecniche

Alimentazione : 12VDC min. - 28VDC max.

Assorbimento : 8A max.

Corrente per fase: 4A max.

Dimensioni : 94 x 110 mm

Product Number: PJ3007

PCB Part Number : L5000-30001

Versione PCB : 4

Peso : 300g

Applicazioni

Il scheda PJ3007 supporta le seguenti applicazioni:

- Motori Stepper bipolari max. 4A per fase.
- Due Motori DC a spazzole max. 4A.
- Un Motore DC a spazzole max. 8A.
- Controllo via RS485.
- Controllo Step, Direction, Enable per CNC.
- Modalità Full Step e Half Step (Motori Stepper)
- Modalità LAP, Signed Magnitude (Motori DC)

Smaltimento



Secondo la Direttiva Europea 2002/96/EC tutti i dispositivi elettrici/elettronici devono essere considerati rifiuti speciali e non devono essere gettati tra i rifiuti domestici. La gestione e lo smaltimento dei rifiuti elettrici/elettronici viene a dipendere dalle autorità locali e governative. Un corretto smaltimento dei rifiuti permette di prevenire conseguenze negative per l'ambiente e ai suoi abitanti. E' obbligo morale, nonché legale, di ogni singolo cittadino, di attenersi alla seguente Direttiva. Per ulteriori chiarimenti l'Autore è a disposizione.

Miglioramenti del progetto e compromessi

Ogni progetto, in quanto tale, rappresenta una serie di compromessi che cercano di soddisfare tutte le specifiche di progetto. La scheda PJ3007 è stata progettata con le intenzioni di differenziarsi dalle altre schede commerciali per motori DC e stepper bipolari. La maggior parte delle schede è infatti sviluppata come lo schema base proposto nel datasheet del ponte H utilizzato, e non permette dunque di capire perché una scheda con ponte H da 4A possa costare 10Euro, mentre una scheda professionale con ponte H da 4A possa costare oltre 300 Euro.

Purtroppo le schede professionali presenti sul mercato sono caratterizzate dall'avere una breve documentazione sui connettori d'interfaccia e interconnessioni, per cui i dilemmi su cosa possa far lievitare i prezzi non sono svelati.

La scheda PJ3007 ha lo scopo, almeno parziale, di colmare questo buco nero, mettendo in luce considerazioni e accorgimenti progettuali che sono tipici di schede professionali e soprattutto trattare aspetti legati alla sicurezza, praticamente non considerati nelle schede a basso costo.

Sebbene il progetto sia nato con una certa presunzione, alcuni compromessi di carattere economico e spazio hanno portato a fare determinate scelte.

L'aver utilizzato componenti through hole (ovvero con terminali) ha il vantaggio di permettere a chiunque di poter montare la scheda in casa, ma lo svantaggio di occupare molto spazio. Se si fossero utilizzati componenti SMD si sarebbe potuto, sulla stessa area della scheda, aggiungere ulteriori filtri EMI ad altre linee di segnali. Altro componente che per ragioni di spazio e costi non è stato montato sulla scheda ma che sarebbe bene montare almeno sulla scheda di alimentazione, è un idoneo common mode choke al fine di limitare i disturbi di modo comune che potrebbero provenire dalla rete ma soprattutto essere trasferiti sulla rete di alimentazione. Come compromesso si potrebbe pensare in ogni modo di utilizzare anelli di ferrite intorno alle linee di alimentazione principale della scheda.

La parte analogica relativa alla lettura della corrente potrebbe essere migliorata facendo uso di amplificatori a guadagno variabile e ottimizzare in questo modo la lettura della corrente in funzione della coppia o del motore utilizzato. Anche in questo caso per ragioni di costo si è scelto di non far uso di amplificatori a guadagno variabile.

L'utilizzo di CPU con architettura a 16 bit – 32 bit è normalmente più idonea qualora si vogliano implementare algoritmi complessi di controllo ma il PIC18F4431 rappresenta un buon compromesso ad 8 bit grazie alle periferiche dedicate allo sviluppo di schede per il controllo di motori. Questo ha permesso di semplificare notevolmente gli esempi sviluppati per il controllo dei vari motori senza rischiare di perdere l'attenzione a causa di architetture troppo complesse o algoritmi sofisticati.

Nonostante la scheda sia progettata con in mente la protezione del sistema e la sicurezza degli utilizzatori finali, ulteriori precauzioni potrebbero essere adottate oltre a quelle che saranno descritte. In particolare le linee e segnali che escono dalla scheda potrebbero essere tutti disaccoppiati per mezzo di optoisolatori, oltre alle protezioni per mezzo di resistori e diodi. Allo stesso modo le alimentazioni sui connettori ausiliari potrebbero essere fornite per mezzo di alimentazioni distinte ed in particolare convertitori DC-DC isolati.

Sebbene la descrizione della scheda inizi con critiche sulla stessa, vi renderete presto conto che rappresenta un'ottima scheda di sviluppo per la comprensione delle problematiche che sono dietro la progettazione dell'hardware e software di controllo per motori DC e Stepper Bipolari. Molte problematiche possono essere inoltre estese ad altri ambiti facendo del progetto un buon riferimento per quelle applicazioni in cui siano richieste considerazioni sul come proteggere un sistema e renderlo affidabile.

Analisi del progetto

In Figura 1 è riportato lo schema elettrico della scheda di sviluppo per motori Stepper e Motori DC a spazzole. Il progetto, grazie ad un accurato sviluppo, permette di realizzare un gran numero di applicazioni per il controllo di motori.

Il scheda oltre che ad essere pensata come piattaforma per imparare ed insegnare i motori Stepper bipolari e motori DC a spazzole, è pensata per assistere una progettazione modulare di sistemi più complessi, quali per esempio CNC, sistemi robot, sistemi puntamento telescopio o quant'altro si debba muovere sotto un opportuno controllo. L'attenta progettazione per la sicurezza permette inoltre lo studio e la comprensione di problematiche inerenti a sistemi embedded in cui siano utilizzati motori di potenza.

Per una corretta descrizione e facile comprensione dello schema, l'hardware verrà introdotto per blocchi funzionali, mostrando così che la complessità dello schema è solo legata all'impatto iniziale.

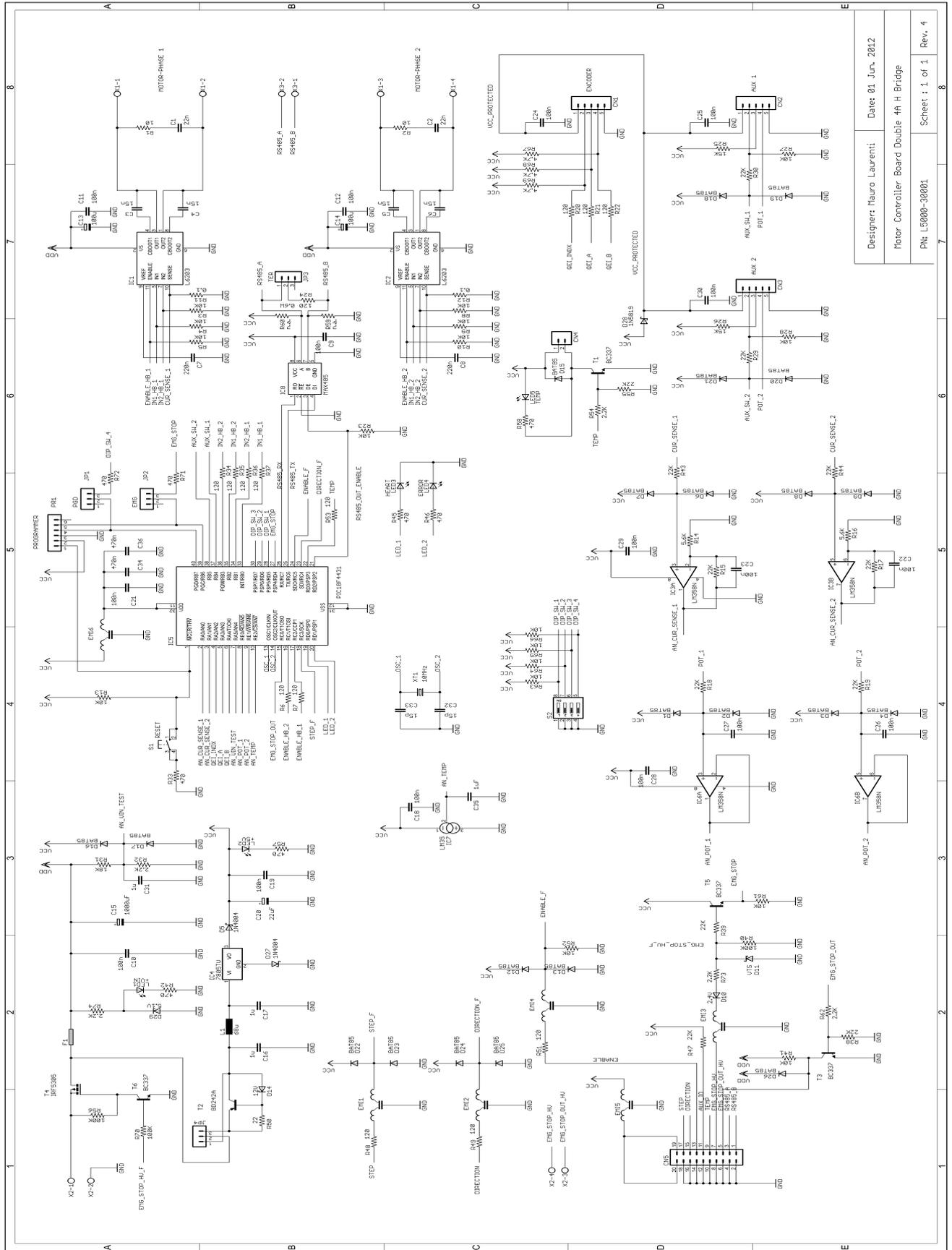


Figura 1: Schema elettrico della scheda PJ3007.

Lista Componenti

Resistori

R1 = 10Ω %5 2W Ossidi di Metallo

R2 = 10Ω %5 2W Ossidi di Metallo

R3 = 10KΩ %5 1/4W

R4 = 10KΩ %5 1/4W

R5 = 10KΩ %5 1/4W

R6 = 120Ω %5 1/4W

R7 = 120Ω %5 1/4W

R8 = 10KΩ %5 1/4W

R9 = 10KΩ %5 1/4W

R10 = 10KΩ %5 1/4W

R11 = 0.1Ω %5 3W Ossidi di Metallo

R12 = 0.1Ω %5 3W Ossidi di Metallo

R13 = 10KΩ %5 1/4W

R14 = 5.6KΩ %5 1/4W

R15 = 22KΩ %5 1/4W

R16 = 5.6KΩ %5 1/4W

R17 = 22KΩ %5 1/4W

R18 = 22KΩ %5 1/4W

R19 = 22KΩ %5 1/4W

R20 = 120Ω %5 1/4W

R21 = 120Ω %5 1/4W

R22 = 120Ω %5 1/4W

R23 = 10KΩ %5 1/4W

R24 = 120Ω %5 0.6W Ossidi di Metallo

R25 = 15KΩ %5 1/4W

R26 = 15KΩ %5 1/4W

R27 = 10KΩ %5 1/4W

R28 = 10KΩ %5 1/4W

R29 = 22KΩ %5 1/4W

R30 = 22KΩ %5 1/4W

R31 = 18KΩ %5 1/4W

R32 = 2.2KΩ %5 1/4W

R33 = 470Ω %5 1/4W

R34 = 120Ω %5 1/4W

R35 = 120Ω %5 1/4W

R36 = 120Ω %5 1/4W

R37 = 120KΩ %5 1/4W

R38 = 22KΩ %5 1/4W

R39 = 22KΩ %5 1/4W

R40 = 100KΩ %5 1/4W

R41 = 10KΩ %5 1/4W

R42 = 470Ω %5 1/4W

R43 = 22KΩ %5 1/4W

R44 = 22KΩ %5 1/4W

R45 = 470Ω %5 1/4W

R46 = 470Ω %5 1/4W

R47 = 22KΩ %5 1/4W

R48 = 120Ω %5 1/4W

R49 = 120Ω %5 1/4W

R50 = 22Ω %5 1/4W

R51 = 120Ω %5 1/4W

R52 = 10KΩ %5 1/4W

R53 = 120Ω %5 1/4W

R54 = 2.2KΩ %5 1/4W

R55 = 22KΩ %5 1/4W

R56 = 100KΩ %5 1/4W

R57 = 470Ω %5 1/4W

R58 = 470Ω %5 1/4W

R59 = vedi paragrafo Interfaccia RS485.

R60 = vedi paragrafo Interfaccia RS485.

R61 = 10KΩ %5 1/4W

R62 = 2.2KΩ %5 1/4W

R63 = 10KΩ %5 1/4W

R64 = 10KΩ %5 1/4W

R65 = 10KΩ %5 1/4W

R66 = 10KΩ %5 1/4W

R67 = 4,7KΩ %5 1/4W (opzionale, vedi paragrafo QEI)

R68 = 4,7KΩ %5 1/4W (opzionale, vedi paragrafo QEI)

R69 = 4,7KΩ %5 1/4W (opzionale, vedi paragrafo QEI)

R70 = 100KΩ %5 1/4W

R71 = 470Ω %5 1/4W

R72 = 470Ω %5 1/4W

R73 = 2.2KΩ %5 1/4W

R74 = 2.2KΩ %5 0.6W Ossidi di Metallo

Condensatori

C1 = 22nF ceramico 50V
C2 = 22nF ceramico 50V
C3 = 15nF ceramico 50V
C4 = 15nF ceramico 50V
C5 = 15nF ceramico 50V
C6 = 15nF ceramico 50V
C7 = 220nF ceramico 50V
C8 = 220nF ceramico 50V
C9 = 100nF ceramico 50V
C10 = 100nF ceramico 50V
C11 = 100nF ceramico 50V
C12 = 100nF ceramico 50V
C13 = 100 μ F elettrolitico 63V 105°C
C14 = 100 μ F elettrolitico 63V 105°C
C15 = 1000 μ F elettrolitico 63V 105°C
C16 = 1 μ F ceramico 50V
C17 = 1 μ F ceramico 50V
C18 = 100nF ceramico 50V
C19 = 100nF ceramico 50V
C20 = 22 μ F elettrolitico 35V 105°C
C21 = 100nF ceramico 50V
C22 = 100nF ceramico 50V
C23 = 100nF ceramico 50V
C24 = 100nF ceramico 50V
C25 = 100nF ceramico 50V
C26 = 100nF ceramico 50V
C27 = 100nF ceramico 50V
C28 = 100nF ceramico 50V
C29 = 100nF ceramico 50V
C30 = 100nF ceramico 50V
C31 = 1 μ F ceramico 50V
C32 = 22pF ceramico 50V
C33 = 22pF ceramico 50V
C34 = 470nF ceramico 50V
C35 = 1 μ F ceramico 50V
C36 = 470nF ceramico 50V

Quarzi

XT1 = 10MHz

Diodi

LED1 = LED 3mm verde
LED2 = LED 3mm verde
LED3 = LED 3mm verde
LED4 = LED 3mm rosso
LED5 = LED 3mm giallo

D1 = BAT85
D2 = BAT85
D3 = BAT85
D4 = BAT85
D5 = 1N4004
D6 = BAT85
D7 = BAT85
D8 = BAT85
D9 = BAT85
D10 = Zener 2.4V 0.5W
D11 = P6KE 39A (Voltage Transient Suppressor)
D12 = BAT85
D13 = BAT85
D14 = Zener 12V 0.5W
D15 = BAT85
D16 = BAT85
D17 = BAT85
D18 = BAT85
D19 = BAT85
D20 = BAT85
D21 = BAT85
D22 = BAT85
D23 = BAT85
D24 = BAT85
D25 = BAT85
D26 = BAT85
D27 = 1N4004
D28 = 1N5819
D29 = Zener 5.1V 0.5W

Transistor**T1:** BC337-40**T2:** BD242A**T3:** BC337-40**T4:** IRF5305 (PMOS)**T5:** BC337-40**T6:** BC337-40**Circuiti Integrati****IC1** = L6203**IC2** = L6203**IC3** = LM358N**IC4** = 7805 (TO220)**IC5** = PIC18F4431-I/P**IC6** = LM358N**IC7** = LM35 (TO92)**IC8** = MAX485**Connettori****CN1** = con-amp-quick 5 pin verticale**CN2** = con-amp-quick 5 pin verticale**CN3** = con-amp-quick 5 pin verticale**CN4** = con-amp-quick 2 pin verticale**CN5** = con-harting 20 pin maschio verticale**X1** = con-wago 4 fori**X2** = con-wago 4 fori**X3** = con-wago 2 fori**PR1** = Jumper 6 pin (90 gradi)**JP1** = Jumper 3 pin (PGD)**JP2** = Jumper 3 pin (EMG)**JP3** = Jumper 3 pin (TER)**JP4** = Jumper 3 pin**Pulsanti****S1** = micro-pulsante per PCB**S2** = DIP Switch 4**Induttori****L1:** 68uH 1.1A**EMI1:** DSS6NZ82A 103 (murata)**EMI2:** DSS6NZ82A 103 (murata)**EMI3:** DSS6NZ82A 103 (murata)**EMI4:** DSS6NZ82A 103 (murata)**EMI5:** DSS6NZ82A 103 (murata)**EMI6:** DSS6NZ82A 103 (murata)**Componenti vari****F1:** Fusibile 10A ritardato e porta fusibile 22.5mm**HS:** Aletta di Raffreddamento**FN:** Ventilatore 5V 40x40mm**Z1:** Zoccolo per IC 40 pin (x1)**Z2:** Zoccolo per IC 8 pin (x3)**SR1:** Viti 3Mx6mm (x8)**SR1:** Viti 3Mx10mm (x2)**RI1:** Anelli plastica per viti**KT:** Kit isolamento elettrico TO220 (2)**SP:** Gommini spaziatori (4)**JP:** Jumper selezione (4)

Il microcontrollore

Come prima cosa cerchiamo di capire con quali microcontrollori sia possibile utilizzare la scheda di sviluppo PJ3007. La scheda è progettata ed ottimizzata per lo sviluppo di applicazioni con il microcontrollore PIC18F4431 a 40 pin. Quindi non deve essere pensata come la scheda Freedom II, che supporta PIC16 e PIC18 a 40 pin di tipo generico. Il PIC18F4431 è ad 8 bit ed è pensato per lo sviluppo di applicazioni di potenza tra cui il controllo di motori DC. Il microcontrollore possiede due moduli PWM che permettono di controllare il motore sia in modalità LAP che Signed Magnitude. Il PIC possiede inoltre il modulo QEI ovvero l'interfaccia per un Encoder in quadratura, permettendo la gestione dello stesso con un minimo intervento da parte della CPU. Oltre a queste due periferiche ottimizzate per il controllo dei Motori, il PIC18F4431 possiede una certa flessibilità nel modulo ADC che può essere utilizzato fino ad una frequenza di campionamento di 200KHz.

ovvero di presenza tensione e il condensatore di Bulk C15 che permette di fornire rapidamente correnti elevate ai ponti H quando richiesto. Il condensatore di Bulk deve essere del tipo utilizzato negli alimentatori switching, ovvero da 105°C e a basso ESR (Resistenza serie Parassita). Questa esigenza discende dal fatto che avere un basso ESR permette di avere una risposta in frequenza migliore, ovvero il condensatore riesce rapidamente a fornire correnti elevate durante i transitori causati per esempio da un cambio del verso di rotazione di un motore. La caratteristica di poter lavorare ad alte temperature ovvero lavorare a 105°C piuttosto che 85°C, permette un più lento deterioramento dello stesso. Si fa infatti presente che schede di controllo motori che possono lavorare con correnti di 4A per fase possono scaldare notevolmente causando un rapido deterioramento di componenti di bassa qualità. Questa osservazione è valida per tutti i condensatori elettroliti in prossimità dell'aletta di raffreddamento.

L'uscita Vdd è posta sul partitore R31-R32 al fine di poter misurare la tensione stessa. Il partitore è necessario poiché il PIC18F4431 essendo alimentato a 5V può accettare al suo ingresso tensioni massime di 5V. Il condensatore C31 rappresenta un semplice filtro per limitare i disturbi in ingresso al PIC. Dal Momento che il PIC18F4431 è alimentato per mezzo di Vcc, che viene ottenuta dalla tensione primaria prima del fusibile F1, è possibile rilevare la rottura del fusibile, che si ha quando Vdd=0V. I diodi D16 e D17, permettono di proteggere l'ingresso analogico del PIC qualora, siano presenti dei picchi di tensione. Come si vedrà a breve, molti ingressi sono protetti per mezzo di due diodi Schottky. Questa esigenza discende dal fatto che carichi induttivi come i motori possono creare extra tensioni che possono causare la rottura del PIC come anche di altri semiconduttori qualora questi non siano propriamente protetti³.

La tensione Vcc a 5V, viene fornita con un normale regolatore lineare della serie 78xx. Proprio per questa ragione è necessaria qualche precauzione, viste le tensioni in gioco. Vediamo l'ingresso del regolatore. In primo luogo è possibile notare la presenza del transistor T2 e di un filtro PI (ovvero p-greco). Il filtro p-greco è utilizzato per limitare i disturbi dell'alimentazione principale verso Vcc. Il regolatore lineare ha di suo un'elevata capacità di filtraggio (PSRR Power Supply Rejection Ratio) e per molte applicazioni questo potrebbe risultare sufficiente, ciononostante un filtraggio non idoneo può causare il reset della CPU per questo si è aggiunto anche il filtro PI. Dal momento che i motori possono lavorare con frequenze PWM dell'ordine 3KHz-50KHz, il filtro è stato dimensionato per una frequenza di taglio di 1KHz.

La presenza del transistor T2 e del diodo zener D14 è necessaria per abbassare la tensione in ingresso, in particolare tale configurazione rappresenta un diodo zener di potenza realizzato per mezzo di un diodo zener a bassa potenza. L'esigenza di avere questo diodo zener di potenza per abbassare la tensione in ingresso discende dal fatto che il regolatore 7805 accetta tensioni in ingresso massime di 35V, ovvero prossimi alla tensione massima di alimentazione. La configurazione diodo zener permette di sottrarre la tensione pari al diodo più la caduta di tensione Vbe del transistor di potenza ovvero circa 14V. Dal momento che il regolatore 7805 richiede una tensione minima di 7.5V, vuol dire che il diodo zener deve essere attivato per mezzo del Jumper JP4, solo quando la tensione risulti di almeno di 24V. Per sottrarre i 14V è necessario che il Jumper JP4 sia posizionato con il ponticello tra 2 e 3 (posizione nominata HIGH sul Jumper JP4). Per disattivare la funzione, il ponticello del Jumper deve essere posizionato tra il pin 1 e 2 (posizione nominata LOW). Qualora la tensione +5V provenga dal connettore CN5, il ponticello deve essere rimosso.

Sull'uscita del regolatore lineare non si ha nulla di particolare, troviamo un diodo LED di segnalazione e i condensatori di filtro. L'unica stranezza è la presenza del diodo D5. La sua presenza è legata al fatto che la tensione Vcc potrebbe, come detto, essere fornita anche dal connettore di

³ Componenti passivi come i condensatori non sono da meno. Infatti tensioni eccessive possono causare la foratura del dielettrico. Per rendere il sistema robusto i condensatori sono stati scelti con tensioni operative circa il doppio di quella operativa della scheda stessa.

controllo CN5, in questo caso D5 si troverà ad essere interdetto prevenendo che correnti inverse circolino nel regolatore lineare 78xx, potendone causare la rottura. Al fine di compensare la caduta di tensione sul diodo D5 è stato introdotto il diodo D27. Visto che la caduta di tensione su D5 viene compensata da D27, non è necessario utilizzare diodi Schottky a bassa caduta, ma semplici diodi 1N4004.

A questo punto vi sarete posti la domanda perché non abbia usato un regolatore switching in configurazione Buck! La domanda me la sono posta anche io ed ho voluto evitare tale soluzione per varie ragioni, tra cui le quali il costo. Per le tensioni in gioco è necessario utilizzare regolatori switching ad alta tensione che per piccoli acquisti possono costare circa 5 Euro. In aggiunta a questo sarebbe necessario aggiungere l'induttore che potrebbe avere un costo di altri 1-2 Euro. Facendo uso della soluzione sopra proposta il progetto ha potuto raggiungere lo stesso scopo con circa 1 Euro. Altra ragione che mi ha spinto a non utilizzare l'alimentatore switching è stato il fatto che le correnti in gioco sono di circa 10mA (100mA quando la ventola risulta attiva), per cui l'efficienza dell'alimentatore non sono molto importanti. L'utilizzare un regolatore lineare ha inoltre aggiunto il vantaggio di filtrare l'alimentazione principale, molto rumorosa per la presenza dei motori. L'utilizzo di un alimentatore switching al posto del regolatore lineare avrebbe richiesto ulteriori filtri visto che l'alimentatore switching avrebbe aggiunto ulteriore rumore oltre a quello dei motori. Quest'ultima ragione mi ha convinto per l'intraprendere senza indugi la strada del regolatore lineare. Facendo vari test di carico tra cui correnti impulsive con picchi fino a 10A (ponti H in parallelo), non sono mai riuscito a causare un Reset della CPU. Questo mostra come scelte oculate e compromessi possano portare buoni frutti.

Un'ultima nota...anche se prima ho affermato il contrario! Sebbene le correnti per l'alimentazione 5V siano soli 100mA massimo, se si considera il caso peggiore, ovvero quando si ha una tensione di 28V, comporta che il dispositivo debba dissipare $(30-5) \cdot 0.1$ W ovvero 2.3W. Al fine di limitare la potenza dissipata nel regolatore 7805 si ha il diodo zener di potenza che per tensioni superiori a 24V sottrae circa 14V, persi sul transistor T2, per cui la metà della potenza è dissipata sul transistor di potenza. Questo permette di aumentare notevolmente la vita media dell'alimentatore ovvero del sistema.

Il fatto di far lavorare un componente in un range 50%-80% del valore massimo, permette in generale di avere un prodotto robusto. Non di rado far lavorare un sistema al 50%-80% del valore massimo consentito rappresenta una specifica di progetto al quale bisogna attenersi, al fine di ottenere un buon progetto.

La presenza della tensione di alimentazione Vdd è segnalata per mezzo del LED1. Il LED trovandosi dopo il PMOS T4 e il fusibile, segnala la presenza della tensione solo se T4 è attivo e il fusibile è funzionante. Il LED risulta spento in caso di rottura del Fusibile o nel caso il PMOS T4 sia disattivo. Si noti che il regolatore lineare 7805 si trova prima del fusibile, per cui il microcontrollore rimane attivo anche qualora il fusibile si dovesse rompere. In particolare misurando la tensione Vdd, il microcontrollore può, come detto, rilevare la rottura del fusibile stesso.

La circuiteria intorno al LED1 è leggermente più complessa di quella a cui siamo abituati, ovvero basata su un semplice resistore, infatti è presente R42, R74, D29.

In un primo momento avevo deciso di usare un solo resistore che nel caso di alimentazione di Vdd max. mi permettesse di avere una corrente di circa 5mA sul LED. Questa soluzione sarebbe andata bene se la scheda fosse stata alimentata sempre a Vdd max., mentre la scheda in questione può essere alimentata a partire da 12V. A queste tensioni il LED1 seppure acceso verrebbe attraversato da basse correnti ottenendo luce molto più fioca, tanto da far sospettare malfunzionamenti anche se la scheda non dovesse avere problemi. Per limitare la variazione di tensione sul LED al variare della tensione in ingresso Vdd, avevo pensato inizialmente di usare la tensione a 5V in uscita dal regolatore 7805 come

riferimento costante, ma questo avrebbe legato il funzionamento del LED1 sia a Vdd che Vcc. Per rendere l'accensione del LED1 dipendente solo da Vdd ho deciso di creare una tensione costante per mezzo di un diodo zener da 5.1V e usare la tensione ai suoi capi come tensione costante per accendere il LED1. Il resistore R42 del LED1 è dello stesso valore usato per gli altri LED controllati dalle uscite del PIC18F4431.

Sebbene ora pensiate che ai capi del diodo zener ci siano 5.1V e il LED venga controllato da questa tensione al variare della tensione di ingresso tra 12V e 28V, vi sbagliate. Volendo limitare le correnti in gioco (ovvero la potenza dissipata sui componenti) ho preferito non far lavorare il diodo zener in zona zener nell'intero intervallo di alimentazione. Sebbene possibile questo richiederebbe potenze per i resistori di circa 1W al fine di far dissipare propriamente l'energia termica derivante dalle correnti necessarie per il diodo zener D29 e il LED. Facendo uso di una resistenza R74 di 2.2K si ha che la corrente che vi circola con Vdd pari a 28V è di circa 11mA per cui la potenza dissipata sul resistore è 0.28W. Quindi utilizzando una resistenza da 0.6W è possibile limitare le dimensioni del resistore utilizzabile.

Se si facesse circolare una corrente di 11mA quando Vdd dovesse essere di 12V verrebbe richiesta una resistenza di potenza maggiore al fine di permettere un funzionamento affidabile a 28V. Come detto le tensioni in gioco non permettono nell'intero intervallo di Vdd di avere i 5.1V ai capi del diodo zener, ciononostante permettono di avere una variazione di tensione sul LED1 tale da non causare un'eccessiva variazione della luminosità del LED al variare di Vdd, cosa che si avrebbe usando un semplice resistore.

Il Ponte H

Oltre al microcontrollore una parte importante di una scheda per il controllo dei motori è rappresentato dal ponte H, ovvero dallo stadio di potenza che permette di pilotare, in accordo ai segnali provenienti dal microcontrollore, il/i motori collegati⁴. La scheda PJ3007 possiede due ponti H del tipo L6203. In Figura 3 è riportato il dettaglio di un solo ponte, nominato Motor-Phase 1 ovvero Motore 1 o Fase 1. Il nome discende dal fatto che essendo presenti due ponti H, ogni singolo ponte può essere utilizzato per comandare un motore o una fase di un motore passo-passo bipolare.

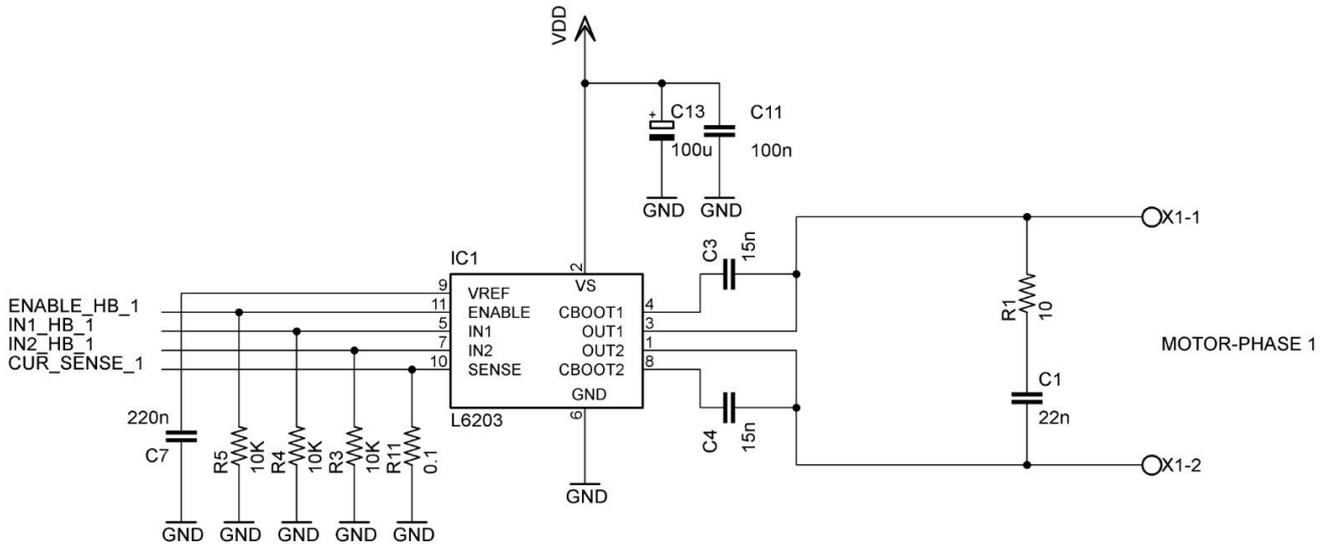


Figura 3: Schema elettrico relativo al ponte H (singolo canale).

Le caratteristiche principali di questo ponte sono⁵:

- Alimentazione massima 48V⁶.
- Corrente di picco 5A
- Corrente massima continua 4A rms
- $R_{DS(ON)} = 0.3\text{ohm}$ (valore tipico)
- Frequenza operativa fino a 100KHz
- Diodi di ricircolo interni
- Protezione termica⁷
- Input compatibili TTL
- DMOS output driver

La scelta del ponte L6203 è stata fatta per le seguenti proprietà:

- correnti elevate
- buona efficienza
- diodi di ricircolo interni
- facile reperibilità

⁴ Maggiori dettagli sul ponte H e tecniche di pilotaggio dei motori possono essere trovati nella Brief Note BN0006 “Automazione: Stadi di potenza per il controllo di Motori DC e passo-passo” scaricabile dal sito www.LaurTec.it.

⁵ Per maggiori dettagli si faccia riferimento al datasheet del ponte L6203 al sito www.st.com.

⁶ Sebbene il driver L6203 supporti tensioni fino a 48V la scheda PJ3007 è progettata e testata per tensioni fino a Vdd. max.

⁷ Il ponte viene disattivato al raggiungimento di una temperatura pari a 150 °C.

- Contenitore facile da montare
- Costo ridotto

Un'alternativa presa in considerazione è stato L298, ma le basse correnti, bassa efficienza⁸ e la necessità di diodi di ricircolo esterni avrebbe richiesto per due ponti un'area di PCB elevata. Un altro ponte preso in considerazione è stato LMD1800 caratterizzato dal un'ottima efficienza e correnti massime di 3A, diodo di ricircolo interni e piedino di warning per la temperatura. Il suo svantaggio risiede prevalentemente nel prezzo normalmente 2-5 volte più costoso di un ponte L6203. Altro aspetto che ho considerato è che diversi motori stepper con coppia 2-3 N·m, richiedono correnti di 3-4A per fase. Avendo un ponte come L6208 si ha la possibilità di pilotare in maniera più “rilassata” i motori da 3A ma al tempo stesso poter pilotare anche motori da 4A.

Vediamo qualche dettaglio della sezione dello schema riportato in Figura 3. Il ponte L6203 possiede ingressi di controllo Enable, IN1 e IN2. Il segnale di Enable permette di abilitare il ponte mentre i segnali IN1 e IN2 permettono di cambiare i rami attivi del ponte H o effettuare una frenatura elettromeccanica⁹. La linea di Enable è collegata al pin RC2 del PIC18F4431, questo permette di pilotare la linea sia con un normale segnale 0-1 ovvero attivo/disattivo, o per mezzo del segnale PWM del modulo CCP presente sul pin RC2.

Questo semplice accorgimento permette dunque di pilotare un motore applicando il segnale PWM sulla linea di Enable e utilizzare i segnali IN1 e IN2 per il controllo della direzione di rotazione (Modalità di controllo Signed Magnitude). Accorgimento analogo è stato preso per il secondo ponte, infatti il PIC18F4431 possiede due uscite PWM, la seconda della quale è posta su RC1, dove appunto è collegato l'Enable del secondo ponte H. Le linee IN1_HB_1 e IN2_HB_1 sono invece collegate alle linee RB0 e RB1. Queste linee hanno la caratteristica di poter comandare le linee IN1_HB_1 e IN2_HB_1 sia come semplici uscite digitali ma anche per mezzo del Power Control Module, che permette di comandare con segnali PWM complementari le linee di controllo¹⁰. Questo permette di comandare il motore in modalità LAP, ovvero utilizzare la linea Enable come semplice linea di attivazione e le linee IN1_HB_1 e IN2_HB_1 controllate con segnale PWM complementare. Cosa analoga è valida per il secondo ponte H le cui linee IN1_HB_2 e IN2_HB_2 sono rispettivamente collegate alle uscite RB2 e RB3.

Si noti che le linee Enable, IN_1 e IN_2 di ogni ponte possiedono un resistore di pull-down. Questo è fatto per garantire un livello logico noto quando il microcontrollore non ha preso ancora il controllo del ponte stesso, o ci dovessero essere anomalie che possano portare le uscite ad essere fluttuanti o ad alta impedenza. I resistori di pull-down permettono in particolare di disabilitare il ponte e mettere in frenatura elettromeccanica il motore ad esso collegato. Il resistore R11 da 0.1Ohm permette di misurare la corrente che circola nel motore o nella fase del motore passo passo. Maggiori dettagli sono riportati nel paragrafo dedicato alla misura della corrente. Il condensatore C7 è necessario per il regolatore lineare presente all'interno del ponte L6203 utilizzato per abbassare la tensione principale proveniente da Vdd. Il regolatore permette dunque di alimentare la circuiteria di controllo interna al ponte senza necessitare di una doppia alimentazione come per esempio per il ponte L298.

Il condensatore C13 elettrolitico è posizionato in prossimità del ponte ed ha il compito fornire correnti in maniera più “locale” e rapida che non il condensatore C15 dell'alimentatore principale. Per migliorare la risposta in frequenza a distanza di pochi millimetri dal ponte stesso sono posizionati i

⁸ La bassa efficienza del ponte L298 discende prevalentemente dal fatto che lo stadio di potenza è realizzato per mezzo di transistor BJT.

⁹ Quando IN1=0 e IN2=1 si ha una rotazione in un verso mentre ponendo IN1=1 e IN2=0 si ha una rotazione in verso opposto. Quando IN1=IN2, siano 0 o 1, si ha una frenatura elettromeccanica.

¹⁰ Il dead time normalmente richiesto per ponti H discreti non è richiesto per il ponte H L6203 poiché già presente internamente. In particolare il dead time dell'L6203 è di 40ns.

condensatori di disaccoppiamento da 0.1 μ F. Al fine di garantire una buona risposta in frequenza si è anche fatto uso di condensatori elettrolitici a basso ESR.

Sul lato destro della Figura 3 sono presenti i condensatori C3 e C4 che rappresentano i condensatori di bootstrap, consigliati di almeno 10nF. La loro funzione è di poter fornire una tensione di circa 10V tra il gate e il source dei N-MOS utilizzati sul ponte (rami alti). Per gli N-MOS collegati con il source a massa (rami bassi), avere una differenza tra Gate e Source di 10V è semplice, visto che il source è a 0V. Questo non vale per il Source dei rami alti collegati ai Drain dell'NMOS del ramo basso. L'esigenza di avere 10V tra Gate e Surce è richiesta per attivare il gate ma anche garantire che la $R_{DS(ON)}$ sia sufficientemente bassa per avere una buona efficienza del ponte stesso¹¹. Utilizzando valori di condensatori inferiori a 10nF si potrebbe avere il problema che pur attivando gli N-MOS la $R_{DS(ON)}$ potrebbe essere troppo alta. Valori di 15nF sono consigliati ed utilizzati in diverse applicazione della ST.

La resistenza R1 e C1 rappresentano uno snubber, ovvero un circuito per mezzo del quale si rallenta la variazione di tensione derivante da variazioni di correnti su carichi induttivi. Rallentando i picchi di tensioni si dà la possibilità ai diodi di ricircolo di intervenire senza ritardi anche se non particolarmente rapidi. Se il diodo di ricircolo non dovesse essere rapido rispetto alle variazioni di tensione si rischierebbe di raggiungere extratensioni troppo elevate prima dell'intervento del diodo stesso, causando la rottura del ponte. Maggiori dettagli sul come dimensionare la rete di snubber è descritta nell'application note AN280 della ST "Controlling Voltage Transients in full bridge driver applications".

Nota

La rete di snubber presente sulla scheda PJ3007 è quella riportata sul datasheet ma potrebbe essere ottimizzata per le proprie applicazioni a seconda delle tensioni operative e correnti richieste dal motore.

Come visto la scheda PJ3007 possiede due ponti H, i quali permettono di comandare un motore stepper bipolare o due motori DC. I due ponti H possono essere anche messi in parallelo e raggiungere le correnti massime per un singolo motore pari a 8A rms (10A di picco). Per fare questo bisogna semplicemente collegare P1+ con P2+ e P1- con P2-. Il motore DC avrà dunque collegato il + a P1+ e P2+ mentre il - a P1- e P2-. In questo caso il programma di gestione dei ponti deve avere cura di inviare lo stesso comando ad entrambi i ponti¹².

¹¹ Maggiore è la resistenza $R_{DS(ON)}$ maggiore è l'energia dissipata internamente al ponte (maggiore riscaldamento). Infatti l'energia dissipata internamente è pari al prodotto tra $R_{DS(ON)}$ e il quadrato della corrente I che circola nel ponte.

¹² Per maggiori dettagli si faccia riferimento al paragrafo dedicato al software. In particolare facendo uso dell'impostazione MOTOR_ALL, piuttosto che MOTOR_1 o MOTOR_2 è possibile comandare direttamente entrambi i ponti H.

L'interfaccia RS485

La scheda PJ3007 rappresenta un sistema di sviluppo che permette di controllare in vari modi i motori DC o Stepper Bipolari. Sebbene siano molte le applicazioni per cui una sola scheda sia sufficiente, ve ne possono essere altre per cui sia necessario comunicare con un'altre schede o PC. Si pensi per esempio ad un sistema quale la CNC o un sistema di posizionamento di un telescopio. In questi casi il colloquio tra schede o un PC può ritornare particolarmente utile.

La scheda PJ3007 supporta per tale ragione l'interfaccia RS485 comandata per mezzo dell'UART interna al microcontrollore. L'interfaccia RS485 è stata scelta per la sua semplicità di utilizzo e l'affidabilità della trasmissione dei dati. Si ricorda infatti che l'interfaccia RS485 è uno standard che prevede un segnale differenziale. In Figura 4 è riportato un dettaglio dello schema elettrico relativo all'interfaccia RS485.

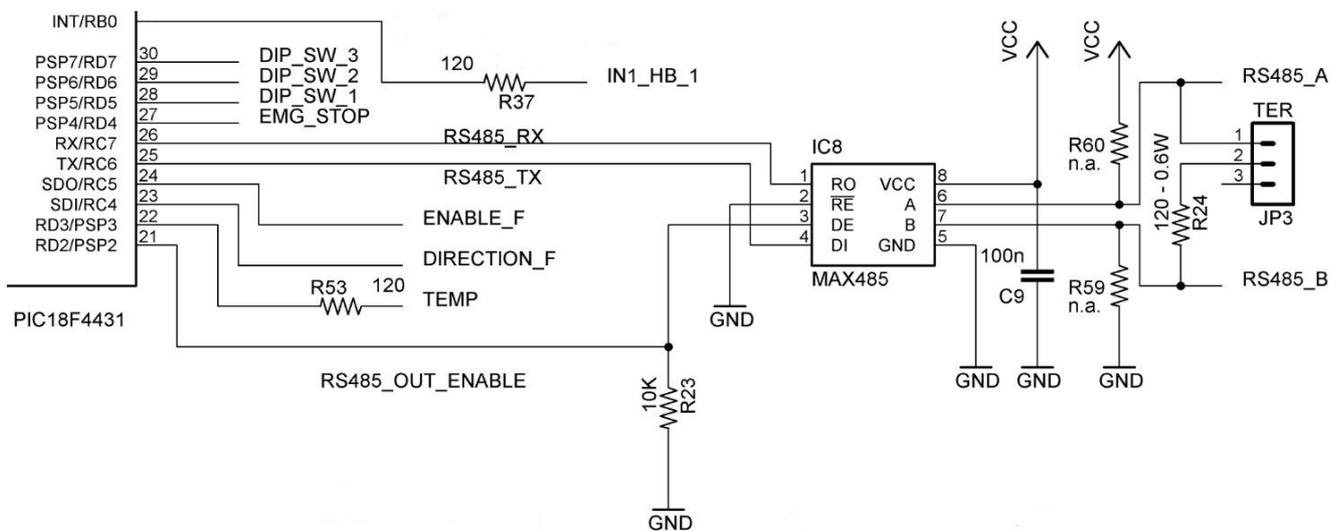


Figura 4: Interfaccia RS485

Tra le caratteristiche dell'interfaccia RS485, si ricorda, oltre al fatto di poter essere abbinata alla UART, che descrive uno standard half duplex, ovvero dove il canale di trasmissione e ricezione fanno uso della stessa linea di trasmissione. Per tale ragione, il microcontrollore deve arbitrare la comunicazione o seguire un arbitro (master) al fine di non corrompere la trasmissione. Lo standard RS485 è stato pensato per sistemi in cui sia presente un solo master (arbitro) e differenti slave, che rispondono, trasmettendo dati, solo sotto richiesta e arbitraggio del master. In questo modo si evita che più sistemi possano trasmettere sul bus quando questo risulta occupato. Lo standard RS485 non prevede alcun arbitraggio come per lo standard CAN, infatti lo standard RS485 descrive solo il cosiddetto physical layer, ovvero l'interfaccia fisica di comunicazione. Ciononostante è possibile implementare qualche forma di arbitraggio e correzione di errori, ovvero implementare un protocollo più o meno complesso, ma questo deve essere fatto dall'application layer ovvero via software.

Dalla Figura 4 è possibile vedere che l'UART, al fine d'interfacciarsi con il bus RS485 richiede un transceiver, in particolare si è usato il MAX485. Il transceiver, possiede la linea RE (pin 2) sempre a massa, per cui il canale di ricezione è sempre attivo. Questo significa che il microcontrollore riceve sempre quello che si sta trasmettendo (funzione echo). La linea DE (pin 3) che controlla invece il buffer di trasmissione, viene controllata dal microcontrollore. Infatti un modulo che non trasmette sul bus deve disattivare il canale di trasmissione (l'uscita del buffer viene posta ad alta impedenza). Al fine di garantire che il driver di trasmissione sia disattivo anche durante la fase di inizializzazione del microcontrollore, la linea DE possiede un resistore di pull-down.

Si osservi che il transceiver possiede in uscita tre resistori R59, R60 e R24. Il resistore R24 rappresenta la terminazione del bus. Si ricorda che la terminazione viene utilizzata all'inizio e alla fine del bus, quindi non tutti i moduli devono averla attiva¹³. La resistenza è di tipo a film metallico al fine di avere ridotti "effetti induttivi" che possono causare errori di trasmissione. I resistori R59 e R60 non sono montati sulla scheda. La loro presenza è richiesta su un solo dispositivo della rete e rappresentano rispettivamente un resistore di pull-up e pull-down. Il loro valore viene a dipendere dal numero di dispositivi sulla rete e serve a garantire che anche quando tutti i transceiver sono disattivi (ovvero in ricezione), sulla linea sia presente un valore logico valido (questo evita, o meglio limita, che del rumore di linea possa essere interpretato come un dato trasmesso). Vediamo qualche dettaglio sulla necessità del resistore di terminazione e come determinare i resistori di polarizzazione.

Resistore di terminazione

In generale ogni qual volta si debba trasmettere un'informazione per mezzo di una linea elettrica (bus) bisogna porsi la domanda se è necessaria o meno una resistenza di terminazione ai capi della linea stessa, questa domanda pertanto si applica anche al bus RS485.

Senza entrare nel dettaglio delle problematiche legate alla trasmissione dei segnali al fine di garantire una corretta trasmissione dei dati¹⁴, in questa sede ricordo solo che la resistenza di terminazione permette di ridurre gli effetti nocivi delle riflessioni. In particolare la resistenza di terminazione deve essere posta all'inizio e alla fine della linea di trasmissione e deve avere il valore pari alla Z₀ della linea di trasmissione ovvero dell'impedenza caratteristica del cavo utilizzato. In linea di massima non avendo spesso le caratteristiche del cavo preso dal cassetto non si può avere un matching perfetto ma terminando entrambi i lati con la stessa resistenza da 120 ohm non si hanno in generale molti problemi.

La necessità di una resistenza di terminazione può facilmente essere vista per mezzo di strumenti ad hoc ma anche per mezzo dell'oscilloscopio.

Come regola empirica si può comunque determinare a priori l'esigenza o meno della resistenza di terminazione, confrontando i tempi caratteristici del segnale elettrico usato nella trasmissione e i tempi di percorrenza del segnale stesso sulla linea di trasmissione. I tempi caratteristici per una trasmissione digitale sono in particolare la durata del bit e i tempi di salita e discesa dei fronti. Il primo parametro è legato alla velocità di trasmissione (baudrate), mentre il secondo è in generale imposto dal transceiver utilizzato¹⁵.

Per il protocollo RS485 considerando di usare un MAX485 si hanno tempi di salita e discesa dei fronti compresi tra 250ns e 2000ns. Utilizzando una trasmissione a 19200bit/s si ha che la durata del bit è circa 50us.

Considerando di utilizzare la scheda per il controllo dei motori in sistemi quali robot, i cavi probabilmente non saranno più lunghi di 1m. In CNC si potrebbero avere cavi di 2-3 metri mentre in applicazioni remote quali per esempio il controllo di un telescopio potrei pensare a cavi di circa 10m. A solo scopo di esempio prendiamo il caso peggiore con 10m di cavo. Dal momento che un segnale elettrico viaggia su un mezzo metallico a circa 2/3 della velocità della luce che è pari a 300.000.000 m/s, si ha che la velocità di un segnale elettrico è circa 200.000.000m/s. Per tale ragione considerando il cavo di 10m si ha che il tempo di percorrenza è pari a:

¹³ Per inserire la resistenza di terminazione bisogna porre il ponticello tra il pin 1-2 del jumper JP3 (TER).

¹⁴ Vi sono molte tecniche e trucchi oltre alla resistenza di polarizzazione, necessarie per garantire una corretta trasmissione di informazioni. Testi interi trattano l'argomento della Teoria dei segnali in maniera dettagliata. La sua difficoltà da un punto di vista pratico ha fatto nascere molti integrati che possiedono una circuiteria di compensazione idonea a compensare gli effetti di una linea di trasmissione, permettendo di raggiungere elevate distanze di trasmissione senza eccessive complicazioni dal lato della progettazione. Questi integrati vengono prodotti per applicazioni in cui le frequenze di trasmissione sono dell'ordine del Gb/s.

¹⁵ In generale oltre che al transceiver è importante considerare la particolare tecnologia utilizzata per realizzare le porte logiche. Tecnologie più veloci avranno fronti di salita e discesa più rapidi rispetto alle tecnologie più lente.

$$t = \frac{s}{v} = \frac{10m}{200.000.000 m/s} = 50^{-9} s = 50 ns$$

dove :

t : tempo di percorrenza della linea.

s: lunghezza della linea.

v: velocità del segnale sulla linea.

Essendo il tempo di percorrenza molto minore¹⁶ della durata del bit si ha che la frequenza di trasmissione non richiede la presenza della resistenza di terminazione. Il limite imposto dalla frequenza di trasmissione, in 10 metri di cavo si verrebbe ad avere qualora la frequenza dovesse essere maggiore di:

$$f = \frac{1}{10 \cdot 50^{-9}} = 200.000 bit/s = 200 Kb/s$$

Nel nostro esempio il tempo caratteristico del segnale più piccolo è quello sui fronti di salita e discesa che sono compresi tra 250ns e 2000ns. Il caso peggiore del fronte sarebbe di 250ns, ovvero 5 volte maggiore del tempo di percorrenza. Il caso peggiore sul fronte richiederebbe dunque la presenza della terminazione. In realtà per i fronti di salita si usano spesso dei margini più rilassati dell'ordine di 3-5 per cui anche sui fronti di salita e discesa del nostro esempio si potrebbe non usare la terminazione. Considerando il tempo massimo dei fronti ovvero 2000ns non sarebbe invece sicuramente necessario utilizzare la terminazione anche usando un fattore per il confronto molto più grande ovvero pari a 10. Per avere però un progetto robusto bisogna considerare il caso peggiore ovvero i 250ns.

Sebbene questo sia un caso un po' limite si capisce che usando un cavo di 5m piuttosto di 10m ci si trova già in una situazione in cui i tempi di percorrenza si dimezzano per cui anche i fronti di salita e discesa del transceiver non sono più un problema.

Alcuni transceiver RS485 hanno i tempi di rise e fall time di circa 3ns per cui usando lo stesso esempio sopra sarebbe necessario la resistenza di terminazione. In casi in cui si abbiano dei fronti rapidi si ha in generale un maggior problema di radiazioni di onde elettromagnetiche per cui non è inusuale vedere in serie al bus (una per linea) dei resistori di qualche decina di ohm. Questo permette di allungare i tempi di rise e fall time e i tempi di percorrenza della linea stessa.

Qualora non si abbia un oscilloscopio per verificare eventuali alterazioni sui bit e ci si trovi in un caso limite è possibile inserire la resistenza di terminazione. Gli svantaggi dell'inserire la resistenza di terminazione sono legati all'aumento dei consumi dell'ordine di qualche decina di mA e la diminuzione dell'ampiezza del segnale disponibile.

Resistori di polarizzazione

Il bus RS485 è half duplex, ovvero il bus è utilizzato sia per trasmettere che per ricevere dati, per cui un dispositivo non può trasmettere e ricevere dati allo stesso tempo a meno di non ricevere i dati che sta trasmettendo lui stesso. In particolare al fine di armonizzare un bus di questo tipo è necessario poter abilitare e disabilitare i buffer di uscita in maniera che un solo buffer di uscita sia connesso sul bus. Quando un buffer di uscita è disabilitato viene posto ad alta impedenza. Qualora su un bus tutti i buffer

¹⁶ Frequentemente in ambito ingegneristico per molto minore o molto maggiore si intende un ordine di grandezza ovvero dieci volte minore o maggiore.

di uscita dei transceiver sono posti ad alta impedenza, ovvero solo buffer d'ingresso sono collegati sul bus, si potrebbero avere livelli di tensione derivanti dall'accoppiamento della linea con segnali esterni (normalmente accoppiamento indotto, ovvero dovuto a radiazione elettromagnetiche). I livelli di tensione derivanti dal rumore esterno potrebbero creare errori di comunicazione derivanti da una erronea interpretazione degli stessi come informazione trasmessa. Per tale ragione, al fine di garantire un livello logico noto sul bus, tale da mantenersi anche nel caso in cui tutti i driver dovessero essere in ricezione, si utilizzano i resistori di polarizzazione. Nella scheda PJ3007 questi resistori sono rappresentati da R59 e R60. La loro presenza permette di far circolare sui resistori di terminazione una corrente tale da mantenere un livello di tensione costante dell'ordine di 200mV. Da quanto detto si capisce che solo una scheda deve avere i resistori di polarizzazione poiché il loro scopo è solo far circolare della corrente sui resistori di terminazione.

Per determinare i valore dei resistori di polarizzazione bisogna calcolare la resistenza totale presente sul bus. Considerando che le resistenze di terminazione siano presenti si ha che la resistenza equivalente derivante da quest'ultime è pari a 60ohm (si sono supposte resistenze di terminazione da 120 ohm). Oltre a questa resistenza bisogna considerare la resistenza equivalente derivante dai buffer d'ingresso collegati sul bus. Da questo si capisce che il numero di nodi influisce nel calcolo delle resistenze di polarizzazione. Per il MAX485 preso nell'esempio precedente si ha che la resistenza minima dichiarata nel datasheet per il buffer d'ingresso è pari a 12Kohm. In applicazioni come CNC si possono avere fino a 4 schede PJ3007 (uno per asse) collegate sullo stesso bus. In aggiunta si possono avere altre schede di controllo 2-3 sempre collegate sul bus, per un totale di 7 schede. Facendo il parallelo tra le impedenze d'ingresso si ha che la resistenza equivalente è pari a circa 1.7 Kohm molto più grande dei 60ohm delle resistenze di terminazione per cui il parallelo tra 1.7 Kohm e 60ohm è circa 60ohm. Usando una formula per riassumere quanto detto con mille parole si ha che la resistenza equivalente sul bus RS485 è pari a:

$$R_{eq} = \frac{\frac{R_{TER} \cdot R_i}{2} \cdot n}{\frac{R_{TER}}{2} + \frac{R_i}{n}}$$

dove:

R_{eq} : Resistenza equivalente sul bus.

R_{TER} : Resistenza di terminazione.

R_i : Impedenza d'ingresso del transceiver.

n: Numero di transceiver sul bus.

nel caso in cui non si faccia uso delle resistenze di terminazione la formula si semplifica in:

$$R_{eq} = \frac{R_i}{n}$$

si fa notare che l'impedenza di uscita del transceiver, essendo ad alta impedenza (dell'ordine del Mohm) non interviene, in generale, nel calcolo della resistenza equivalente.

A questo punto al fine di avere 200mV su R_{eq} , si ha che la corrente che deve circolare su R_{eq} è:

$$I_p = \frac{200mV}{R_{eq}} = \frac{200mV}{60} = 3.3mA$$

dove:

I_p: Corrente di polarizzazione.

R_{eq}: Resistenza equivalente sul bus.

Avendo R59 e R60 tra massa e Vcc (+5V) si ha che per far circolare una corrente I_p le resistenze di polarizzazione devono essere pari a:

$$R_p = \frac{1}{2} \cdot \left(\frac{V_{cc}}{I_p} - R_{eq} \right)$$

dove:

R_p: Resistenza di polarizzazione

Nel nostro esempio con R_{eq}= 60ohm si ha R_p pari a 728ohm, usando resistori standard si può usare un resistore di 750ohm. Tale valore non è critico per cui l'aggiungere o togliere qualche modulo non comporta in generale alcun problema.

Interfaccia Pulsanti

Sebbene l'interfaccia rappresentata dai pulsanti dip-switch sia piuttosto semplice, risulta particolarmente utile in varie situazioni. In Figura 5 è possibile vedere il collegamento dei dip-switch.

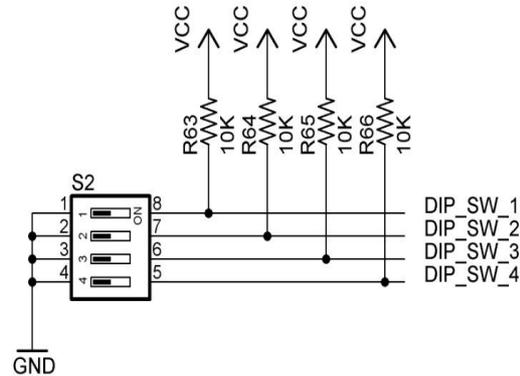


Figura 5: Pulsanti dip-switch.

Il collegamento dei dip-switch al PIC18F4431 è il seguente:

- DIP_SW1 → RD5
- DIP_SW2 → RD6
- DIP_SW3 → RD7
- DIP_SW4 → RB7

Dal momento che RB7 viene usato in fase di programmazione è necessario aprire il Jumper JP1, ovvero posizionarlo tra il pin 1 e il pin 2. Durante la programmazione deve essere aperto anche il Jumper JP2, collegato altrimenti sulla linea Emergency Stop.

I dip-switch possono tornare utili al fine di impostare il limite della corrente, in applicazioni quali la CNC o un sistema di controllo motori in cui sia necessario variare la corrente a seconda del motore utilizzato¹⁷. Qualora il sistema faccia uso dell'interfaccia RS485, tutti i parametri di sistema potrebbero essere impostati da remoto ed in particolare i dip-switch potrebbero essere utilizzati al fine d'impostare l'indirizzo della scheda sul bus.

¹⁷ Si ricorda che i motori Stepper sono normalmente pilotati da un sistema di controllo in cui viene controllata la corrente che attraversa ogni fase. Normalmente la tecnica utilizzata fa uso del PWM ed in particolare della cosiddetta tecnica chopping, che altro non è che la tecnica PWM utilizzata per i motori DC (modalità Signed Magnitude).

Interfaccia LED

Come per pulsanti, anche l'interfaccia LED è tanto semplice quanto utile. La scheda possiede cinque LED che vengono utilizzati per segnalare diversi stati della scheda e il corretto funzionamento della stessa. I LED presenti sulla scheda sono riportati in Tabella 1.

LED	Descrizione	Colore
LED1	Segnalazione presenza Vdd	Verde
LED2	Segnalazione presenza Vcc (+5V)	Verde
LED3	LED sistema HEART	Verde
LED4	LED sistema ERROR	Rosso
LED5	LED segnalazione ventilatore	Giallo

Tabella 1: Funzione dei LED di sistema.

I LED di segnalazione della tensione, seppure accesi non indicano necessariamente che il valore sia quello giusto.

I LED nominati di sistema (LED3, LED4) sono quelli pilotati direttamente dal microcontrollore e la sua funzione può variare a seconda del Firmware installato. In generale però il LED3 è pensato per segnalare che il sistema sta operando correttamente entro specifica è per questo nominato HEART, ovvero cuore. Il LED4 nominato ERROR è invece pensato per segnalare errori. All'attivazione del LED ERROR si potrebbe associare anche l'attivazione del segnale EMG_STOP_OUT che viene riportato sia sul connettore X2 che CN5. Segnali di errore, potrebbero essere inviati da remoto anche via RS485.

Errori che potrebbero essere segnalati sono:

- Corrente eccessiva assorbita dai motori.
- Temperatura di sistema elevata.
- Tensione Vdd troppo bassa o troppo alta.
- Errori di comunicazione.
- Errori software interni (es. Watchdog, BOR).

Il LED5, di colore giallo, segnala un'altra informazione utile, ovvero l'accensione del ventilatore. Il segnale che pilota il LED5 è nominato TEMP (temperatura). Il segnale TEMP viene riportato anche in uscita sul connettore CN5 al fine di permettere anche un'accensione di una ventola in maniera centralizzata. Infatti qualora siano presenti più schede è possibile avere le ventole posizionate in un unico punto al fine di garantire aria forzata all'intero sistema.

Il collegamento dei LED al microcontrollore è il seguente:

LED3 → RD0
LED4 → RD1
LED5 → RD3

Sensore di Temperatura e ventola di raffreddamento

La scheda per il controllo motori è una scheda di potenza. Sebbene sia progettata per mezzo dei ponti H L6203 ovvero ponti H in tecnologia CMOS con bassa $R_{DS(ON)}$ di soli 0.3ohm, qualora si faccia uso di alte tensioni e correnti, il ponte può scaldarsi notevolmente. Per tale ragione i ponti H come anche i regolatori lineari sono collegati ad un'aletta di raffreddamento. Sull'aletta di raffreddamento è inoltre previsto il montaggio di un eventuale ventola di raffreddamento al fine di agevolare lo scambio termico con l'ambiente circostante, e quindi il raffreddamento dell'aletta stessa. Normalmente per tensioni di circa 12V e correnti di 1A, l'aletta di raffreddamento è sufficiente a garantire che i componenti operino in condizioni ottimali. Ciononostante si ricorda che le condizioni ambientali in cui viene posto il sistema hanno notevole importanza per cui è sempre necessario valutare caso per caso la necessità o meno di aria forzata (ventilatore). Al fine di garantire un corretto funzionamento del sistema e la gestione dello stesso da un punto di vista termico è presente il sensore termico LM35 posizionato direttamente sull'aletta di raffreddamento¹⁸. In Figura 6 è riportato un dettaglio dello schema elettrico associato al sensore termico LM35.

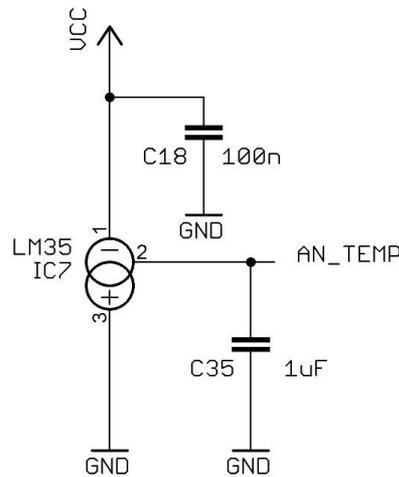


Figura 6: Schema elettrico relativo al sensore LM35

Sebbene il sensore sia stato direttamente collegato all'ADC senza far uso di operazionali o configurazioni “strane” è possibile senza problemi misurare temperature sopra 0 °C con risoluzione di 1 °C¹⁹.

Dalla misura della temperatura è possibile valutare la necessità o meno di attivare la ventola di raffreddamento o segnalare eventuali stati di warning. I ponti H hanno una protezione termica che disattiva i ponti se raggiungono una temperatura interna di 150 °C. Sebbene sia utile avere questa opzione è bene intervenire in maniera coscienziosa prima che i ponti di disattivino.

Al fine di garantire che i componenti abbiano una temperatura di funzionamento tale da non causarne un rapido deterioramento è bene lavorare a temperature molto più basse. Oltre a questo aspetto è bene considerare che essendo la scheda pensata per poter sperimentare diverse applicazioni, per cui essere spesso tenuta sul tavolo di lavoro, è bene che non si eccedano temperature di circa 60 °C.

Per esempio si potrebbe attivare la ventola intorno a 40 °C, avere una warning a 50 ° e disattivare il sistema a 60 °C.

Per poter usare temperature più elevate, sebbene i condensatori siano già di tipo alte temperature, è necessario anche che ogni altro componente sia certificato per alte temperature (resistori, diodi,

¹⁸ Nonostante il sensore sia collegato direttamente sull'aletta di raffreddamento, la temperatura segnalata è normalmente inferiore a quella dei componenti di potenza, infatti l'aletta di raffreddamento non ha una temperatura omogenea e il contatto con il sensore LM35 con l'aletta non è perfetto.

¹⁹ Trascurando tutti gli errori, dal momento che il fattore di scala dell'LM35 è di 10mV/°C, ed essendo l'ADC a 10bit (PIC18F4431), si avrebbe una risoluzione teorica anche di 0.5 °C.

transistor, circuiti integrati), cosa normalmente non fatta per le schede di sviluppo.

Come detto la lettura di temperatura permette di valutare la necessità di attivare la ventola di raffreddamento. In Figura 7 è riportato il dettaglio dello schema elettrico relativo alla ventola di raffreddamento, attivabile per mezzo della linea RD3.

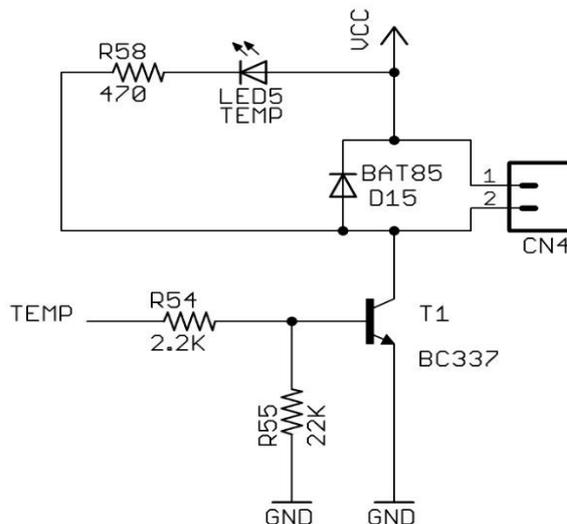


Figura 7: Schema elettrico relativo alla ventola di raffreddamento.

Come visibile dallo schema, ogni qual volta che venga attivata la ventola viene attivato anche il LED di segnalazione TEMP. Il segnale TEMP viene fornito anche in uscita sul connettore CN5 al fine di poter segnalare ad altre schede l'attivazione della ventola, o centralizzare l'attivazione di una ventola più grande²⁰. La segnalazione dell'accensione della ventola potrebbe anche essere segnalata via RS485 al fine di avere una segnalazione via hardware e software, cosa spesso necessaria per segnali associati alla sicurezza²¹.

²⁰ Si potrebbe per esempio non collegare la mini ventola direttamente sull'aletta di raffreddamento e fare in modo che venga attivata un'aletta centralizzata che possa raffreddare l'intero sistema (alimentatore o altre schede motore).

²¹ Segnali di sicurezza potrebbero essere elaborati anche solo via software ma questo richiede in generale un livello di sicurezza del software più elevato e certificati più costosi. Avere un segnale ridondante hardware permette in generale di abbassare costi associati ai test e certificazioni.

Sensore di Corrente

Il controllo di un motore è un'attività complessa. Ricordo ancora quando da bambino collegavo i motori elettrici dei giocattoli alle batterie. L'unico scopo era quello di far girare il motore. Il controllo di un motore coinvolge attività più complesse quali il monitoraggio della velocità, accelerazione, coppia, stallo. In particolare il controllo della coppia, stallo o correnti di avvio e funzionamento è possibile grazie al sensore della corrente. In Figura 8 è riportato un dettaglio della sezione relativa alla misura della corrente.

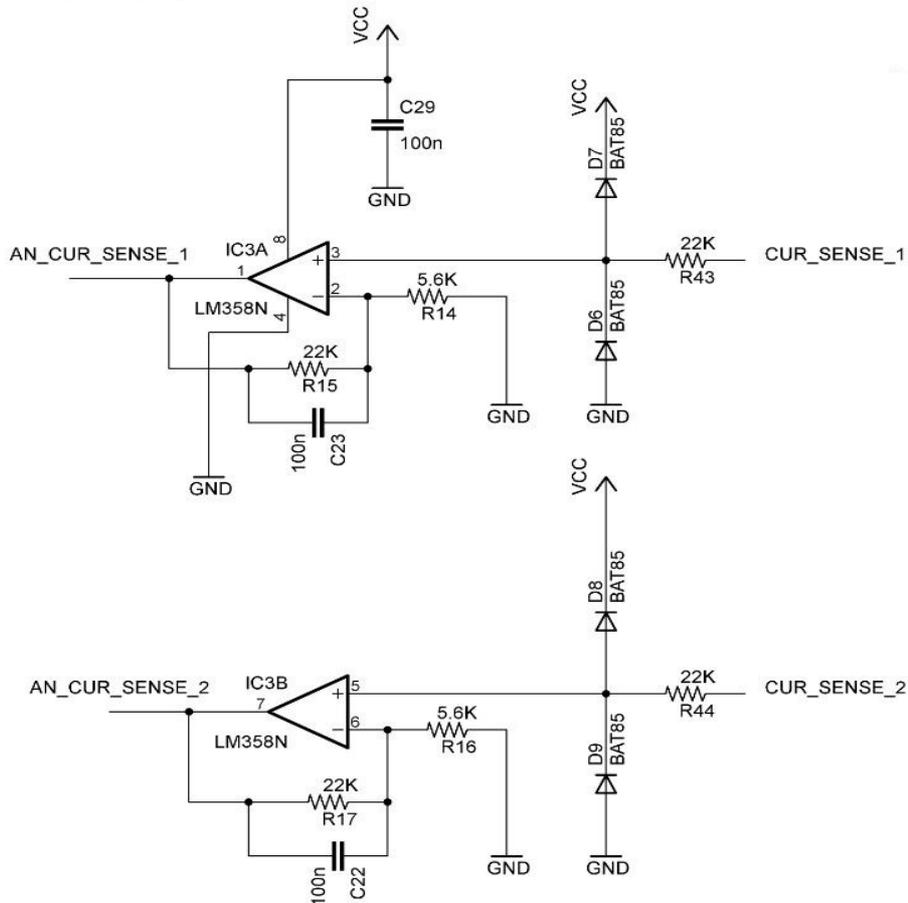


Figura 8: Schema elettrico relativo al sensore di corrente.

Il sensore di corrente è rappresentato semplicemente da un resistore da 0.1ohm. In particolare R11 è posizionato sul ponte H associato al Motore 1 o Fase 1, mentre R12 è associato al ponte H che controlla il Motore 2 o Fase 2. Tutta la corrente che scorre all'interno del motore, passa anche all'interno dei resistori appena menzionati, per cui la misura della tensione ai loro capi permette di misurare la corrente che attraversa il motore. Dal momento che la corrente è al massimo di 4A vuol dire che la caduta massima di tensione è pari a 0.4V. Visto che si ha a che fare con un ADC da 10bit²², si ha che 100mA equivarrebbero ad una d.d.p (differenza di potenziale) ai capi del resistore di 0.01V ovvero due soli quanti²³. Al fine di garantire una buona lettura anche per correnti di 100mA, la d.d.p ai capi di ogni resistore è amplificata per mezzo di un amplificatore operazionale, come riportato in

²² Considerando che la tensione di alimentazione da 5V è utilizzata come riferimento per il convertitore ADC interno al PIC, si ha che il quanto (minima quantità misurabile) è di 0.005V.

²³ L'offset massimo in ingresso dell'LM358 è di 5mV per cui è bene effettuare una calibrazione al fine di limitare i suoi effetti (per esempio leggendo la tensione in uscita all'operazionale quando i motori sono disattivi). Si ricorda che l'offset varia al variare della temperatura.

Figura 8. L'amplificatore posto per ogni resistore è simmetrico dunque spiegherò in dettaglio un solo canale, in particolare CUR_SENSE_1, ovvero la misura ai capi di R11.

La configurazione scelta per l'amplificatore è di tipo non invertente al fine di garantire un'alta impedenza del sistema di misura. Questa storia dell'impedenza è quello che viene scritto sempre sui libri, ma in questo caso non importa in realtà molto, infatti la sorgente è a bassa impedenza (0.1 ohm) per cui essere ad alta impedenza rispetto ad una sorgente da 0.1ohm è comunque piuttosto semplice e anche una configurazione invertente non avrebbe avuto problemi. Usare la configurazione invertente avrebbe portato però lo svantaggio dell'uscita invertita, per cui senza indugi ho scelto la configurazione non invertente.

La configurazione non invertente è a guadagno pari a circa 4.93 V/V (diciamo 5), per cui avendo una tensione massima di 0.4V (ovvero nel caso in cui scorrono 4A sul motore) si ha in uscita una tensione di 2V. Questo permette di ottimizzare il segnale di misura rispetto alla dinamica dell'ADC, che è pari a 5V. L'ottimizzazione potrebbe essere migliorata ma ci sono delle ragioni per cui non l'ho fatto, e come dico sempre è il progettista che deve giustificare le scelte o dare delle motivazioni nella stesura delle specifiche.

La prima ragione è che l'operazionale è alimentato a 5V, per cui la sua dinamica di uscita è limitata e pari a circa 5V-2V ovvero 3V. Si sarebbe potuto utilizzare un operazionale rail to rail, ma visto i costi ho deciso di usarne uno standard come LM358²⁴. Altra soluzione sarebbe stata quella di avere un'altra alimentazione ma questo avrebbe potuto complicare le cose aggiungendo altri componenti. Avere una tensione più alta per l'operazionale avrebbe potuto portare il rischio di rottura del PIC qualora l'uscita dell'operazionale fosse andata oltre i 5V, a meno di non proteggere l'ingresso.

Una protezione da alta tensione è stata prevista ma all'ingresso dell'operazionale. Questa protezione permette di preservare sia l'operazionale che il PIC, limitando dunque il numero di diodi di protezione²⁵.

Sebbene l'operazionale LM358 possa supportare tensioni in uscita fino a 3V, si è comunque scelto di non andare oltre i 2V, visto che per brevi periodi i ponti H possono anche permettere correnti di circa 5A, per cui rifacendo i conti si avrebbe che la tensione in uscita all'operazionale è 2.5V lasciando un margine fino a misurare correnti di 6A, che comunque dal momento che il ponte H L6203 supporta fino a 5A, 6A non devono essere raggiunti.

Chiarita la scelta dell'operazionale è bene dire due parole sul condensatore C23. La configurazione non invertente è in realtà un filtro passa basso ad un polo, con frequenza di taglio di 72Hz. Dal momento che le frequenze PWM utilizzate sono in generale di diversi KHz, e considerando che anche il motore interviene con la sua induttanza a stabilizzare la corrente, si è deciso di far uso di un solo polo. Un piccolo filtraggio aggiuntivo potrebbe essere effettuato semplicemente via software facendo una media dei campioni. Questo filtro rappresenta anche il filtro passa basso necessario prima della conversione analogico digitale, ovvero necessaria al fine di rendere "limitata" la banda del segnale in ingresso.

In ultimo è bene spendere altre due parole sul resistore R43. Tale resistore è stato dimensionato al fine di limitare la corrente nei diodi di protezione. Il caso peggiore preso in considerazione è stato il caso in cui Vdd sia direttamente applicato all'ingresso dell'operazionale, o meglio del resistore. Una situazione di questo tipo si potrebbe verificare nel caso di un corto circuito, ma anche nel caso in cui si rompa il ponte H o il resistore per la misura della corrente si danneggi risultando aperto. In questi casi, piuttosto che far bruciare il resto dei componenti la protezione interviene permettendo al PIC di poter accorgersi

²⁴ Molti amplificatori operazionali Rail to Rail hanno la piedinatura compatibile con LM358, che di fatto è uno standard per il package DIL a due operazionali, volendo si potrebbe dunque cambiare operazionale con altri rail to rail ed avere un guadagno maggiore.

²⁵ Per maggiori dettagli si faccia riferimento alla Brief Note BN0002 "Utilizzo del diodo per proteggere i sistemi elettronici".

di eventuali problemi e segnalare uno stop di emergenza e disattivare la scheda. La corrente massima del diodo BAT85 è circa 200mA, per cui considerando V_{dd} max pari a 30V si ha che la corrente nei diodi in caso di corto è circa 1.4mA.

Il resistore R43 è stato scelto da ¼ di W e la potenza che si verrebbe a dissipare sul resistore in caso di corto è di circa 0.04W, per cui il corto può essere sostenuto per lunghi periodi. Il sopportare lunghi periodi di corto vale per il sistema di protezione ma se la causa dell'attivazione del sistema di protezione dovesse essere la rottura del ponte si potrebbero avere correnti di corto piuttosto elevate.

In questo caso dovrebbe intervenire la protezione per limitare la corrente disattivando il sistema o eventualmente la rottura del fusibile²⁶.

²⁶ Si fa notare che i resistori per il monitoraggio della corrente dei motori non possono rilevare dei cortocircuiti del ponte verso massa. In queste situazioni la protezione del sistema è offerta dal fusibile. Questa caratteristica deriva dal fatto che la corrente viene misurata sul ramo “basso”. Un high side current sensing permette di rilevare un corto circuito verso massa ma la misura della tensione ai capi del resistore sul ramo “alto” è più complessa.

Segnali di controllo Servo e fine corsa

La scheda PJ3007 possiede due connettori con uscite ed ingressi ausiliari (CN2, CN3 nominati AUX1 e AUX2) ottimizzati per essere utilizzati con interruttori di fine corsa (per esempio in applicazioni quali CNC). Ogni connettore supporta anche un potenziometro pensato per poter implementare un servo, ovvero collegando un potenziometro all'asse di un motore e controllare la sua posizione leggendo la tensione ai capi dello stesso. Un dettaglio dei connettori CN2 e CN3 è riportato in Figura 9.

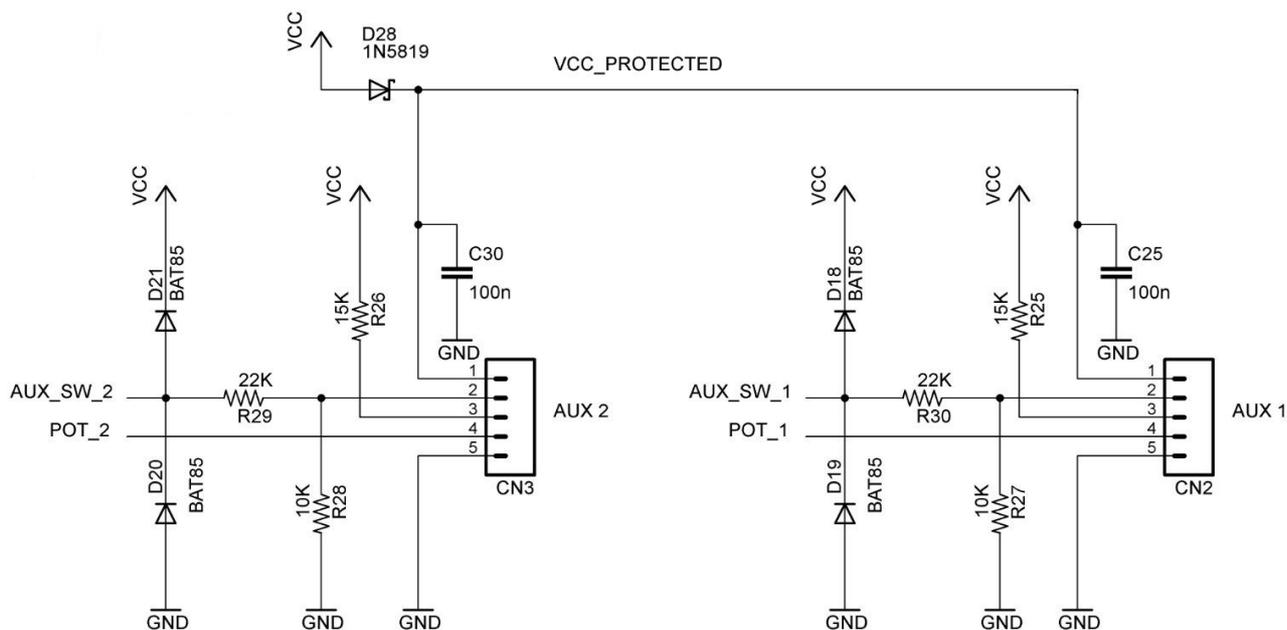


Figura 9: Schema elettrico relativo ai connettori CN2 e CN3 (AUX1-AUX2).

Si noti che i connettori possiedono rispettivamente il segnale di Ground e Vcc, al fine di poter supportare eventuali sensori esterni (vedi di seguito). In particolare il segnale Vcc viene fornito attraverso il diodo Schottky D28. Questa scelta è stata fatta per proteggere la linea Vcc da eventuali cortocircuiti con la linea Vdd a più alta tensione. Infatti se questo dovesse accadere il diodo D28 risulterebbe inversamente polarizzato. Cortocircuiti verso massa non sono direttamente protetti dal diodo ma dal regolatore lineare 7805²⁷.

Sia l'ingresso digitale assegnato all'interruttore di fine corsa che l'ingresso assegnato al potenziometro, ovvero ingresso analogico, sono protetti per mezzo di diodi di protezione. Come spiegato nel paragrafo "Sensore di Corrente" i resistori R29 e R30 sono dimensionati al fine da poter permettere di sopportare dei cortocircuiti con l'alimentazione principale (fino a Vmax) senza che i diodi vengano danneggiati ed in particolare il microcontrollore.

I pulsanti di fine corsa nominati SW1 e SW2 sono collegati rispettivamente agli ingressi del PIC RB4 e RB5. Questo permette di poter gestire tali pulsanti per mezzo delle interruzioni presenti sulla PORTB. Si noti che gli ingressi hanno i resistori di pull-down R27-R28 al fine di vincolare il loro valore a massa. Si consiglia per ragioni di sicurezza di usare interruttori normalmente chiusi. In questo modo collegando l'interruttore tra Vcc e il pin 2, questo avrà sempre il valore 1 logico a meno di non essere premuto. Il fatto di avere il valore di Vcc permette di riconoscere un eventuale cortocircuito verso massa o eventuale apertura dell'interruttore dovuta alla rottura dello stesso.

²⁷ A seconda delle applicazioni si potrebbe anche pensare di sostituire tale diodo con un resistore o toglierlo al fine di riguadagnare la caduta di potenziale ai suoi capi.

Il potenziometro deve essere collegato con un pin verso massa, il centrale collegato al pin 4 (POT_1 o POT_2) e il terzo terminale sul pin 3, come riportato in Figura 10. Questo collegamento “particolare” è necessario al fine di non saturare l'operazionale utilizzato come buffer di lettura. Se infatti il potenziometro fosse stato collegato direttamente a Vcc la tensione al suo terminale centrale potrebbe variare tra 0V e 5V. Come abbiamo visto nel paragrafo dedicato alla misura della corrente, gli operazionali, qualora non siano rail to rail non possono raggiungere in uscita il valore di alimentazione. Dal momento che il buffer utilizzato per disaccoppiare il potenziometro dal convertitore analogico digitale interno al microcontrollore è in configurazione a guadagno unitario (si veda Figura 11) il suo ingresso non invertente può accettare tensioni fino a circa 3V. Facendo uso dei resistori R25 e R26 si ha la possibilità di abbassare leggermente la tensione al piedino centrale del potenziometro. Facendo uso di un potenziometro da 10Kohm, dal momento che R25 è pari a 15Kohm, si ha che la tensione massima sul piedino centrale, applicando la regola del partitore, è pari a 2V (ovvero digitalizzato con il valore 400).

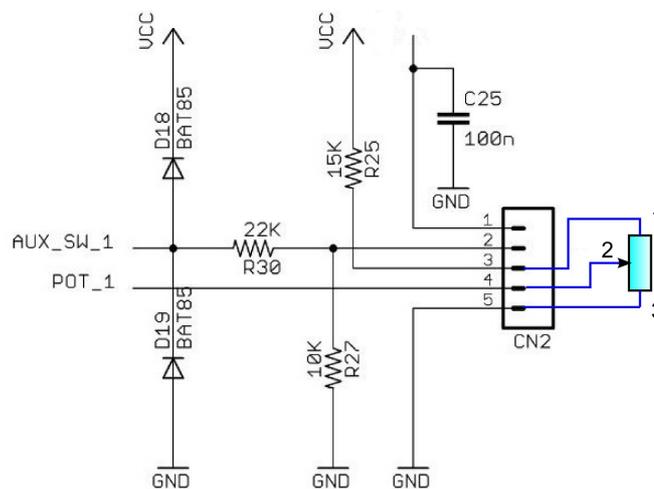


Figura 10: Schema elettrico relativo al collegamento del potenziometro.

I buffer associati ai potenziometri, come detto, sono in configurazione non invertente. In Figura 11 è riportato il dettaglio dello schema elettrico associato ai due Buffer. In particolare è possibile vedere la protezione dei diodi e dei resistori R18 e R19. In aggiunta alla protezione è presente il condensatore di filtro (C26 e C27) che permette di limitare eventuali disturbi associati sia al rumore termico dei resistori ma soprattutto il rumore che si accoppia ai cavi esterni collegati al potenziometro. Questi ingressi sarebbero potuti essere altri due buoni candidati per avere un filtro RF a T come quelli usati per il connettore CN5. In questo caso dal momento che non era presente altro spazio sul PCB, tra il proteggere il connettore CN5 associato alla CNC e quello associato al potenziometro, ho preferito proteggere le linee dei segnali assegnati per la CNC visto che il cavo potrebbe essere piuttosto lungo ed affiancato da segnali impulsivi (segnali Step, Enable, Direction).

I connettori AUX1 e AUX2 appena descritti, sebbene pensati per interruttori di fine corsa e potenziometri, potrebbero essere utilizzati per implementare un semplice robot, in particolare gli interruttori di fine corsa, essendo collegati agli ingressi con interruzioni, RB4 e RB5 possono essere utilizzati come interruttori d'urto, ovvero segnalare quando è il robot ha urtato (baffi del robot). L'ingresso analogico potrebbe essere utilizzato per leggere sensori analogici ad infrarossi, per esempio i modelli sharp che possiedono sia trasmettitore che ricevitore ad infrarossi. Questi sensori possono essere utilizzati per la misura delle distanze degli oggetti di fronte al robot. I sensori possono essere alimentati direttamente dalla tensione Vcc presente sui rispettivi connettori. Per supportare ulteriormente applicazioni quali mini robot la piedinatura dei connettori AUX è compatibile con sensori ad ultrasuoni tipo il modello SFR05.

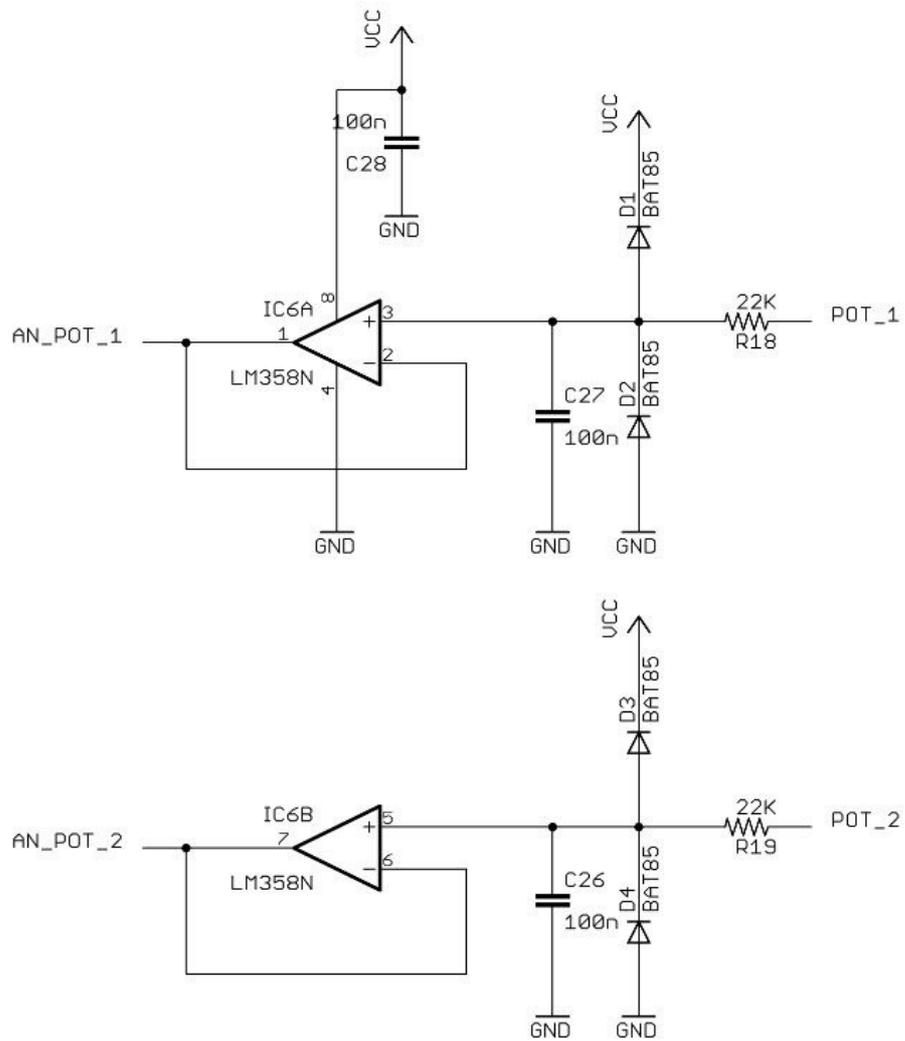


Figura 11: Schema elettrico relativo al buffer dei potenziometri.

Segnali Quadrature Encoder

La scheda PJ3007 è basata sul PIC18F4431 che come detto è ottimizzato con diverse periferiche di supporto per applicazioni quali il controllo dei motori. Tra le periferiche di rilievo vi è l'encoder in quadratura (QEI, Quadrature Encoder Interface), frequentemente utilizzato per il controllo dei motori. L'encoder in quadratura, diversamente dai semplici encoder, che inviano impulsi a frequenza variabile proporzionale alla velocità di rotazione, permettono anche di determinare il verso della stessa.

Gli encoder in quadratura sono normalmente caratterizzati da due segnali in quadratura (fasi) nominati spesso A e B, e un segnale di index che viene generato ad ogni rotazione completa dell'asse del motore.

Frequentemente le linee A, B e index degli encoder in quadratura sono di tipo open collector, dunque al fine di leggere propriamente i segnali dell'encoder sono necessari i resistori di pull-up. In Figura 12 è riportato un dettaglio dello schema elettrico associato all'interfaccia Encoder.

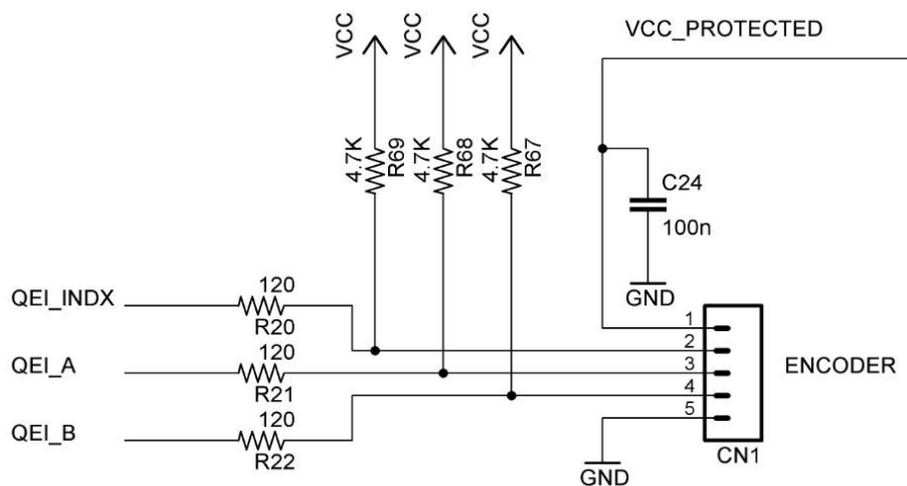


Figura 12: Schema elettrico relativo al buffer dei potenziometri.

È possibile notare che l'encoder è reso disponibile su un connettore a 5 poli con piedinatura simile ai connettori CN2 e CN3 dedicati all'implementazione dei Servo e interruttori di fine corsa. Si noti la presenza dei resistori R67, R68 e R69 utilizzati come resistori di pull-up. Normalmente non sono montati visto che alcuni encoder in quadratura possiedono già i resistori di pull-up interni. Normalmente il valore di 4.7Kohm è idoneo per molti tipi di encoder che dovessero richiedere i resistori di pull-up.

I resistori R20, R21 e R22 da 120ohm permettono di ridurre la rapidità dei transienti di tensione e corrente associati alla natura impulsiva dei segnali degli encoder.

Sebbene il connettore CN1 con l'interfaccia dell'encoder possa essere un buon candidato per avere i diodi di protezione, la mancanza di spazio sulla scheda ha portato alla scelta del non proteggere ulteriormente tali linee, infatti introdurre altri 6 diodi sulla scheda non sarebbe stato possibile.

Connettore di espansione e segnali di controllo

La scheda di controllo motori PJ3007 nonostante possieda il bus RS485, sul quale è formalmente possibile inviare una qualunque informazione, possiede anche un connettore di espansione per per ribbon cable (cavo piatto). Un dettaglio dei segnali presenti sono riportati in Figura 13.

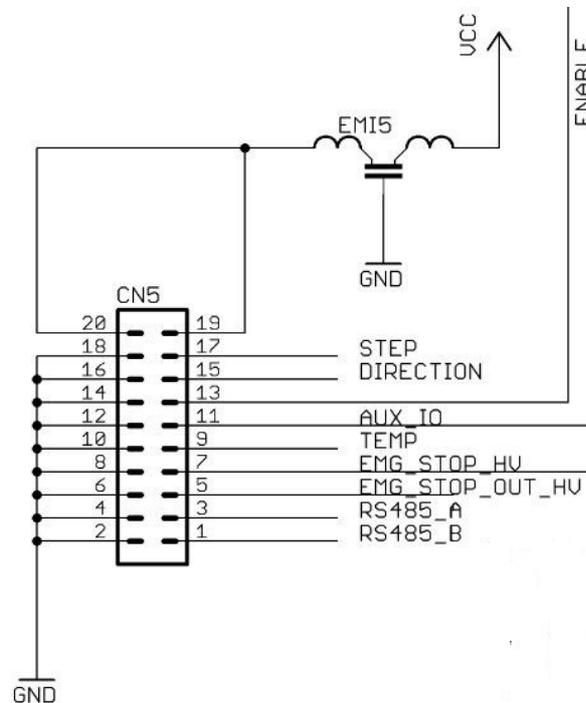


Figura 13: Connettore di espansione CN5 versione PCB 2 e successiva

E' subito possibile notare che sia Vcc che GND sono presenti. Questo significa che l'alimentazione dei 5V potrebbe essere fornita direttamente per mezzo del connettore CN5, piuttosto che usare il regolatore lineare presente sulla scheda stessa. Questo risulta utile qualora si stiano utilizzando più schede e si abbia un'alimentazione centralizzata. Come detto, in questa situazione, al fine di risparmiare effettivamente energia è necessario rimuovere il Jumper JP4.

La linea Vcc proveniente dal connettore CN5, prima di andare sulla linea Vcc della scheda possiede un filtro EMI al fine di filtrare parte di eventuali disturbi presenti sui cavi, che possono essere più o meno lunghi. Per Vcc sono state dedicate 2 linee al fine di permettere anche correnti di 200mA. La linea GND possiede invece 9 linee ed in particolare sono posizionate in maniera da essere alternate ad ogni altro segnale (esempio GND, STEP, GND, DIRECTION...).

Quando si ha a che fare con un bus, la precauzione presa nel mettere la massa (GND) in maniera alternata su ogni linea, è di particolare importanza al fine di ottenere una migliore qualità dei dati trasmessi, proteggendo ogni linea di segnale da eventuali disturbi elettromagnetici che si potrebbe accoppiare con la linea stessa.

Si osservi che ogni linea di segnale, oltre alla precauzione appena detta possiede un filtro EMI a T, ovvero formato da due induttori e un condensatore. Oltre a questo, al fine di limitare la rapidità dei fronti di salita e limitare gli effetti nocivi di radiazioni elettromagnetiche (esempio accoppiamento con altre linee di segnali), è presente un resistore da 120ohm.

I segnali presenti sul connettore CN5 sono riportati in Tabella 2. In particolare i segnali STEP, DIRECTION, ENABLE, sono i segnali standard presenti sulle schede di controllo CNC, ovvero schede di controllo per motori stepper. Il segnale AUX_IO è pensato per un uso futuro e possiede un resistore al fine di non compromettere eventuali usi futuri in cui possa essere presente un segnale di

uscita. In particolare la presenza del resistore permette anche di segnalare in maniera piuttosto semplice il tipo e il numero di schede connesse oltre al permettere un controllo indiretto della qualità della connessione del cavo .

Segnale	Descrizione	Tipo	Pin PIC
STEP	Segnale Step per motori Stepper	Input	RC3
DIRECTION	Imposta la direzione di rotazione per motori Stepper	Input	RC4
ENABLE	Segnale di enable Scheda/Motore Stepper	Input	RC5
AUX_IO	Segnale ausiliario (es. riconoscimento scheda)	I/O	n.a.
TEMP	Segnale di attivazione ventilatore	Output	RD3
EMG_STOP_HV	Segnale Emergency Stop In alta tensione	Input	RD4/RB6
EMG_STOP_OUT_HV	Segnale Emergency Stop Out alta tensione	Output	RC0
RS485_A	Linea A bus RS485	I/O	n.a.
RS485_B	Linea B bus RS485	I/O	n.a.

Tabella 2: Segnali del Connettore CN5.

Si noti che al fine di garantire una buona connessione del cavo, visto il possibile ambiente ostile in cui può essere utilizzata la scheda, è presente un connettore IDC con alette laterali. Questa precauzione è normalmente richiesta in applicazioni in cui un malfunzionamento dovuto a connessioni “fluttuanti” può causare problemi non trascurabili. Con questo non voglio infondere alcun timore ma è bene pensare che schede di potenza devono essere trattate con “rispetto”. Naturalmente questo aspetto non viene trattato in nessuna documentazione di alcun prodotto commerciale perché non conviene spaventare un cliente. Ciononostante una domanda che bisogna porsi sempre è: cosa potrebbe succedere se...?

Tornando a noi, sul connettore è presente il segnale TEMP, che come detto in precedenza potrebbe essere utilizzato per realizzare una ventilazione centralizzata. Questa risulta utile e necessaria in applicazioni con correnti e tensioni elevate, in cui l'aletta di raffreddamento e la piccola ventola di supporto non dovessero permettere il corretto smaltimento del calore generato.

In ultimo, ma non meno importante sono presenti le linee EMG_STOP_HV e EMG_STOP_OUT_HV presenti anche sul connettore X2. Queste linee di particolare importanza sono spiegate nel paragrafo dedicato ai segnali di emergenza. In questa sede è bene solo ricordare che questi segnali permettono di fermare ogni attività della scheda o segnalare gravi problemi di cui si è accorto il controllore.

Segnali di emergenza

La scheda PJ3007 è stata pensata per soddisfare esigenze di affidabilità e sicurezza. Nei paragrafi precedenti si sono visti alcuni degli aspetti e soluzioni progettuali adottati al fine di garantire una buona affidabilità del sistema. Nel seguente paragrafo si descrivono invece alcuni degli aspetti legati alla sicurezza ed in particolare al segnale nominato EMG_STOP (Emergency Stop, Stop di emergenza) pensato per bloccare il sistema sia in maniera manuale (Pulsante di stop di emergenza) che in maniera automatica.

In Figura 14 è riportato un dettaglio dei segnali di emergenza presenti sulla scheda PJ3007, in particolare il dettaglio associato al connettore CN5 in cui è presente sia il segnale di emergenza di ingresso che di uscita.

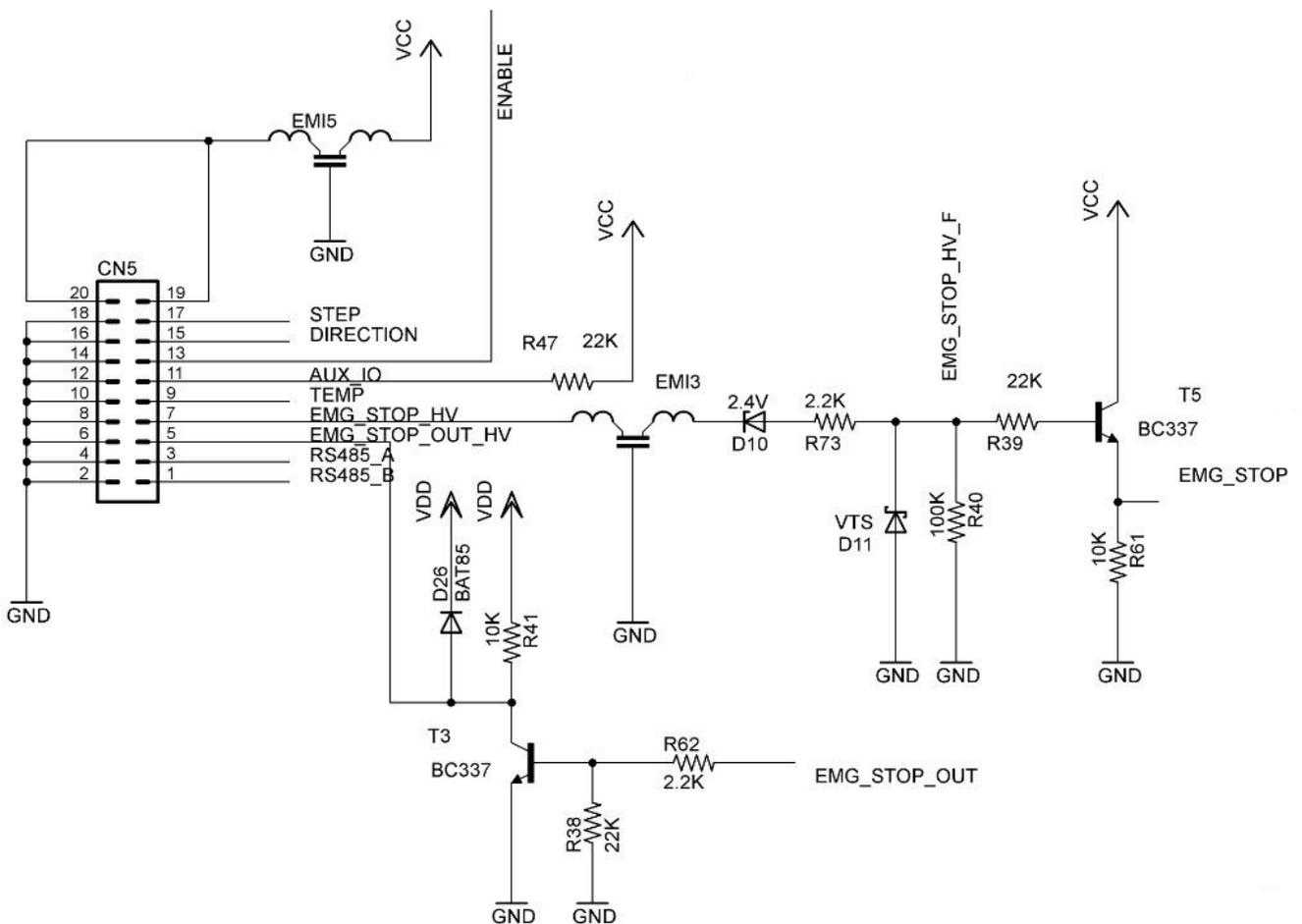


Figura 14: Segnali di emergenza

Prima di procedere nella spiegazione del funzionamento è bene descrivere le caratteristiche tecniche e funzionali delle linee di emergenza in ingresso e di uscita:

Segnale Stop di Emergenza (INPUT)

- Tensione di ingresso livello alto $V_{MIN} = 3V$, $V_{MAX} = 28V$ (valori nominali).
- Tensione di ingresso livello basso $V_{MIN} = 0V$ $V_{MAX} = 2V$ (valori nominali).
- Il segnale di emergenza è non abilitato quando la linea è a livello alto.
- La scheda risulta attiva quando il segnale di emergenza non è abilitato.

Segnale Stop di Emergenza (OUTPUT)

- Tensione di uscita livello alto $V_{MAX} = V_{dd}$ (valore nominale).
- Tensione di uscita livello basso $V_{MAX} = V_{dd} - V_{CE\ SAT\ T3}$ (valore nominale).
- Il segnale di emergenza è non abilitato quando la linea è a livello alto.
- La scheda risulta attiva quando il segnale di emergenza non è abilitato.

Uno degli aspetti più importanti per un segnale di emergenza è quello di poter essere affidabile anche in situazioni in cui il resto del sistema potrebbe avere dei problemi. Infatti un segnale di emergenza potrebbe dover essere utilizzato proprio in queste situazioni. Ci sono molti standard relativamente ai segnali di emergenza e all'affidabilità che devono raggiungere. Qualora un segnale di emergenza dovesse essere attivato solo via software richiederebbe applicazioni software e microcontrollori con elevati standard di sicurezza²⁸. Per tale ragione spesso si adotta la soluzione di avere due canali di sicurezza, uno via software e uno via "cavo". Per mezzo di questo approccio di ridondanza è possibile aumentare l'affidabilità del sistema. Nella scheda PJ3007 è stata utilizzata tale ridondanza al fine di garantire una buona affidabilità anche nel caso in cui si dovesse sviluppare un software affetto da bug²⁹.

Stop di Emergenza (linea di ingresso)

Prima di procedere oltre vediamo la nomenclatura dei segnali in ingresso al connettore CN5. Il segnale di emergenza in ingresso si chiama `EMG_STOP_HV` che sta per segnale di emergenza ad alta tensione. Per alta tensione significa solo che il suo livello può essere pari a V_{dd} , ovvero nel range 0V-28V. Tale segnale dopo essere passato per il filtro EMI viene nominato come `EMG_STOP_HV_F`, ovvero filtrato. In uscita dal transistor T5, che ha il compito di adattare il livello di tensione per il PIC18F4431 (0V-5V), il segnale viene nominato `EMG_STOP`. Ragionamento analogo si ha per il segnale di emergenza in uscita nominato prima del transistor T3 `EMG_STOP_OUT` mentre `EMG_STOP_OUT_HV` dopo il transistor T3.

Il segnale di emergenza `EMG_STOP_HV`, può essere inviato alla scheda sia tramite il connettore CN5 che il connettore X2. La differenza tra i due connettori dipende solo da come si voglia usare lo stop di emergenza. Per esempio facendo uso di un tasto rosso di emergenza è meglio usare il connettore X2 mentre nel caso di uno stop centralizzato può tornare utile usare CN5 eventualmente abbinato a X2. Il segnale `EMG_STOP_HV`, al fine di limitare falsi allarmi viene filtrato per mezzo di un filtro EMI. Successivamente il segnale passa per un diodo zener al fine di creare una soglia minima per il segnale di emergenza sotto il quale non viene rilevato. Il valore del diodo zener è stato scelto da 2.4V in maniera da poter rendere il segnale di emergenza pilotabile da segnali TTL³⁰. Si ricorda che un segnale TTL in ingresso è tale da riconoscere un livello logico 0 nell'intervallo di tensione 0V-0.8V (nominato V_{IL}) mentre un livello logico 1 nell'intervallo di tensione 2V-5.5V (nominato V_{IH}). Al fine di garantire il corretto funzionamento della porte logiche TTL sono presenti delle specifiche anche di uscita, in particolare il livello logico 0 in uscita deve essere nell'intervallo 0V-0.4V (nominato V_{OL}) mentre il livello logico 1 deve essere nell'intervallo 2.4V-5.5V (nominato V_{OH}).

Considerando i livelli di uscita a cui deve sottostare un segnale TTL si ha che l'ingresso di emergenza

²⁸ Un microcontrollore pensato per applicazioni ad alta affidabilità è per esempio l'MCU Hercules della Texas Instruments, basata su core ARM Cortex R4.

²⁹ In generale per quanti Test Cases si facciano su di un software al fine di trovare dei bug, non è possibile affermare che il software scritto non abbia bug, ma solo che i Test Cases siano eseguiti senza errori.

³⁰ Diodi zener con valore di tensione più alto potrebbero essere utilizzati al fine di aumentare la soglia di rilevamento ovvero il filtraggio. Si consiglia in ogni modo di non avere tensioni zener superiori al valore massimo della tensione utilizzata meno 5V.

potrebbe essere anche pilotato da un segnale TTL visto che il segnale verrebbe rilevato a partire da circa 3V. Ciononostante tale ingresso a meno di non cambiare il diodo zener non è 100% TTL compatibile ma dal momento che molti microcontrollori e porte logiche hanno specificato un segnale in uscita con tensioni non inferiori a 4V si capisce che la linea di emergenza potrebbe essere attivata anche da un microcontrollore. Questo risulta utile in caso in cui sia presente un sistema esterno di monitoraggio piuttosto che un pulsante manuale.

Dopo il filtraggio del segnale di emergenza è presente una protezione da extra tensioni. Questa volta la protezione è svolta per mezzo di un Voltage Transient Suppressor DC (VTS) la cui soglia di scatto è circa 39V. A questi 39V va però sommata la tensione del diodo zener da 2.4V ottenendo circa 42V. Questo significa che in ingresso, seppur il valore massimo è imposto a 28V, eventuali picchi di tensione verranno limitati a circa 42V. Il compito di R73 è quello di limitare la corrente massima nel momento in cui il diodo VTS entra in conduzione. A questo punto vi sarete chiesti per quale ragione ho usato un diodo VTS e non la configurazione a diodi precedentemente utilizzata. La ragione principale è stata quella di permettere al segnale di emergenza di assumere un valore indipendente da Vdd. Se infatti Vdd fosse di 12V mentre per il segnale di emergenza si volesse usare una tensione di 24V questo non sarebbe possibile usando la configurazione con i diodi Schottky precedentemente descritta. Infatti se i diodi fossero stati collegati a massa e Vdd, il diodo collegato a Vdd entrerebbe sempre in conduzione quando il segnale di emergenza assume il valore di 24V (superiore a Vdd).

Continuando il percorso della linea dello Stop di Emergenza si incontra R40 che rappresenta un resistore di pull-down necessario per garantire che T5 e T6 siano interdetti qualora lo stop di emergenza non sia collegato³¹. Successivamente il segnale filtrato va alla base del transistor T5 e T6 (vedi Figura 2). Il transistor T5 ha il compito di adattare il livello di tensione della linea dello Stop di Emergenza al PIC18F4431 mentre il transistor T6 ha il compito di pilotare il PMOS T4, ovvero di abilitare la scheda. Da questo si capisce che se lo stop di emergenza non venisse utilizzato ed in particolare non venisse posto ad un livello superiore a 3V la scheda non risulterebbe alimentata poiché il PMOS rimarrebbe disattivato. Questa è appunto la funzione dello Stop di Emergenza, ovvero disabilitare la scheda per mezzo del PMOS qualora venga attivato. Se non si dovesse usare lo Stop di Emergenza, al fine di attivare la scheda, è necessario con un ponticello collegare sul connettore X2, la linea EMG_STOP con il + della scheda stessa (vedi Figura 23). La linea dello stop di emergenza può essere inoltre considerata una linea di Enable per la scheda permettendo di disabilitarla e ridurre i consumi, qualora non sia utilizzata. Questa funzione potrebbe essere usata per applicazioni quali Robot alimentati a batteria.

Come detto il transistor T5 ha il compito di traslare il livello di tensione della linea dello Stop di Emergenza prima di andare sul PIC18F4431. È possibile vedere che quando la linea EMG_STOP_HV_F è attiva (normale funzionamento) il transistor T5 è in saturazione (interruttore chiuso) e dunque EMG_STOP vale 5V. Quando lo stop di emergenza viene premuto, ovvero EMG_STOP_HV_F vale 0V, il transistor è interdetto e EMG_STOP vale 0V.

L'aver collegato il resistore sull'emettitore piuttosto che sul collettore è stato fatto al fine di avere un livello logico alto di EMG_STOP e poter dunque rilevare cortocircuiti verso massa. Volendo un resistore sul collettore si sarebbe dovuto utilizzare un transistor PNP, ma dal momento che tutti gli altri sono NPN e in generale gli NPN costano meno, facendo uso di un NPN BC337 si è avuto il vantaggio di avere un solo modello di transistor (evitando errori di montaggio) e risparmiare 1 centesimo (poca cosa rispetto al vantaggio di semplificare il montaggio, almeno per piccole quantità).

La linea EMG_STOP in uscita dal transistor T5 rappresenta la seconda linea di difesa via software qualora il PMOS fosse rotto e non disattivasse la scheda. La rilevazione della linea EMG_STOP a 0 indicherebbe il problema e permetterebbe al PIC18F4431 di intervenire disabilitando i ponti H ovvero

³¹ Si noti che il resistore è posto dopo il diodo zener. Questo è necessario poiché altrimenti la sua funzione sarebbe inibita dal diodo stesso, in quanto interdetto. Lo stesso ragionamento sarebbe valido per un diodo normale (non zener).

il motore/i. Per rilevare in maniera opportuna la variazione della linea EMG_STOP si è fatto uso di linee del PIC con interrupt, in particolare si è sfruttata la linea RD4/FLTA³² presente nel PWM Power Control Module del PIC18F4431. Questa linea speciale permette di impostare le uscite del modulo PWM in condizione di Fault Condition, in particolare permette di disattivarle. La caratteristica della linea FLTA, oltre che a generare un interrupt è quella di disattivare le uscite PWM in automatico senza intervento del software.

Qualora si facesse uso del normale modulo PWM contenuto nel modulo CCP standard e non fosse dunque possibile sfruttare la linea FLTA, la linea EMG_STOP può essere deviata sulla linea RB6 per mezzo del Jumper JP2, sfruttando dunque gli interrupt sulla PORTB. In questo caso però è necessario disabilitare le uscite PWM e la linea di Enable dei ponti H all'interno della routine di gestione dell'interrupt stesso.

Da quanto appena detto si capisce che la linea Stop di Emergenza ha due percorsi distinti, in maniera da aumentare la sua affidabilità. Facendo uso di un pulsante di emergenza con doppio pulsante normalmente chiuso³³ si ha la possibilità di utilizzare un pulsante per la linea stop di emergenza e la seconda linea per alimentare la scheda. Con questo approccio il pulsante della linea di emergenza sarebbe a bassa corrente mentre il pulsante usato per portare l'alimentazione sarebbe ad elevata corrente³⁴ (maggiormente stressato). Un esempio di collegamento di un pulsante di emergenza è riportato in Figura 15.

Un'ultima nota riguardo lo stop di emergenza software è che potrebbe essere inviato anche via RS485 rendendolo totalmente indipendente dalla linea EMG_STOP.

Stop di Emergenza (linea di uscita)

Lo Stop di Emergenza di uscita è relativamente più semplice di quello di ingresso. Consiste semplicemente di un transistor pilotato dal segnale EMG_STOP_OUT, proveniente dalla linea RC0 del PIC18F4431. Come visibile in Figura 14 la resistenza R38 garantisce che il transistor T3 rimanga interdetto qualora il microcontrollore abbia dei problemi. Quando la linea RC0 è 0 logico il transistor rimane interdetto e la linea EMG_STOP_OUT_HV si trova a livello alto, ovvero a Vdd. Quando la linea RC0 vale 1 logico il transistor entra in saturazione e EMG_STOP_OUT_HV viene collegata a massa. La presenza del diodo D26 rappresenta una protezione per extra-tensioni del tipo utilizzata per i relay. Infatti il cavo che collega la linea EMG_STOP_OUT_HV ad un altro sub-sistema, potrebbe essere lungo un metro o oltre per cui rappresenta un carico induttivo. Il resistore R41 potrebbe essere rimosso ottenendo un'uscita open collector, utile qualora si voglia utilizzare una tensione differente da Vdd³⁵ o realizzare una connessione wired-or.

Lo Stop di Emergenza di Uscita potrebbe essere usato per segnalare ad altri sistemi che la scheda è stata disattivata. Potrebbe per esempio essere attivata quando il LED2 ERROR viene attivato dopo aver riscontrato problemi sulla scheda. Oltre a questi utilizzi potrebbe anche essere utilizzato per applicazioni generiche in cui si voglia pilotare un carico o segnalare un determinato stato. Anche in

³² Il PIC18F4431 possiede due pin di fault condition nominati FLTA e FLTB. La scheda PJ3007 fa uso solo della linea FLTA. In particolare la linea FLTA è disponibile sia sul pin RC1 che il pin RD4. Dal momento che RC1 viene utilizzato come segnale di ENABLE per il ponte, si è fatto uso della linea RD4. Per utilizzare la linea RD4 come segnale FLTA è necessario impostare la configuration word del PIC in maniera opportuna.

³³ Il fatto di utilizzare un pulsante normalmente chiuso permette di aumentare l'affidabilità del sistema permettendo di individuare eventuali aperture accidentali della linea (rottura del pulsante, cavo scollegato, cavo mangiato da un topo...). Il fatto di usare un livello di tensione alto come tensione di stato di emergenza disattivo permette invece di individuare eventuali corto circuiti verso massa, che sono in generale più probabili di quelli verso Vdd.

³⁴ Si capisce che il pulsante deve essere idoneo e sopportare le correnti in gioco utilizzate per il motore.

³⁵ Lasciando collegato il diodo D26 è richiesto comunque che la tensione non sia superiore a Vdd. È consigliabile aggiungere un ulteriore diodo di protezione vincolato alla tensione di uscita utilizzata.

questo caso uno “stop” della scheda, ovvero dei motori potrebbe essere segnalato via RS485.

Il segnale Stop di Emergenza di uscita, allo stesso modo del segnale di emergenza d'ingresso è disponibile sul connettore CN5 e connettore X2.

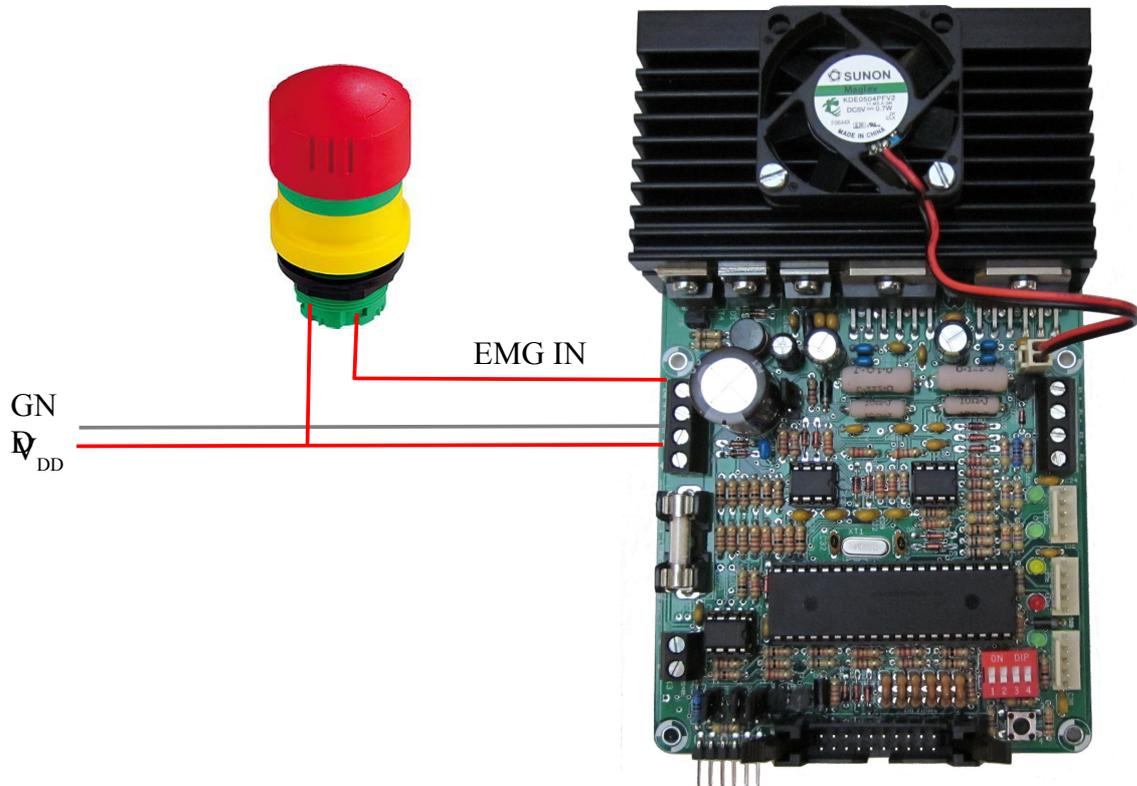


Figura 15: Collegamento Segnale di emergenza.

segnale EMG_STOP che potrebbe trovarsi vincolato a Vcc per mezzo del transistor T5.

Sebbene i resistori R71 e R72 proteggano il programmatore da correnti di cortocircuito, la programmazione potrebbe non terminare con successo qualora non si aprano i Jumper JP1 e JP2 o li si posizioni tra il pin 1 e 2. Problemi di programmazione possono essere risolti semplicemente riprogrammando il microcontrollore.

Nota:

Durante la fase di programmazione è bene non toccare la scheda. Eventuali contatti potrebbero causare errori di programmazione a richiedere una seconda programmazione.

Layout Periferiche

Durante la fase di sviluppo risulta di fondamentale importanza una guida rapida del Layout utilizzato per le varie periferiche. Sebbene questo possa essere ricavato anche dallo schema elettrico una tabella in cui vengano riportati direttamente i pin può risultare più utile. Si ricorda che a seconda delle scelte progettuali alcune parti della scheda potrebbero non essere disponibili. Ulteriore aiuto oltre alla Tabella 3 può venire dall'utilizzo dell'header file con le relative definizioni di tutti i pin e periferiche (vedi il paragrafo relativo al Software).

Pin	Nome Pin	Direzione	Nome Linea	Funzione
1	MCLR	Input	RESET	Reset della scheda.
2	RA0/AN0	Input	AN_CUR_SENSE_1	Misura della corrente del ponte H IC1.
3	RA1/AN1	Input	AN_CUR_SENSE_2	Misura della corrente del ponte H IC2.
4	RA2/INDX	Input	QEI_INDX	Segnale Index dell'encoder in quadratura.
5	RA3/QEA	Input	QEI_A	Fase A dell'encoder in quadratura.
6	RA4/QEB	Input	QEI_B	Fase B dell'encoder in quadratura.
7	RA5/AN5	Input	AN_VIN_TEST	Misura della tensione in ingresso ai ponti H.
8	RE0/AN6	Input	AN_POT_1	Misura del Potenzimetro su AUX1.
9	RE1/AN7	Input	AN_POT_2	Misura del Potenzimetro su AUX2.
10	RE2/AN8	Input	AN_TEMP	Misura della Temperatura.
11	VDD		VCC	+5V.
12	VSS		GND	0V.
13	OSC1	Input	OSC1	Connessione quarzo 10MHz.
14	OSC2	Input	OSC2	Connessione quarzo 10MHz.
15	RC0	Output	EMG_STOP_OUT	Segnale di emergenza di uscita (Out).
16	RC1	Output	ENABLE_HB_2	Enable del ponte H IC2.
17	RC2	Output	ENABLE_HB_1	Enable del ponte H IC1.
18	RC3/INT0	Input	STEP_F	Segnale di Step filtrato, proveniente dal connettore CN5.
19	RD0	Output	LED_1	LED Verde Heart (segnala normale operatività).
20	RD1	Output	LED_2	LED Rosso Error (Segnala la presenza di errori di sistema).
21	RD2	Output	RS485_OUT_ENABLE	Enable RS485 Driver
22	RD3	Output	TEMP	Segnale Temperatura alta. Il segnale attiva LED5 e ventola.
23	RC4/INT1	Output	DIRECTION_F	Segnale Direzione filtrato, proveniente dal connettore CN5.
24	RC5/INT2	Output	ENABLE_F	Segnale Enable filtrato, proveniente dal connettore CN5.
25	RC6/TX	Output	RS485_TX	Linea Trasmissione dati RS485.
26	RC7/RX	Input	RS485_RX	Linea Ricezione dati RS485.
27	RD4/FLTA	Input	EMG_STOP	Segnale di emergenza d'ingresso.
28	RD5	Input	DIP_SW_1	Dip-switch linea 1.
29	RD6	Input	DIP_SW_2	Dip-switch linea 2.
30	RD7	Input	DIP_SW_3	Dip-switch linea 3.
31	VSS		GND	0V
32	VDD		VCC	+5V
33	RB0/PWM0	Output	IN1_HB_1	Segnale IN1 del ponte H IC1
34	RB1/PWM1	Output	IN2_HB_1	Segnale IN2 del ponte H IC1
35	RB2/PWM2	Output	IN1_HB_2	Segnale IN1 del ponte H IC2
36	RB3/PWM3	Output	IN2_HB_2	Segnale IN2 del ponte H IC2
37	RB4/	Input	AUX_SW_1	Segnale pulsante di fine corsa AUX1
38	RB5	Input	AUX_SW_2	Segnale pulsante di fine corsa AUX2
39	RB6	Input	EMG_STOP	Linea alternativa per EMG_STOP
40	RB7	Input	DIP_SW_4	Dip-switch linea 4.

Tabella 3: Tabella riassuntiva delle connessioni tra il PIC18F4431 e le periferiche di sistema. Per una visione completa delle funzionalità di ogni pin si rimanda al datasheet del PIC18F4431. In Tabella sono riportate solo le funzioni principali dei pin.

Istruzioni per il montaggio

La scheda per il controllo dei motori PJ3007 è un sistema realizzato su PCB a doppia faccia. Il PCB può essere richiesto alla sezione servizi del sito www.LaurTec.it. Il suo assemblaggio non risulta particolarmente complicato ma necessita certamente di attenzione.

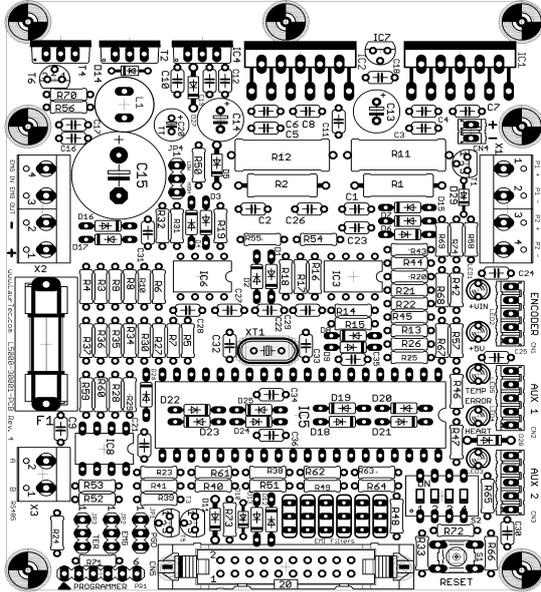


Figura 18: Serigrafia della scheda PJ3007

Per semplificare il montaggio, il PCB è realizzato con serigrafia dei componenti e relativo nome. Lo schema di montaggio è riportato in Figura 18. Il PCB reale del sistema PJ3007 è riportato in Figura 19.

Per il montaggio dei componenti è consigliabile seguire la regola legata all'altezza dei componenti stessi; dunque è bene iniziare dai resistori per poi passare ai diodi. Questa regola ha solamente un'utilità pratica associata al fatto che frequentemente, per fare le saldature, il PCB verrà posto sotto sopra. Per i primi componenti è bene accertarsi che il codice dei colori sia corretto mentre per i secondi è necessario che il verso d'inserimento sia rispettato secondo la serigrafia riportata in Figura 18. In particolare i diodi hanno un anello colorato che segnala la posizione del catodo, questo anello è riportato anche sulla serigrafia³⁷. I diodi zener oltre all'anello del catodo sono caratterizzati da una

tensione operativa diversa, anch'essa stampata sul contenitore. Successivamente si può procedere al montaggio delle capacità di filtro da 0.1uF e delle capacità ceramiche. Si ricorda che l'ordine fin qui proposto non è obbligatorio ma può risultare pratico per il montaggio.

I circuiti integrati è bene montarli su appositi zocchetti in modo da poterli cambiare in possibili aggiornamenti della scheda ed evitare comunque il loro danneggiamento in fase di saldatura. Gli integrati sono infatti sensibili alla temperatura, come d'altronde tutti i semiconduttori. Un'esposizione ad alte temperature può infatti portare alla rottura dell'integrato.

Ulteriore accorgimento va riservato alle capacità polarizzate, per le quali bisogna rispettare il verso legato alla polarizzazione. Sulla serigrafia del PCB di Figura 18 è facilmente visibile il terminale positivo delle capacità indicato con un +. Se sul PCB non si dovesse ben leggere qualche carattere a causa di via³⁸ far sempre riferimento alla Figura 18.

A montaggio completato la scheda apparirà simile a Figura 20 (si possono avere differenze da

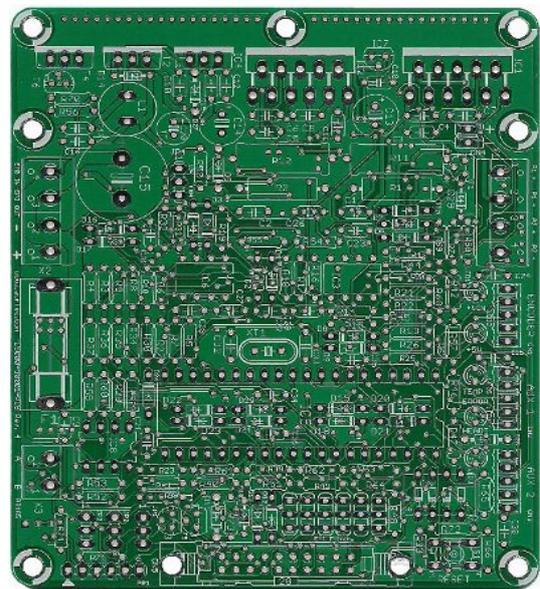


Figura 19: PCB PJ3007

³⁷ I diodi LED hanno il catodo segnalato da una smussatura sulla capsula del diodo stesso. Un altro modo per individuare l'anodo e il catodo è controllare la lunghezza dei terminali. L'anodo risulta il terminale più lungo.

³⁸ I via rappresentano i fori metallizzati che permettono la realizzazione dei PCB a doppia faccia. Il loro uso permette infatti ad un segnale di poter passare da un lato all'altro del PCB.

montaggio a montaggio a seconda dei componenti che si vuole montare o dalla versione del circuito stampato stesso³⁹).



Figura 20: Scheda controllo motori a montaggio ultimato.

Le regole sopra riportate sono generiche, altri dettagli sul montaggio dei KIT elettronici possono essere trovati nel sito www.LaurTec.it nel Tutorial “Strumenti e Tecniche per il montaggio di KIT elettronici”.

Oltre alle regole d'oro per il montaggio del KIT, avendo personalmente già montato diverse schede per il controllo dei motori è bene aggiungere qualche nota. Infatti la presenza di numerosi componenti comporta in generale maggiori accorgimenti e attenzioni.

- **Montaggio dei diodi**

Sono presenti diversi diodi BAT85, il cui contenitore, tolta la scritta sul contenitore è praticamente uguale ai diodi zener. Per tale ragione montare prima i diodi zener. Successivamente, è bene montare i diodi 1N4004 e 1N5819 visto il loro package simile. In questo modo tutti i diodi mancanti saranno i BAT85 e il VTS, dunque potranno essere montati con minor preoccupazione di sbagliare il posto.

- **Resistori**

Sono presenti diversi resistori da 120ohm, ma R24 è da 0.6W (anelli marrone, rosso, nero, nero, marrone su sfondo blu), seppur delle stesse dimensioni degli altri resistori da ¼ W. Tale resistore rappresenta la resistenza di terminazione utilizzata nel bus RS485. Anche il resistore R74 utilizzato per pilotare il LED1 è di colore blu (rosso, rosso, nero, marrone, marrone) ed è bene montarlo subito dopo. Gli altri resistori di potenza R1-R2 e R11-R12 sono piuttosto grandi per cui è difficile sbagliarsi nel loro montaggio. I resistori di potenza hanno inoltre il valore ohmico scritto chiaramente sul contenitore stesso. Montati i resistori blu è possibile procedere con il montaggio degli altri resistori.

- **Condensatori**

La scheda PJ3007 per la natura del sistema stesso, possiede diversi condensatori di disaccoppiamento al fine di garantire un ambiente non troppo rumoroso. Gran parte dei condensatori sono da 100nF, ma con stesso contenitore sono anche presenti altri condensatori. Per tale ragione prima di montare i condensatori da 100nF è bene montare i condensatori C1, C2 da 22nF, i condensatori C3-C6 da 15nF, i condensatori C7-C8 da 220nF, i condensatori C16, C17, C31, C35 da 1uF e i condensatori C34, C36 da 470nF. Dopo aver montato questi

³⁹ Il circuito stampato può subire variazioni senza che il seguente articolo venga aggiornato.

condensatori si limiteranno le possibilità di montare i condensatori da 100nF nel posto errato. Al fine di rendere più facile il montaggio dei condensatori, in Figura 21 è riportato un dettaglio del montaggio dei condensatori con valore diverso da 100nF.

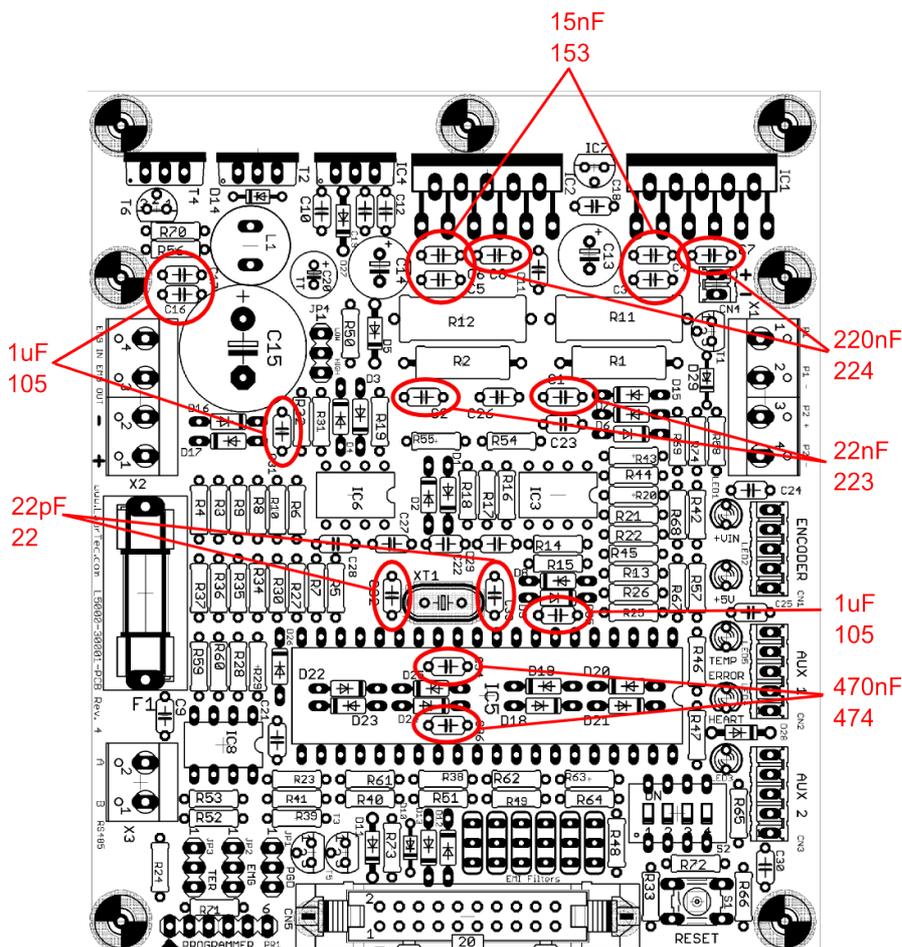


Figura 21: Montaggio dei condensatori diversi da 100nF.

• Montaggio dell'aletta di raffreddamento

La presenza dell'aletta di raffreddamento richiede qualche accorgimento, al fine di permetterne un agevole montaggio della stessa.

Prima di tutto non montare i componenti ad alto profilo presenti in prossimità dell'aletta di raffreddamento, in particolare L1, C13, C14, C15, C20. Questi componenti dovrebbero essere montati solo dopo aver fissato l'aletta di raffreddamento e i componenti con package TOxxx.

I transistor, regolatore 7805 e i ponti H devono essere montati solo dopo aver fissato l'aletta di raffreddamento al PCB.

Per fissare l'aletta di raffreddamento, allineare quest'ultima come in Figura 20, porre gli anelli neri isolatori (usati in questo caso per allineare la vite con il foro)⁴⁰. Dopo aver fissato l'aletta di raffreddamento è possibile procedere con il montaggio di T4, T2, IC4, IC2, IC7, IC1.

⁴⁰ Sono presenti solo due lunghezze di viti. Le due viti lunghe sono utilizzate per il fissaggio della ventola di raffreddamento, mentre le viti corte sono usate per il fissaggio dell'aletta di raffreddamento al PCB e il fissaggio dei componenti con contenitore TOxxx.

In particolare per ogni componente (eccetto IC7 con contenitore TO92) è presente una vite di fissaggio e un anello di plastica per allineare la vite stessa.

I transistor T4 e T2, per la natura dei componenti, richiedono la piastrina isolante da porre sotto il package stesso, al fine di evitare che si possano verificare cortocircuiti (la parte metallica non è infatti posta a massa come per gli altri componenti collegati all'aletta di raffreddamento). In particolare in questo caso l'anello d'isolamento, viene utilizzato effettivamente per isolare la vite, oltre che allineare la stessa.

Il montaggio dei componenti con contenitore TOxxx deve essere fatto affiancando lo stesso all'aletta di raffreddamento accertandosi di porre il foro allineato con la linea filettata dell'aletta di raffreddamento⁴¹. I componenti con contenitore TOxxx devono essere propriamente fissati per mezzo di una vite prima di procedere alla saldatura degli stessi. Solo in questo modo si può infatti essere certi che il componente verrà propriamente allineato senza problemi.

Una volta fissati tutti i componenti è possibile procedere con il montaggio dei componenti ad alto profilo precedentemente esclusi dalla saldatura.

⁴¹ Se il foro è ben allineato la vite si avvita facilmente altrimenti si possono avere difficoltà e avvitare le viti storte, rovinando la filettatura.

Collaudo e messa in funzione

In sistemi complessi come la scheda PJ3007, prima di mettere in funzione il sistema è bene far riferimento alle regole base per la messa in funzione di un kit assemblato in casa. Alcuni dettagli sul montaggio dei KIT elettronici e la loro messa in funzione possono essere trovati nel sito www.LaurTec.it nel Tutorial “Strumenti e Tecniche per il montaggio di KIT elettronici”.

Prima messa in funzione dopo il montaggio

1. Non montare nessun integrato prima dei test preliminari.
2. Controllare con il tester, prima di collegare la scheda all'alimentatore, che il terminale + e – non siano in corto.
3. Collegare l'alimentazione con tensione 12V accertandosi che il Jumper JP4 sia tra il pin 1-2 (ovvero posizione per alimentazioni a bassa tensione). Misurando la corrente con un amperometro si deve misurare 0mA e tutti i LED devono essere spenti.
4. Attivare la scheda mettendo un ponticello tra il + e il terminale d'ingresso EMG_IN. Il LED1 e LED2 devono accendersi. Misurando nuovamente la corrente del sistema si devono avere consumi di circa 20mA (± 5 mA).
5. Rimuovere il fusibile. Il LED1 si deve spegnere e il LED2 deve rimanere acceso (ricollegare poi il fusibile).
6. Staccare l'alimentazione e collegare gli integrati nei rispettivi zoccoli, facendo attenzione al verso d'inserimento.
7. Caricare il programma di Test nel PIC18F4431. Collegare due motori DC da 12V sulle rispettive uscite della scheda (Il primo motore su P1+ e P1-, e il secondo motore su P2+ e P2-). Verificare le seguenti funzioni:
 - All'accensione della scheda il LED Heart ed Error lampeggiano in maniera alternata.
 - La ventola di raffreddamento si accende per poi spegnersi per pochi secondi (il LED5 TEMP si accende assieme alla ventola).
 - I motori girando uno in verso orario e uno in verso antiorario.

Nota

Non precedere al passo successivo qualora un test fallisca. In particolare se il punto 2 mostra un corto, bisogna cercare e rimuovere eventuali cortocircuiti derivanti da saldature. Se il punto 4 mostra correnti superiori a quella nominale potrebbe essere ancora indice di una saldatura mal fatta che mette in contatto due punti che non rappresentano però un cortocircuito tra le alimentazioni.

Messa in funzione ordinaria

- Ogni volta che si dovesse attivare la scheda per applicazioni generiche e software applicativi differenti bisogna sempre accertarsi che l'alimentazione e i carichi (Motori) siano propriamente collegati e non vi siano fili volanti che possano causare dei cortocircuiti.
- Il Jumper JP4 deve essere propriamente posizionato. Deve essere su 1-2 (Low) per tensioni inferiori a 24V mentre deve essere sulla posizione 2-3 (High) per tensioni superiori⁴². In Figura 23 è riportato un collegamento di due motori DC mentre in Figura 22 è riportato un collegamento di un motore passo passo.

⁴² Il valore della tensione non è critico. Per esempio non spostare il ponte a 25V sulla posizione 2-3 non causa alcun danno.

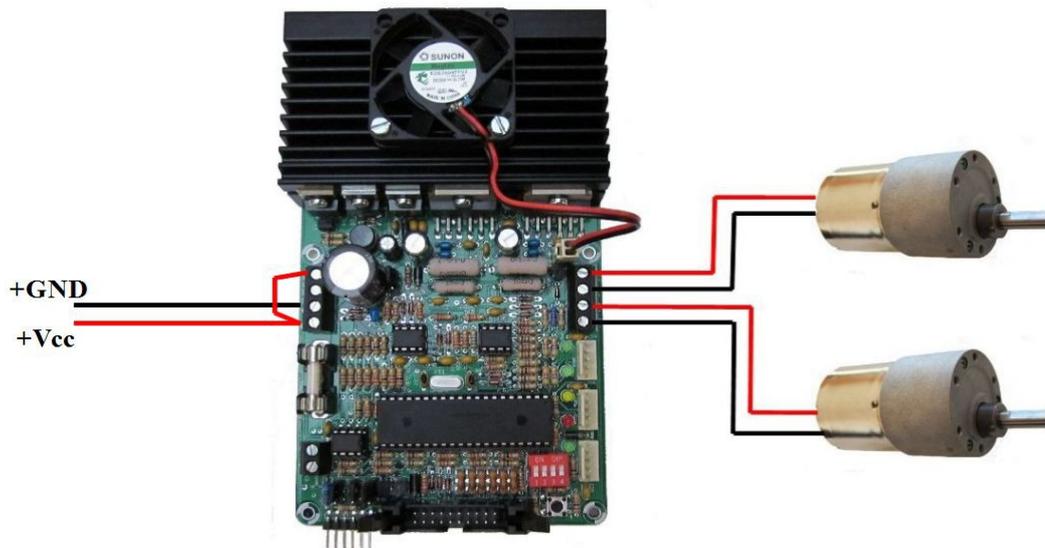


Figura 23: Esempio di collegamento di due motori DC.

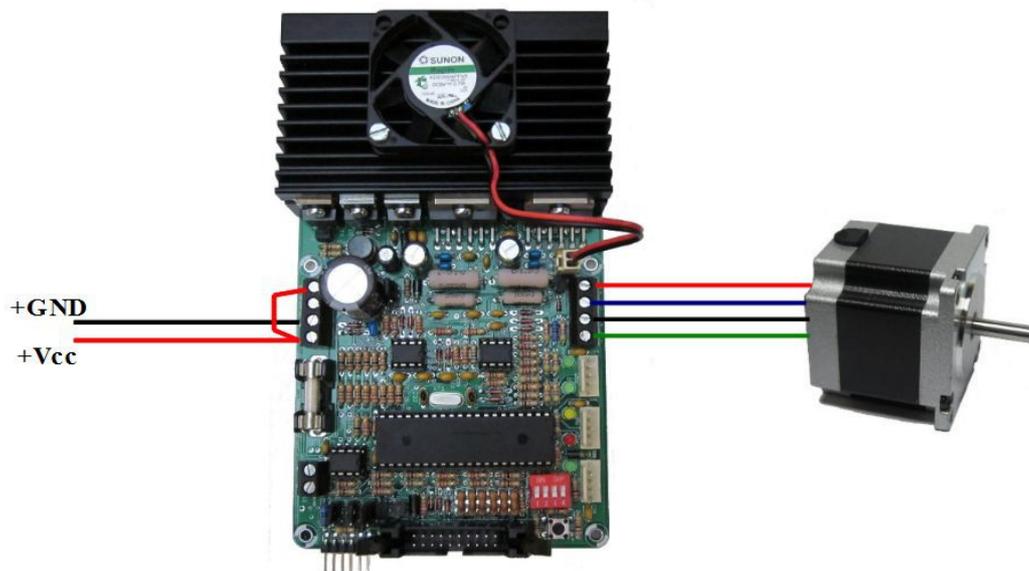


Figura 22: Esempio di collegamento di un motore passo passo (colore dei fili usati dalla Trinamics).

La scheda PJ3007 ha le stesse dimensioni fisiche della scheda Freedom II. Le due schede possono essere impilate per mezzo della scheda di espansione PJ7010, ovvero la scheda di espansione verticale di Freedom II. Un dettaglio di un montaggio è riportato in Figura 24. La necessità della scheda di espansione PJ7010 risiede nel fatto che i fori degli angoli della scheda PJ3007 sono usati dall'aletta di raffreddamento per cui è necessario utilizzare i fori adiacenti alla stessa. I fori secondari in prossimità dell'aletta di raffreddamento possono essere utilizzati anche per impilare più schede PJ3007⁴³ oltre alla scheda Freedom II.

⁴³ In applicazioni di potenza in cui sia necessario utilizzare l'aria forzata, non è consigliato impilare le schede a meno di non permettere una corretta areazione delle stesse.

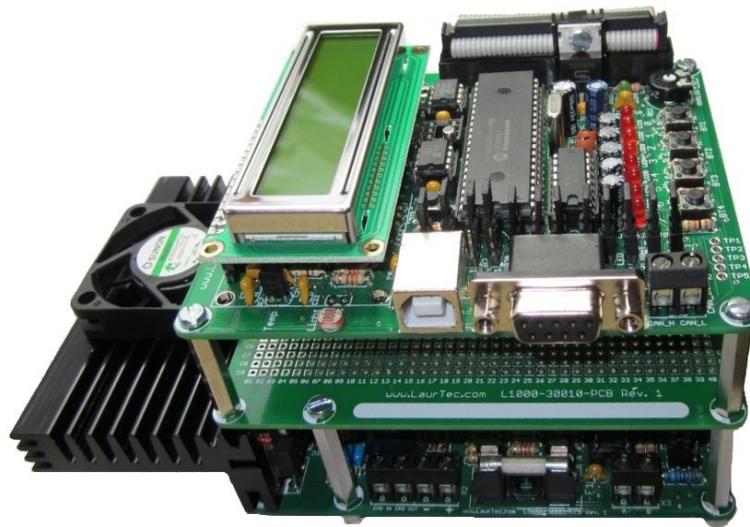


Figura 24: Esempio di montaggio verticale Freedom II e la scheda per il controllo dei motori.

La possibilità di impilare la scheda Freedom II alla scheda PJ3007 permette di realizzare sistemi con interfaccia grafica e aggiungere tutte le periferiche presenti su Freedom II. Applicazioni che possono giovare da una tale configurazione sono per esempio mini robot. La scheda Freedom II e PJ3007 possono comunicare per mezzo del bus RS485 o altri segnali. Le interconnessioni possono essere effettuate direttamente sulla scheda millefori PJ7010 posta tra Freedom II e la scheda PJ3007.

Software ed esempi

Poiché la scheda PJ3007 permette lo sviluppo di molte applicazioni, si hanno alcune complessità in più rispetto a semplici schede per il controllo dei motori, pertanto sono richieste maggiori conoscenze per lo sviluppo delle applicazioni stesse. Ciononostante, grazie agli esempi base resi disponibili sul sito www.LaurTec.it è possibile realizzare la propria applicazione senza dover perdere troppo tempo. La libreria PJ3007 composta dal file header `init.h` e `init.c` definiscono ed implementano le funzioni base per inizializzare ed utilizzare le periferiche di sistema. L'utilizzo di tali funzioni permette di avere un programma pronto all'uso semplicemente creando un main in cui vengono richiamate le funzioni. Per esempio la funzione:

```
char board_initialization (void);
```

Include l'inizializzazione di tutti i pin del PIC18F4431. Ogni linea del PIC18F4431, come visibile in Figura 1, ha un nome esplicativo della sua funzione. Lo stesso nome utilizzato nello schema è definito all'interno del file `init.h`.

Per esempio il LED_1 è definito:

```
#define LED_1 LATDbits.LATD0
```

e allo stesso tempo si è definito:

```
#define LED_ON 1  
#define LED_OFF 0
```

per cui per accendere il LED_1 basta scrivere:

```
LED_1 = LED_ON;
```

Senza dover pensare dove sia collegato effettivamente il LED. Aprendo i main dei vari esempi disponibili si possono facilmente leggere le varie funzioni ed impostazioni del programma stesso. Si consiglia di leggere il file header per verificare le altre costanti definite a supporto della programmazione. La libreria PJ3007 allo stato attuale non è ottimizzata in termini di performance bensì in termini di semplicità, in particolare gli esempi non fanno uso di interruzioni. Questo permette una facile integrazione della libreria in un proprio progetto, basta infatti includere il file `init.h` e `ini.c` al fine di poter usare tutte le funzioni disponibili. Esempi più complessi facenti uso delle interruzioni sono comunque disponibili.

Le funzioni attualmente sviluppate permettono di realizzare semplici applicazioni sia con motori DC che motori Stepper. Qualora si debba sviluppare un'applicazione per motori DC, al fine di richiamare e compilare solo le parti della libreria effettivamente necessarie, è necessario definire la seguente costante prima dell'inclusione della libreria stessa.

```
#define DC_MOTOR
```

Non necessario specificare alcuna costante qualora si faccia uso di un motore stepper.

Qualora si faccia uso delle interruzioni è necessario definire la seguente costante in maniera da permettere il controllo del modulo PWM al loop di controllo.

```
#define INTERRUPT_CONTROL
```

La libreria include nella sua versione attuale molte funzioni che permettono di utilizzare l'intero

hardware ma senza ottimizzare l'applicazione stessa. Al fine di ottimizzare il codice è richiesto che l'utilizzatore finale, qualora voglia scrivere dei propri programmi faccia riferimento al datasheet del PIC18F4431 studiando in particolare il modulo ADC, Power Control Module, il modulo CCP e il modulo QEI. La scelta degli ingressi analogici per le varie funzioni non è casuale ma permette di ottimizzare la lettura degli stessi permettendo di sviluppare codici ottimizzati in termini di risorse temporali.

Per maggiori dettagli sulle funzioni è bene far riferimento al file `init.h`. Di seguito verranno messe in evidenza solo alcuni aspetti delle stesse e solo per quelle fondamentali. Studiando gli esempi scaricabili dal sito sarà infatti facile comprendere la modalità di utilizzo di ognuna. Maggiori dettagli si possono trovare all'interno della directory `Documentation` presente nella cartella della libreria. La documentazione è stata creata per mezzo di Doxygen ed è visualizzabile per mezzo di un normale Browser semplicemente aprendo il file `index.html`. In Figura 25 è riportato un dettaglio della documentazione.

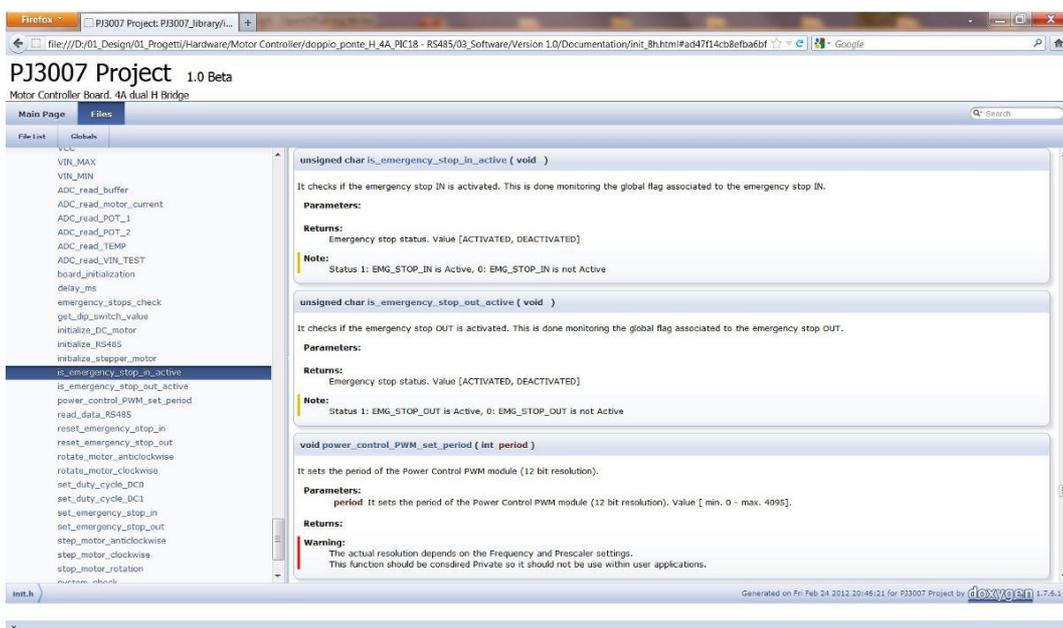


Figura 25: Esempio della documentazione realizzata per mezzo di Doxygen.

Prima di procedere oltre, faccio notare che il quarzo della scheda PJ3007 è di 10MHz e che viene attivato il PLL interno con fattore 4x, quindi la frequenza di clock del PIC è di 40MHz. Per fare questo si è fatto uso della seguente direttiva:

```
#pragma config OSC = HSPLL
```

Tale configurazione è dichiarata all'interno del file `PIC18F4431_config.h` che contiene tutte le configurazioni del PIC (cosa necessaria per evitare warning con il compilatore XC8)

Funzioni di Sistema

```
char board_initialization (void);
```

```
char system_check (void);
```

Note:

- La funzione `system_check` deve essere richiamata periodicamente al fine di garantire che vengano effettuati i controlli di sistema. In particolare che la corrente, tensione e temperatura

rientrano nei valori impostati. La corrente massima rappresenta quella impostata durante l'inizializzazione del motore.

Funzioni per Motori DC

```
char initialize_DC_motor (unsigned char modality, int pwm_frequency, int max_current );  
  
char rotate_motor_clockwise (unsigned char motor, int speed);  
  
char rotate_motor_anticlockwise (unsigned char motor, int speed);  
  
char stop_motor_rotation (unsigned char motor);
```

Note:

- Il parametro `max_current` è utilizzato dalla funzione `system_check` ed è espresso in mA.
- Il parametro `speed` non è monitorato ma rappresenta il valore desiderato. Attualmente rappresenta il valore che imposta il duty cycle del segnale PWM per cui non è espresso in giri al minuto. Il suo valore massimo dipende dalla risoluzione del PWM (vedere il datasheet per maggiori dettagli).
- La funzione `stop_motor_rotation` interviene a fermare il motore indipendentemente dal tipo (DC o Stepper bipolare).

Funzioni per Motore Stepper bipolare

```
char initialize_stepper_motor (unsigned char modality, unsigned char pwm_frequency, int max_current );  
  
char step_motor_clockwise (int current);  
  
char step_motor_anticlockwise (int current);  
  
char stop_motor_rotation (unsigned char motor);
```

Note:

- Il parametro `max_current` è utilizzato dalla funzione `system_check` ed è espresso in mA.
- Il parametro `current` non è monitorato ma rappresenta il valore desiderato. Rappresenta il valore che imposta il duty cycle del segnale PWM per cui non è espresso in mA. Il suo valore massimo dipende dalla risoluzione del PWM (vedere il datasheet per maggiori dettagli). Per un esempio di controllo della corrente fare riferimento all'esempio di controllo di un motore stepper per sistemi CNC.

Funzioni per interfaccia RS485

```
char initialize_RS485 (unsigned char baud_rate, unsigned char rx_interrupt_enable, unsigned char tx_interrupt_enable);  
  
char read_data_RS485 (unsigned char * data);  
  
char write_const_string_RS485 (const rom char *data);  
  
char write_int_ASCII_RS485 (int value, char number_of_digits);
```

Funzioni per ADC

```
int ADC_read_POT_1 (void);  
int ADC_read_POT_2 (void);  
int ADC_read_TEMP (void);  
int ADC_read_VIN_TEST (void);  
int ADC_read_motor_current (unsigned char motor);
```

Note:

- La lettura degli ingressi avviene senza sfruttare il buffer interno.

Funzioni per Emergency Stop

```
unsigned char is_emergency_stop_out_active (void);  
unsigned char is_emergency_stop_in_active (void);  
unsigned char set_emergency_stop_out (void);  
unsigned char set_emergency_stop_in (void);  
unsigned char reset_emergency_stop_out (void);  
unsigned char reset_emergency_stop_in (void);  
unsigned char emergency_stops_check (void);
```

Funzioni varie

```
char get_dip_switch_value (void);  
void delay_ms (int value);  
void turn_on_FAN (void);  
void turn_off_FAN (void);
```

La scheda PJ3007, oltre a possedere una libreria di facile utilizzo è corredata da esempi che mostrano come utilizzare la libreria stessa. Di seguito sono riportati alcuni degli esempi disponibili, i codici sorgenti e altri esempi sono scaricabili dal sito www.LaurTec.it. Applicazioni più complesse hanno articoli dedicati e possono essere scaricati alla pagina del progetto PJ3007.

A partire dalla libreria versione 1.1 è supportato solo il nuovo compilatore XC8. Gli esempi sono stati riadattati per questo compilatore.

Esempio 1

In questo esempio si mostra come inizializzare i due motori DC collegati alla scheda in modalità Signed magnitude. Dopo l'inizializzazione della scheda e dei motori, i due motori vengono rispettivamente fatti ruotare uno in senso orario e uno in senso antiorario. All'interno del ciclo while si noti la presenza della funzione `system_check ()` che garantisce che tutti i parametri di sistema si mantengano nei limiti.

```
#include <PIC18F4431_config.h>

#define DC_MOTOR

#include "init.h"
#include "init.c"

#include "delay.h"
#include "delay.c"

#define MAX_CURRENT 1000

/**
 * @note This Example shows how to initialize a DC motor in Signed magnitude Modality.
 *       After Initialization both motors are driven.
 *       One motor is rotating clockwise, while the second one anticlockwise.
 */

//*****
// Main Program
//*****

void main (void){

    board_initialization ();

    initialize_DC_motor (SIGNED_MAGNITUDE,PWM_FREQ_10KHZ, MAX_CURRENT);

    rotate_motor_clockwise (MOTOR_1, 180);

    rotate_motor_anticlockwise (MOTOR_2, 150);

    delay_ms (100);

    while (1) {

        system_check ();

    }

}
```

Esempio 2

In questo esempio si mostra come inizializzare un motore Stepper in modalità Half Step. Dopo l'inizializzazione della scheda e del motore, vengono eseguiti 1000 passi in senso orario. Dopo una breve pausa vengono eseguiti altri 1000 passi in senso antiorario, ritornando al punto di partenza. All'interno del ciclo for si noti la presenza della funzione `system_check ()` che garantisce che tutti i parametri di sistema si mantengano nei limiti. In questo esempio il tempo di esecuzione della funzione rappresenta il tempo tra uno step ed un altro.

```
#include <PIC18F4431_config.h>

#include "init.h"
#include "init.c"

#include "delay.h"
#include "delay.c"

#define NUMBER_OF_STEPS 1000
#define PHASE_CURRENT 95
#define MAX_CURRENT 5000

/**
 * @note This Example shows how to initialize a Stepper motor in half Step Modality.
 *       After Initialization the motor makes NUMBER_OF_STEPS steps clockwise.
 *       After NUMBER_OF_STEPS steps, the motor is halted keeping the position.
 *       After a little pause the motor rotates anticlockwise to the original position.
 *       This loop is repeated indefinitely.
 */

//*****
// Main Program
//*****

void main (void) {

    int i;

    board_initialization ();

    initialize_stepper_motor (HALF_STEP, PWM_FREQ_20KHZ, MAX_CURRENT);

    while (1) {

        for (i=0; i < NUMBER_OF_STEPS; i++) {
            step_motor_clockwise (PHASE_CURRENT);
            system_check ();
        }

        delay_ms (1000);

        for (i=0; i < NUMBER_OF_STEPS + 1; i++) {
            step_motor_anticlockwise (PHASE_CURRENT);
            system_check ();
        }

        delay_ms (1000);
    }
}
```

Esempio 3

In questo ultimo esempio si mostra come applicazioni medio complesse possano essere sviluppate con l'aggiunta di poche righe di codice. In particolare l'esempio permette di pilotare due motori per mezzo del PC. In particolare è possibile avviare i due motori indipendentemente, aumentare la loro velocità di rotazione e fermarli. Per eseguire questo esempio è necessario avere un convertitore USB o RS232 a RS485. Si consiglia l'utilizzo di un convertitore optoisolato al fine da garantire l'isolamento della scheda dal PC.

```
#include <PIC18F4431_config.h>

#define DC_MOTOR

#include "init.h"
#include "init.c"

#include "delay.h"
#include "delay.c"

#define MAX_CURRENT 1000

/**
 * @note This Example shows how to drive the DC motors via RS485
 *       After Initialization both motors are halted.
 *       Speed is increased pressing 1 or 2 from the keyboard.
 *       Pressing '1' increases Motor 1 speed while pressing '2' Motor 2 speed.
 *       Pressing 's' halts the motors.
 *       An USB/RS232 to RS485 is required to run the demo.
 */

//*****
// Main program
//*****

void main (void) {

    int motor_1_speed = 0;
    int motor_2_speed = 0;

    unsigned char data;

    board_initialization ();

    initialize_RS485 (BAUD_RATE_19200, RX_INTERRUPT_OFF, TX_INTERRUPT_OFF);

    initialize_DC_motor (SIGNED_MAGNITUDE, PWM_FREQ_10KHZ, MAX_CURRENT);

    while (1) {

        // Waiting for new data

        while (read_data_RS485 (&data) == ERROR_RX_DATA_NOT_READY)
            system_check ();

        ERROR_LED = LED_OFF;

        if (data == '1') {
            motor_1_speed += 10;
            rotate_motor_clockwise (MOTOR_1, motor_1_speed);
        } else

        if (data == '2') {
            motor_2_speed += 10;
            rotate_motor_clockwise (MOTOR_2, motor_2_speed);
        } else

        if (data == 's') {
```

```
        stop_motor_rotation (MOTOR_ALL);  
        motor_1_speed = 0;  
        motor_2_speed = 0;  
    } else {  
        ERROR_LED = LED_ON;  
    }  
  
}
```

Indice Alfabetico

A		funzione echo.....	19
alimentatori switching.....	13	fusibile.....	12
Alimentazione.....	4	H	
application layer.....	19	half duplex.....	21
Assorbimento.....	4	HEART.....	25
AUX_IO.....	35	HIGH.....	13
AUX1.....	32	I	
AUX2.....	32	ICD.....	42
B		IDC.....	36
baffi del robot.....	32	IN1_HB_1.....	17
baudrate.....	20	IN1_HB_2.....	17
C		IN2_HB_1.....	17
CAN.....	19	IN2_HB_2.....	17
Circuiti Integrati.....	10	index.html.....	53
CN1.....	34	Induttori.....	10
CN5.....	13, 25, 32, 35	init.c.....	52
CNC.....	6, 19, 32	init.h.....	52
condensatore di Bulk.....	13	interfaccia Encoder.....	34
Condensatori.....	9	Interfaccia LED.....	25
Connettori.....	10	Interfaccia Pulsanti.....	24
Corrente per fase.....	4	interruttore di emergenza.....	12
CUR_SENSE_1.....	29	J	
D		JP1.....	42
Dimensioni.....	4	JP2.....	40, 42
Diodi.....	9	JP4.....	13, 35
diodo zener.....	13	L	
DIP_SW_4.....	42	L'interfaccia RS485.....	19
DIRECTION.....	35	L298.....	17
Documentation.....	53	L6203.....	12, 16
Doxygen.....	53	LAP.....	11, 17
E		Layout Periferiche.....	44
Emergency Stop.....	37	LED.....	25
EMG_STOP.....	12, 37 e seg.	LED2.....	40
EMG_STOP_HV.....	36, 38	Lista Componenti.....	8
EMG_STOP_HV_F.....	12, 38	LMD1800.....	17
EMG_STOP_OUT.....	25, 38, 40	LOW.....	13
EMG_STOP_OUT_HV.....	36, 38, 40	M	
Enable.....	17	MAX485.....	20
ENABLE.....	35	modulo CCP.....	40
encoder in quadratura.....	34	MOTOR_1.....	18
Encoder in quadratura.....	11	MOTOR_2.....	18
ERROR.....	25, 40	MOTOR_ALL.....	18
ESR (Resistenza serie Parassita).....	13	MPLAB®.....	42
F		O	
Fault Condition.....	40	open collector.....	34
filtro PI.....	13	P	
filtro RF.....	32	p-greco.....	13
FLTA.....	40	PCB.....	45
Freedom II.....	50	PCB Part Number.....	4

Peso.....	4	schema di montaggio.....	45
physical layer.....	19	schema elettrico.....	6
PIC18F4431.....	11	segnale di index.....	34
PICKIT.....	42	Segnali di controllo Servo e fine corsa.....	31
PJ7010.....	50	Segnali di emergenza.....	37
PMOS.....	12	Segnali Quadrature Encoder.....	34
ponte H.....	16	Sensore di Corrente.....	28
PORTB.....	31, 40	sensori ad ultrasuoni.....	32
POT_1.....	32	serigrafia.....	45
POT_2.....	32	SFR05.....	32
Power Control Module.....	17	Signed Magnitude.....	11
Product Number.....	4	stadio di potenza.....	16
programmatori Microchip.....	42	STEP.....	35
programmazione e debug.....	42	Stop di emergenza.....	37
PSRR Power Supply Rejection Ratio.....	13	Stop di Emergenza di uscita.....	40
Pulsanti.....	10	strumenti Microchip.....	42
Q		SW1.....	31
QEI.....	11	SW2.....	31
QEI, Quadrature Encoder Interface.....	34	T	
Quarzi.....	9	T3.....	38, 40
R		T4.....	39
RB0.....	17	T5.....	38 e seg.
RB1.....	17	T6.....	39
RB2.....	17	telescopio.....	6, 19
RB3.....	17	TEMP.....	25, 27, 36
RB4.....	31	TER.....	20
RB5.....	31	Transistor.....	10
RB6.....	40, 42	U	
RB7.....	42	UART.....	19
RC0.....	40	V	
RC1.....	17	Versione PCB.....	4
RC2.....	17	via.....	45
RDS (ON).....	26	Voltage Transient Suppressor.....	39
resistenza di terminazione.....	20	VTS.....	39
Resistore di terminazione.....	20	X	
Resistori.....	8	X2.....	25, 36
Resistori di polarizzazione.....	21	Z	
robot.....	6	Z0.....	20
RS485.....	19	zener.....	13
S		zener di potenza.....	13
S2.....	42		

Bibliografia

- [1] www.LaurTec.it : sito ufficiale della scheda PJ3007 dove poter scaricare ogni aggiornamento e applicazione. Il PCB è reso disponibile alla sezione servizi sotto donazione di supporto al sito stesso.
- [2] www.microchip.com : sito dove scaricare i datasheet del PIC18F4431.
- [3] www.st.com : sito dove scaricare il datasheet del ponte H L6203 e l'Application Note AN280.

History

Data	Versione	Nome	Descrizione Cambiamento
16.11.13	1.2a	Mauro Laurenti	Inserito valore mancante R38 nella lista componenti.
24.02.13	1.2	Mauro Laurenti	Introdotta la Versione Libreria 1.1. Modificati gli esempi da C18 a XC8 Rilasciato il nuovo esempio per controllo CNC
08.07.12	1.1	Mauro Laurenti	Introdotta la Versione PCB 4. L'hardware rimane 100% compatibile. Migliorata la serigrafia a layout PCB. Ridotto il tempo per il montaggio del KIT.
04.03.12	1.0	Mauro Laurenti	Versione Originale. Versione PCB 3 1.1