



UNIVERSITÀ DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA IN INGEGNERIA ELETTRONICA

**Sistema di controllo per  
l'automatizzazione di un cambio di  
derivazione motociclistica**

*Laureando:*  
Fabiano FOSSATO

*Relatore:*  
Dott. Simone BUSO

Anno Accademico 2010/2011



# Sommario

Questo documento tratta la realizzazione del controllo dell'attuazione pneumatica di un cambio di origine motociclistica montato nella monoposto di FORMULA SAE MG05/10 del RACE-UP TEAM dell'Università degli Studi di Padova. Nel seguito vengono descritti gli argomenti trattati nei vari capitoli.

**Primo capitolo** Viene analizzato nel suo complesso tutto il sistema che si vuole controllare, sia dal punto di vista della meccanica del cambio, sia per quanto riguarda l'elettronica presente nel veicolo.

**Secondo capitolo** In questo capitolo si effettuano delle considerazioni sui vari sistemi di attuazione e di controllo che si possono utilizzare per migliorare al massimo sia il rendimento che l'affidabilità della vettura.

**Terzo capitolo** In questa parte verrà trattato il sistema hardware che è stato utilizzato: dal tipo di controllore utilizzato, al suo sistema d'alimentazione, per finire con il sistema di interfacciamento descrivendo e motivando le varie scelte.

**Quarto capitolo** Viene descritto il software utilizzato, scegliendo innanzitutto il tipo di linguaggio, poi l'algoritmo, per finire con il codice e la spiegazione di ogni istruzione.

**Quinto capitolo** In ques'ultimo capitolo viene analizzato il comportamento del sistema di controllo, descrivendone i punti di forza e quelli deboli che sono stati riscontrati durante il suo utilizzo e di conseguenza viene proposta una possibile evoluzione.



# Indice

<b>Sommario</b>	<b>iii</b>
<b>1 Analisi del sistema</b>	<b>3</b>
1.1 Descrizione generale . . . . .	3
1.2 Il motore . . . . .	3
1.3 Il cambio . . . . .	4
1.3.1 Il sincronizzatore . . . . .	5
1.3.2 La cambiata senza frizione . . . . .	6
1.3.3 Il <i>quick-shifter</i> . . . . .	6
1.3.4 Problematiche e precauzioni . . . . .	8
1.4 L'elettronica . . . . .	9
1.4.1 La centralina motore . . . . .	9
1.4.2 Il sistema di acquisizione dati . . . . .	10
<b>2 Sistema di controllo</b>	<b>11</b>
2.1 Valutazione dei vari sistemi di automazione . . . . .	11
2.1.1 Attuazione con motore elettrico . . . . .	11
2.1.2 Attuazione con sistema meccanico . . . . .	12
2.2 L'attuazione del cambio . . . . .	12
2.3 Il controllo . . . . .	14
2.3.1 La scalata . . . . .	14
<b>3 Studio del sistema hardware</b>	<b>17</b>
3.1 Gli ingressi . . . . .	17
3.2 Le uscite . . . . .	17
3.3 Il microcontrollore . . . . .	18
3.4 Il regolatore di tensione . . . . .	21
3.5 L'interfacciamento . . . . .	23
3.6 Lo schema elettrico . . . . .	26
<b>4 Studio del sistema software</b>	<b>29</b>
4.1 Il linguaggio . . . . .	29
4.2 Algoritmo . . . . .	30
4.2.1 Inizializzazione PIC . . . . .	32
4.2.2 Polling . . . . .	39
4.2.3 Esegui SHIFT_UP . . . . .	39
4.2.4 Esegui SHIFT_DOWN . . . . .	40
4.3 Il codice: . . . . .	44
4.3.1 Analisi . . . . .	49
<b>5 Considerazioni finali</b>	<b>53</b>
5.1 La scheda . . . . .	53
5.2 Una possibile evoluzione . . . . .	55
<b>A Regolamento SAE: sistema elettrico</b>	<b>61</b>

<b>B</b>	<b>Regolamento SAE: sistema frenante</b>	<b>63</b>
<b>C</b>	<b>Controllore PIC: Configurazione dei comparatori</b>	<b>65</b>

# Introduzione

La FORMULA SAE è un progetto internazionale nel quale varie università di tutto il mondo si impegnano, seguendo un regolamento, a progettare e costruire una monoposto da competizione per poi metterla alla prova in eventi organizzati in vari paesi. La particolarità di queste competizioni è che, a differenza di un normale competizione automobilistica, la classifica finale non tiene conto soltanto dell'aspetto puramente prestazionale della vettura, ma anche del costo che ha avuto, delle soluzioni tecniche adottate, della qualità generale e dei consumi. Tra le varie prove, infatti, una consiste in una relazione che, simulando la produzione di 1000 vetture, valuti i costi di produzione in relazione alla qualità della vettura e che adotti soluzioni commerciali per cercare di rendere il proprio progetto migliore agli occhi di un potenziale acquirente. Un'altra prova consiste invece in una valutazione delle soluzioni tecniche più innovative introdotte dalle squadre, con l'intento di stimolare il più possibile la fantasia degli studenti. Oltre a queste prove "teoriche" vi sono le classiche prove pratiche che valutano l'affidabilità, l'accelerazione, l'agilità e in generale l'aspetto prestazionale della vettura, sia tramite prove specifiche, sia tramite giri in pista cronometrati. Tutti questi aspetti vengono accuratamente soppesati e alla fine viene stabilita una classifica con un vincitore assoluto dell'evento, che viene chiamato così proprio per sottolineare che è differente rispetto alla classica competizione automobilistica. Per sottolineare che questi eventi non vogliono avere fini commerciali, ma unicamente educativi e formativi per gli studenti, il premio per la squadra vincitrice non è monetario, ma consiste nell'iscrizione gratuita al medesimo evento l'anno successivo.

Il progetto descritto in questa tesi si è occupato dell'automatizzazione della cambiata delle marce nella monoposto di FORMULA SAE 05/10, la quale ha partecipato agli eventi nella stagione 2010. L'obiettivo è stato quello di aumentare le prestazioni della vettura permettendo al pilota sia di effettuare le cambiata senza staccare le mani dal volante sia di velocizzare i tempi stessi di cambiata. In precedenza, infatti, tutte le monoposto del team hanno sempre utilizzato un sistema di cambiata manuale.

La possibilità di effettuare le cambiata con le mani saldamente attaccate al volante dà la possibilità al pilota non solo di poter effettuare le cambiata anche quando è necessario applicare al volante forza con entrambe le braccia, ma anche di diminuire lo sforzo fisico imposto dalla cambiata manuale e, non ultimo, di poter mantenere costante il livello di concentrazione sul comportamento della vettura. Aspetto, quest'ultimo, che nella guida a livello competitivo risulta essere tutt'altro che trascurabile.

La velocizzazione dei tempi di cambiata invece è assicurata dal fatto che la leva delle marce non sarà più comandata manualmente, ma da un sistema meccanico a controllo elettronico. Tale sistema oltre a garantire cambiata più veloci ha un rendimento costante, a differenza di quello manuale che è soggetto all'affaticamento fisico del pilota.

A questa serie di vantaggi vanno considerati anche gli svantaggi che ne possono derivare, ma fondamentalmente è soltanto uno: *il peso*. In una qualsiasi vettura infatti il peso è qualcosa che influisce molto su prestazioni e consumi, quindi, a maggior ragione, questi svantaggi vengono evidenziati nelle vetture da competizione, dove si ricerca sempre la massima prestazione. L'aggiunta di quantità anche minime di peso quindi, deve essere sempre accuratamente valutata e motivata; cosa che è stata fatta anche in questo caso. Tuttavia, l'aggravio di peso viene abbondantemente compensato da tutti i vantaggi sovracitati. Ciò tuttavia non giustifica l'utilizzo di un sistema inutilmente pesante, si è infatti cercato di utilizzare un sistema che, rispettando le prestazioni richieste, risulti essere il più leggero possibile in modo da diminuire il più possibile gli svantaggi.



# Capitolo 1

## Analisi del sistema

### 1.1 Descrizione generale

Nel mondo della FORMULA SAE con il passare degli anni e quindi con l'evolversi delle auto, delle tecnologie e delle conoscenze si sono delineate due filosofie di pensiero nella progettazione delle monoposto che, a seconda della scelta, impongono differenti approcci per quanto riguarda la realizzazione di telaio e motore. Queste due filosofie potremmo così definirle:

**Essenzialista** Le particolari caratteristiche di un'evento di FORMULA SAE, quali l'estrema tortuosità dei tracciati e le potenze dei motori contenute dal regolamento tramite una strizione nel condotto d'aspirazione hanno fatto sì che alcune squadre abbiano deciso di non utilizzare un propulsore con la massima cilindrata ammessa. Questa strada a primo impatto può sembrare una contraddizione ma, poiché il diametro massimo di questa strizione è di un valore fissato ed indipendente dalla cubatura del motore, un'unità di cilindrata inferiore potrà contare di perdere, in proporzione, meno potenza rispetto ad un'unità di cilindrata superiore. Ovviamente la cosa è relativa, quindi il motore sarà comunque meno potente rispetto a uno di cubatura maggiore, ma sarà anche più piccolo e leggero e questo può permettere la realizzazione di un telaio di dimensioni più contenute. Queste caratteristiche si ripercuotono su tutta l'auto, dando la possibilità di realizzare tutto opportunamente scalato: dall'impianto frenante a quello di raffreddamento, dalle sospensioni alla trasmissione. In sostanza, quindi, il peso può risultare molto contenuto facendo sì che il rapporto peso/potenza dell'auto risulti vicino a quello di una vettura dotata di un motore più prestante ma, dato il minore peso assoluto, di avere inerzie minori, quindi un'agilità superiore.

**Prestazionale** Questa filosofia costruttiva, invece, sfrutta appieno i limiti regolamentari di cilindrata costruendo l'auto in funzione della massima potenza assoluta che è possibile ottenere. Questo ovviamente comporta, al contrario di quanto detto prima, la necessità di dover realizzare tutte le componenti della vettura di dimensioni maggiori e quindi più pesanti. L'obbiettivo è quindi di riuscire a sopperire alla minore agilità di cui si dispone con le maggiori prestazioni assolute che è in grado di fornire un motore di maggiore cilindrata.

Le differenti correnti di pensiero sulla costruzione della vettura, dunque, ruotano attorno alle caratteristiche che ciascuna squadra decide di fornire al proprio motore. La storia della FORMULA SAE, per altro, non ha mai fatto emergere una filosofia nettamente superiore all'altra, premiando piuttosto la capacità di ciascuna università di ottimizzare al meglio la propria vettura a prescindere dalla strada che ciascun team decide di intraprendere.

### Il motore

Il RACE-UP ha da sempre seguito la filosofia prestazionale e, nello specifico, l'unità termica della MG05/10 ha, come caratteristica principale, quella di essere un quattro cilindri in linea prodotto da Kawasaki nel 2005 con cilindrata pari a  $600\text{cm}^3$ , questo per rispettare il regolamento della FORMULA SAE che impone l'utilizzo di un motore a pistoncini a 4 tempi con cilindrata

massima pari a  $610\text{cm}^3$ . La scelta di utilizzare in particolare questa unità è dovuta a varie ragioni:

- Rispetto ad altre configurazioni dei cilindri la potenza risulta essere maggiore e la massa più centralizzata;
- Il RACE-UP ha, fin dalla realizzazione della sua prima monoposto, sempre utilizzato motori Kawasaki con questa architettura per cui se ne aveva una buona conoscenza;
- Risulta essere performante e al contempo molto affidabile.

Il motore è di derivazione motociclistica e ha la caratteristica, come nella quasi totalità delle moto di concezione moderna, di avere il cambio e l'unità termica contenuti entrambi in un'unica carter<sup>1</sup>. Questa caratteristica consente di ottimizzare dimensioni e peso riducendoli al minimo e centralizzando notevolmente le masse.

## 1.2 Il cambio

Il cambio in questione è della tipologia *a innesti frontali o sequenziale*, la cui caratteristica principale è che la cambiata viene effettuata agendo su un'apposita leva di comando ruotandola in due direzioni: in senso orario o antiorario, passando al rapporto superiore o inferiore. Ciò significa che ogni qualvolta si desidera innestare una nuova marcia è necessario ruotare la leva di comando fino a quando la rotazione viene bloccata da un apposito fermo, raggiunta tale posizione il nuovo rapporto risulta inserito. A questo punto si può rilasciare la leva, che ritornerà automaticamente nella posizione neutra di partenza, il cambio risulta ora nuovamente pronto per effettuare una nuova cambiata.

Il cambio sequenziale presenta due importanti caratteristiche che nel nostro caso risulteranno molto utili:

- L'inserimento del nuovo rapporto termina una volta che la rotazione della leva di comando raggiunge il fermo, quindi la fase di ritorno della leva in posizione di neutro non è da includere nel tempo di cambiata in quanto l'albero<sup>2</sup> d'ingresso e d'uscita del cambio risultano già essere relazionati da un nuovo rapporto di rotazione.
- Si ha la possibilità di utilizzarlo anche continuando a tenere la leva di comando in posizione di finecorsa senza comprometterne né il funzionamento né l'affidabilità dello stesso, ma incorrendo nell'unico svantaggio di non poter comunque effettuare una nuova cambiata prima del ritorno della leva nella posizione neutra.

Rispetto ai tradizionali cambi per l'uso automobilistico (denominati "ad H") vi sono quindi delle differenze rilevanti. In particolare in un'automobile la leva di comando non ha soltanto la possibilità di effettuare due movimenti bensì tre, questa infatti non ha solo la possibilità di essere ruotata ma anche di essere traslata lungo il suo asse di rotazione<sup>3</sup>. Questo terzo movimento conferisce al guidatore la possibilità, aggiuntiva rispetto al cambio sequenziale, di poter innestare in qualsiasi momento il rapporto che più desidera senza dover inserire anche tutti i rapporti intermedi. Quest'ultima caratteristica a primo avviso può sembrare uno svantaggio importante per i cambi sequenziali, ma vanno fatte alcune considerazioni:

- L'utilizzo dell'automobile a livello competitivo impone di sfruttarne al massimo le prestazioni e per far questo si deve cercare di sfruttare il più possibile le caratteristiche del motore non solo in fase di accelerazione, ma anche di frenata, sfruttando il freno motore<sup>4</sup>. Questo, nella pratica, impone di dover scalare i rapporti singolarmente rilasciando ogni

<sup>1</sup>Viene definito come un coperchio con il compito di contenere e proteggere un determinato organo meccanico contenuto al suo interno es: carter motore, carter cambio, carter frizione

<sup>2</sup>L'albero è un'organo di trasmissione di un moto rotatorio

<sup>3</sup>In quasi tutte le automobili con cambio manuale il movimento di rotazione della leva di comando presente nel cambio è associato al movimento sinistra-destra della leva delle marce presente in abitacolo, mentre la traslazione dell'asse di rotazione è associata al movimento avanti-indietro

<sup>4</sup>Per freno motore si intende l'azione di rallentamento di un veicolo ottenuta scalando uno o più rapporti del cambio in modo da innalzare il numero di giri del motore; questa operazione, data l'alimentazione scarsa sfrutta le perdite meccaniche e di pompaggio per imprimere una coppia frenante alle ruote

volta la frizione; diversamente facendo, infatti, il freno motore risulta essere molto violento per il repentino innalzarsi dei giri motore, rendendolo di difficile gestione e incorrendo nel rischio di bloccare le ruote, di fare un fuorigiri<sup>5</sup> o, nel tentativo di non ritrovarsi nelle situazioni precedenti, di non sfruttarlo al meglio.

- Con la tipologia del cambio ad H si può incorrere nell'errore di inserire involontariamente un rapporto molto diverso da quello desiderato, circostanza questa che è facilitata in condizioni di guida estrema, dove si è sotto intenso stress fisico e psicologico. Quest'errore può causare, nel caso dell'inserimento di un rapporto troppo lungo, a una rilevante perdita in accelerazione o, inserendo un rapporto troppo corto, di un pericoloso fuorigiri.

Alla luce di queste considerazioni quindi, quelli che inizialmente potevano apparire svantaggi non indifferenti, risultano invece essere dei vantaggi concreti, non a caso questa tipologia di cambi viene utilizzata anche in molte competizioni automobilistiche, oltre che in tutte quelle motociclistiche; motivo per cui il cambio montato nella nostra monoposto risulta essere quello originale Kawasaki.

### 1.2.1 Il sincronizzatore

Come già detto l'innesto di un rapporto nel cambio in questione avviene tramite la rotazione della leva di comando in due direzioni, quest'operazione impone un cambiamento del rapporto di velocità tra i due alberi del cambio. Normalmente, nei moderni cambi automobilistici, il pareggiamento della velocità dei due alberi è agevolata da un componente meccanico, il sincronizzatore.

Il sincronizzatore è un'organo meccanico che consente di poter connettere tra loro due alberi che ruotano a velocità differente in modo dolce, evitando quindi la tipica "grattata". Nella pratica, immaginando che uno dei due ingranaggi da rendere solidali sia fermo e l'altro sia in movimento, il sincronizzatore si comporta come una frizione che, prima di connettere i due ingranaggi, accelera quello che è fermo fino a quando la sua velocità raggiunge il valore dettato dal nuovo rapporto di trasmissione. Questa operazione, tuttavia, generalmente non avviene accelerando gli ingranaggi, bensì gli alberi di trasmissione. Tutti gli ingranaggi montati in uno dei due alberi sono infatti sempre in presa, mentre gli altri sono normalmente in "folle", cioè montati su dei cuscinetti che vengono opportunamente "disabilitati" per consentire il trasferimento di coppia.

Il sincronizzatore offre quindi possibilità di innestare i rapporti anche se i due alberi non girano con un corretto rapporto di velocità, in mancanza di questo dispositivo il pareggiamento delle velocità dei due alberi deve essere effettuato dal pilota portando il motore, e quindi l'albero primario, al corretto regime di rotazione in funzione di quello delle ruote, ovvero dell'albero secondario. Questo dispositivo, pur presentando degli innegabili vantaggi per quanto concerne la facilità nell'effettuare la cambiata, in ambito competitivo risulta avere anche svantaggi rilevanti. Questi sono principalmente legati al tempo durante il quale il sincronizzatore pareggia la velocità dei due alberi prima che il rapporto venga effettivamente inserito, questo infatti porta ad un rallentamento del tempo effettivo di cambiata. V'è da dire inoltre che i sincronizzatori sono componenti soggetti ad usura proprio in quanto fungono da frizione e quindi sono soggetti a malfunzionamenti o rotture se non cambiati con una certa frequenza, in più contribuiscono ad innalzare il peso della vettura. Per tutte queste ragioni, normalmente, in ambito competitivo i sincronizzatori non sono utilizzati; questo in parte limita la vita del cambio in quanto sottoposto ad uno stress maggiore, ma a tutto vantaggio delle prestazioni della vettura che risultano considerevolmente incrementate.

Il cambio della nostra vettura ha la caratteristica di non disporre di sincronizzatori tuttavia, come precedentemente spiegato, questo non risulta essere uno svantaggio, inoltre dato lo scarso chilometraggio alla quale è sottoposta una vettura di FORMULA SAE non si pongono problemi di eccessiva usura del cambio.

---

<sup>5</sup>Condizione nella quale un motore si ritrova ad avere un regime di rotazione superiore al massimo consentito, incorrendo nel rischio di danneggiarsi

### 1.2.2 La cambiata senza frizione

Con un opportuna tecnica è possibile effettuare le cambiata senza l'utilizzo della frizione con lo scopo di velocizzare i tempi di cambiata, ma a patto di limitare la vita del cambio; tuttavia, come precedentemente esposto, questo inconveniente se mantenuto entro certi limiti nel nostro caso non risulta essere un problema.

L'obiettivo di questa tecnica è quello di limitare il più possibile la coppia trasmessa tra gli ingranaggi durante la cambiata, in modo da poter effettuare le cambiata dolcemente, senza incorrere nel rischio di danneggiare il cambio. Questo obiettivo è raggiunto in maniera differente a seconda se si desidera passare ad un rapporto superiore o inferiore:

**Passaggio ad un rapporto superiore:** In questo caso, prima della cambiata, ci si trova in piena accelerazione dunque è il motore che trasmette coppia alle ruote; quindi, per poterla limitare è necessario parzializzare l'acceleratore, senza tuttavia chiuderlo completamente in quanto questo comporterebbe una fase di decelerazione del veicolo, ovvero trasmissione di coppia dalle ruote al motore, che impedirebbe lo sganciamento dell'ingranaggio. A rapporto sganciato per poter ingranare la marcia successiva, che diminuisce la differenza di velocità tra i due alberi, è necessario abbassare i giri motore considerando che la velocità delle ruote resta pressapoco costante o al massimo diminuisce per effetto degli attriti che tendono a fermare il veicolo. In definitiva in una prima fase è necessario parzializzare l'acceleratore e poi parzializzarlo ulteriormente fino al raggiungimento della velocità ottimale dell'albero primario del cambio e quindi innestare il nuovo rapporto.

**Passaggio ad un rapporto inferiore:** Questo caso è l'opposto del precedente, ovvero la vettura si trova in fase di decelerazione con trasmissione di coppia dalle ruote verso il motore, è quindi necessario aumentare leggermente l'alimentazione di quest'ultimo in modo da limitare la coppia trasmessa dagli ingranaggi. A questo punto si può sganciare il rapporto e, dopo aver aumentato ulteriormente l'alimentazione in modo da incrementare i giri motore e raggiungere quindi il nuovo rapporto di trasmissione, innestare la nuova marcia.

Questa serie di operazioni possono essere svolte dal pilota o da un'opportuno dispositivo che le esegua automaticamente. In particolare, l'automatizzazione risulta indispensabile se si desidera scalare rapporto senza l'utilizzo della frizione in quanto l'esecuzione manuale risulta particolarmente complessa inoltre, se non è eseguita con una certa rapidità, comporta una perdita consistente di freno motore. Per questa ragione infatti raramente la tecnica di scalata è eseguita dal pilota, che ricorre piuttosto all'utilizzo della frizione.

In ultima istanza c'è da dire che, in entrambe le operazioni sopra esposte, le due fasi distinte di apertura dell'acceleratore, in caso di incremento di un rapporto, o di chiusura, in caso di decremento, hanno validità solo teorica. Infatti nella realtà, laddove la rapidità di cambiata è la priorità, non vengono rispettate in quanto l'arco temporale che intercorre tra un rapporto ed il successivo è talmente ristretto, che le inerzie meccaniche intrinseche del motore non consentono quest'operazione. Di conseguenza la cambiata viene effettuata fornendo un'unica variazione d'apertura dell'acceleratore.

### 1.2.3 Il *quick-shifter*

Da alcuni anni sul mercato si possono trovare dei dispositivi, chiamati *quick-shifter* o "*ambio elettronico*", applicabili ai normali cambi sequenziali delle motociclette che consentono l'inserimento della marcia superiore senza che si debba chiudere l'acceleratore o premere la frizione.

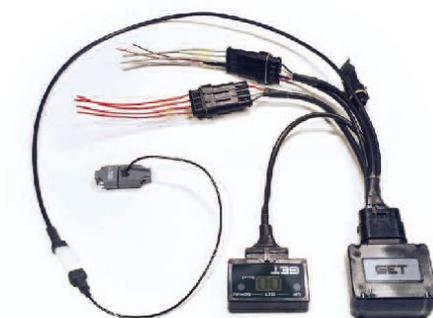
Questi dispositivi sono generalmente costituiti da un microinterruttore ed una centralina di controllo, con le seguenti funzioni:

- Il microinterruttore viene fissato lungo l'asta che comanda la leva del cambio, con lo scopo di segnalare alla centralina il momento in cui il pilota decide di innestare la marcia superiore, con l'accorgimento di montare una molla interna in modo da evitare che questo scatti inutilmente per di piccoli movimenti della leva causati a piccole sollecitazioni esterne (ad esempio: vibrazioni, buche sulla strada). Vedi fig:1.1
- La centralina viene collegata al microinterruttore e, tramite opportuni adattatori, alle bobine del motore. Una volta installato il *quick-shifter* quando il pilota desidererà innestare

la marcia superiore il microinterruttore scatterà, la centralina rileverà tale informazione e provvederà a togliere l'alimentazione alle bobine per un piccolo intervallo di tempo chiamato tempo di *cut-off* (tipicamente 40-70ms) spegnendo di fatto il motore, il tutto mentre il pilota sta innestando la marcia. Quest'operazione coincide con quella descritta in 1.3.2 e consente quindi al pilota di cambiare marcia senza che debba occuparsi d'altro, sostituendosi a lui nell'operazione di "alleggerire" l'acceleratore. Solitamente la centralina è dotata di alcuni tasti e un piccolo monitor o un barra led per poter regolare il tempo di *cut-off*. Esempio di kit completo in fig:1.2



**Figura 1.1:** Esempio di microinterruttori che funzionano rispettivamente in trazione o in compressione



**Figura 1.2:** Un kit quick-shifter completo di centralina e microinterruttore

Questo sistema di automatizzazione della salita di marcia si è rivelato molto efficace e poco stressante per il cambio, tanto che alcune case motociclistiche quali Aprilia, Ducati e BMW hanno deciso di fornire questo dispositivo di serie per i propri modelli più sportivi. Inoltre alcune marche che costruiscono centraline motore già da anni hanno integrato questi dispositivi nei loro modelli, consentendo quindi di poter utilizzare il *quick-shifter* senza dover montare centraline aggiuntive o effettuare modifiche al cablaggio.

L'MG 05/10, in particolare, monta una centralina di gestione del motore prodotta e fornita da ATHENA, sponsor ufficiale della squadra, che dispone di un sistema *quick-shifter* integrato. Per poter utilizzare tale dispositivo è sufficiente collegare ad un apposito pin di ingresso della centralina un segnale che le comunichi quando far partire il tempo di *cut-off* e impostargli via software la sua durata. Il segnale di inizio non è quindi indispensabile che sia fornito da un microinterruttore, ma può anche essere fornito con altri sistemi che mirino a ritardare il più possibile l'inizio del tempo di *cut-off*, visto che la sua durata è relazionata unicamente alle caratteristiche del cambio.

#### 1.2.4 Problematiche e precauzioni

Va specificato che il fuorigiri è una condizione estremamente pericolosa per un motore termico, specialmente per uno a quattro tempi come in questo caso; le motivazioni per le quali si incorre in questa situazione sono fondamentalmente due, che possiamo così classificare:

**Forzato** In questo caso il fuorigiri viene raggiunto a causa di un'eccessiva alimentazione<sup>6</sup> del motore in relazione a un determinato carico<sup>7</sup>. La causa del fuorigiri, quindi, è unicamente imputabile ad un'errata gestione dell'alimentazione da parte del pilota, tuttavia, al giorno d'oggi questa problematica può essere agevolmente risolta. In particolare, grazie alla centralina elettronica che gestisce il motore, è possibile tagliare o ridurre<sup>8</sup> l'alimentazione nel caso si raggiungano regimi troppo elevati; questa tipologia di dispositivi, spesso integrati nella centralina principale, vengono denominati limitatori di giri.

**Indotto** In questo caso il fuorigiri è indotto dall'esterno, ovvero risulta essere indipendente dall'alimentazione del motore. Tale condizione, infatti, si verifica quando il ruolo del motore e del carico si invertono ed è il carico che imprime al motore una coppia che tende ad incrementarne il numero di giri motore, mentre il motore si oppone, fornendo una coppia frenante. Quando il carico genera una coppia superiore a quella massima che il motore è in grado di opporvi, cioè quella che sfrutta attriti e perdite per pompaggio con alimentazione nulla, si raggiunge la condizione di fuorigiri. In un veicolo questa condizione si può verificare per esempio quando, con il veicolo in movimento, viene innestato un rapporto eccessivamente corto, oppure lungo una discesa, dove l'energia potenziale del veicolo viene convertita in cinetica e, nel caso in cui questa risulti superiore a quella che riesce ad opporvi il motore, ne fa aumentare la velocità del veicolo fino al raggiungimento e superamento del massimo regime di rotazione. In tale situazione appare evidente l'inutilità della tecnica adottata nel punto precedente in quanto siamo già nella condizione di massima coppia frenante che il motore può generare. L'unica tecnica adottabile per porre rimedio a questa situazione è lo sganciamento dell'albero motore da quello del carico, condizione che si può ottenere o premendo la frizione o innestando la folle; possibilità, quest'ultima, che tuttavia non sempre è agevole da raggiungere in un cambio sequenziale, visto che la folle non è inseribile da qualsiasi marcia.

Da questa descrizione è emerso che, mentre il fuorigiri forzato è facilmente evitabile, quello indotto risulta essere un problema serio e di non banale soluzione. In molte competizioni, tuttavia, questo problema viene tralasciato confidando nell'abilità, nella competenza e nell'esperienza del pilota che quindi non può permettersi di sbagliare.

L'eventualità di incorrere in un fuorigiri all' "automobilista medio" può sembrare molto remota in quanto anche in gran parte delle auto d'uso comune è presente un cambio manuale, eppure l'eventualità di effettuare un fuorigiri risulta sconosciuta ai più per quanto è rara. Questa considerazione tuttavia non tiene conto che, durante una competizione automobilistica, il motore viene molto sollecitato alla continua ricerca delle massime prestazioni e quindi della massima potenza, obbligandolo ad operare a regimi mediamente molto elevati, non lontani da valori critici. Condizione questa che fa sì, nel caso in cui per errore si inserisca una marcia più corta, di incorrere in un sicuro fuorigiri indotto. A questo punto è facile capire quanto queste circostanze siano differenti da quelle di normale utilizzo di un veicolo, dove, anche se si sbaglia a innestare un rapporto, la consistente lontananza dei giri motore dai valori massimi evita problemi di sorta.

V'è altresì da aggiungere che, nei cambi tradizionali, il tentativo di innesto di un rapporto che porterebbe a un fuorigiri indotto, o anche solamente a un consistente innalzamento del regime di rotazione del motore, non è un'operazione così semplice. La leva di comando, infatti, risulta essere particolarmente dura opponendosi di fatto a tale operazione. La causa di questo fenomeno è da imputarsi a una grande differenza di giri tra l'albero primario e secondario del cambio, questo fa sì che sia necessario uno sforzo aggiuntivo per permettere all'ingranaggio di innestarsi; condizione, questa, che solitamente porta il guidatore ad accorgersi dell'errore in tempo e ad innestare quindi il rapporto corretto. Questo tipo di accorgimenti purtroppo non risultano altrettanto efficaci in campo competitivo, dove il pilota innesta sempre la marcia con estrema violenza con il fine di limitare al minimo il tempo di cambiata e al contempo quello in cui la guida è affidata ad una sola mano.

<sup>6</sup>Per alimentazione si intende la percentuale di acceleratore che risulta aperto, non considerando cioè quantità assolute di carburante o aria, ma sempre rispettando la corretta carburazione del motore

<sup>7</sup>Con carico si intende una coppia applicata al motore che si oppone a quella da esso generata

<sup>8</sup>Impostando tramite software, sotto opportune condizioni, la chiusura parziale o totale della valvola d'ingresso dell'aria del motore, la sospensione dell'alimentazione di una o più bobine, di uno o più iniettori, la riduzione dei tempi di iniezione di questi ultimi, la variazione dell'anticipo d'accensione o una qualsiasi combinazione di queste possibilità

## 1.3 L'elettronica

In questa sezione tratto brevemente l'elettronica dell'intera vettura in modo da fornire al lettore una comprensione più ampia dei dispositivi presenti, della loro utilità e di come si possono eventualmente sfruttare in merito alla gestione del cambio.

La vettura, innanzitutto, è dotata di un sistema elettrico che rispetta il regolamento, dispone quindi di due interruttori generali d'emergenza (vedi Appendice A) e di uno in caso di malfunzionamento dell'impianto frenante (Appendice B). Oltre a questi interruttori vi sono alcuni relè e fusibili, i primi con lo scopo di comandare il motorino d'avviamento, di alimentare i vari dispositivi elettronici e per permettere alla centralina di comandare dispositivi di potenza quali pompe benzina, ventole di raffreddamento, ecc. mentre i secondi servono per proteggere i dispositivi elettrici/elettronici presenti sull'auto in modo da salvaguardarli da sovraccarichi o cortocircuiti. Questi componenti sono indispensabili per distribuire adeguatamente l'elettricità lungo tutta la vettura e per poter intervenire tempestivamente in caso di guasti, ma non risultano interessanti per quanto concerne il nostro progetto, lo sono invece altri due componenti:

- La centralina motore
- Il sistema di acquisizione dati

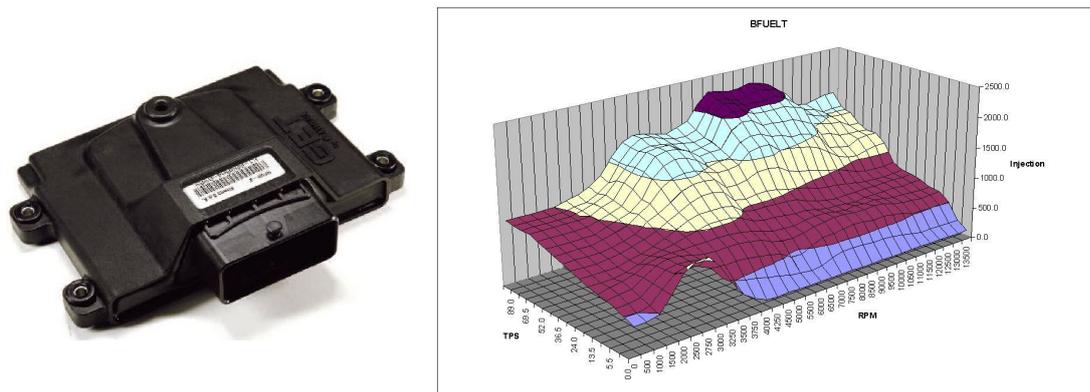
### 1.3.1 La centralina motore

La centralina motore montata in vettura ha lo scopo principale di ottimizzarne la carburazione in ogni condizione operativa del motore. Per far ciò dispone di numerosi ingressi adibiti a rivelare vari parametri, tra questi i principali sono:

1. Giri del motore
2. Apertura della farfalla dell'aria
3. Temperatura motore
4. Temperatura atmosferica
5. Pressione atmosferica
6. Sonda Lambda

Tutte queste informazioni vengono elaborate dalla centralina che fornisce in uscita dei segnali che vanno a comandare le bobine e gli iniettori. I primi cinque ingressi consentono alla centralina di valutare qual'è lo stato del motore e di conseguenza di decidere quali uscite deve fornire, mentre la sonda lambda è un sensore che rileva quanto buona è in quel momento la combustione, fornisce quindi una valutazione postuma della bontà della combustione, consentendo però di capire se è opportuno effettuare dei correttivi alla modellizzazione del motore. Oltre a questo scopo la centralina serve anche per gestire tutti quei "servizi" che consentono al motore stesso di non spegnersi o danneggiarsi, è la centralina che infatti controlla l'alimentazione della pompa benzina ed è sempre lei che regola la temperatura del motore agendo sulle ventole di raffreddamento. La valutazione dei parametri di carburazione sono stati effettuati dal RACE-UP montando il motore in un banco prova, ovvero in un'apposito banco che riproducendo le varie condizioni di funzionamento del motore permette di poterne valutare il comportamento e quindi la corretta carburazione.

Oltre a queste operazioni fondamentali la centralina in questione acquisisce altri sensori con lo scopo di poter effettuare un controllo di stabilità nel caso venga montata in una motocicletta. Tutti questi dati possono essere forniti in uscita verso altri dispositivi e per far ciò utilizza un protocollo di comunicazione denominato CAN, un noto standard automobilistico. Questi ingressi aggiuntivi sono riservati a delle ruote foniche, ovvero dei sensori che rilevano la velocità di rotazione di una certa ruota dentata.



**Figura 1.3:** Centralina Athena HPUH e una delle mappe di carburazione del motore

### 1.3.2 Il sistema di acquisizione dati

Il sistema di acquisizione dati è un dispositivo che ha lo scopo di registrare tutti i parametri della vettura in un opportuno supporto in modo che questi possano essere successivamente analizzati. Il fine di questa operazione di registrazione è quello di facilitare la comprensione di un'eventuale guasto, nonché quella di intervenire con precisione nelle regolazioni della vettura. Il sistema di acquisizione dati che è stato utilizzato è un Motec ADL2; questo dispositivo ha la caratteristica di avere un grande display che consente di poter visualizzare istantaneamente qualunque informazione della vettura, viene pertanto utilizzato anche per comunicare al pilota alcune informazioni che possono aiutarlo nella guida (ad esempio regime motore, marcia inserita, ecc.). Nel caso in cui i parametri che richiedono l'attenzione del pilota siano numerosi si ha la possibilità di visualizzare diverse schermate, le quali possono essere selezionate in qualsiasi momento premendo degli appositi pulsanti che sono stati posti sul volante. C'è inoltre la possibilità di lanciare degli allarmi, ovvero la comparsa di una scritta accompagnata da un'eventuale lampeggio della retroilluminazione del display, che segnala l'incorrenza di una condizione critica che deve essere immediatamente segnalata al pilota (ad esempio: surriscaldamento del motore, scarsa lubrificazione, ecc.).

Il sistema di acquisizione dati deve acquisire in parte gli stessi segnali che sono necessari alla centralina motore per poter svolgere la sua funzione, sarebbe quindi necessario duplicare tutti i collegamenti a questi sensori, cosa non molto intelligente in primo luogo perché si sprecano molti ingressi del sistema di acquisizione dati e in secondo luogo perché si complica notevolmente il cablaggio. Per questo motivo anche il sistema di acquisizione dati dispone di una porta CAN, consentendo quindi di poter acquisire non solo tutte le informazioni che acquisisce la centralina, ma anche quelli derivati dall'elaborazione di questi (ad esempio i tempi di iniezione, che sono un'indice del consumo).



**Figura 1.4:** Il Motec ADL2 montato in vettura

## Capitolo 2

# Sistema di controllo

Le caratteristiche del cambio consentono di poter comandare la cambiata semplicemente premendo o tirando un'apposita leva; questa operazione, che nell'uso per cui è stato concepito viene svolta dalla gamba del motociclista, può facilmente essere sostituita da un pistone meccanico oppure da un motore elettrico montato coassialmente all'albero di comando.

### 2.1 Valutazione dei vari sistemi di automazione

A questo punto si è iniziato a valutare quale fosse la soluzione più conveniente tra l'utilizzo di un motore elettrico e un pistone meccanico, tuttavia i primi si sono presto rivelati poco adatti al controllo di questo sistema.

#### 2.1.1 Attuazione con motore elettrico

I motori per piccole applicazioni, infatti, nonostante abbiano pregi quali leggerezza e consumi energetici ridotti, non dispongono di una coppia sufficiente per poter azionare la leva di comando, per questa ragione sono stati scartati fin da subito. Di conseguenza si sono cercati dei motori più grandi, in grado di raggiungere i valori di coppia richiesti; anche questi sono stati scartati quasi immediatamente, fondamentalmente per due motivi:

1. In primo luogo, risultavano essere eccessivamente pesanti e ingombranti e questo, oltre a incidere sulle prestazioni della vettura per ragioni dovute al peso non più trascurabile, risultavano essere anche un problema altrettanto importante a livello di ingombro, in quanto il telaio, per ragioni puramente meccaniche, avvolge il motore e quindi trovare lo spazio per inserire un motore elettrico sarebbe risultato molto difficile;
2. In secondo luogo, vi è il problema che montare un motore elettrico che generi coppie elevate per brevi periodi di tempo richiede per forza di cose anche che questo necessiti di grandi picchi di potenza elettrica in ingresso e questo risulta essere un problema non irrilevante. Per capirne le motivazioni è bene ricordare le funzioni alle quali ha il compito di assolvere una batteria in un veicolo:
  - fornire un'iniziale picco di potenza necessario a far girare il motorino d'avviamento e quindi avviare il motore termico
  - mantenere stabile la tensione una volta che il motore termico è avviato

A motore termico acceso, infatti, vari dispositivi richiedono per poter funzionare di energia elettrica quali: la centralina motore, la pompa del carburante, le ventole di raffreddamento, le bobine, gli iniettori, i sistemi di controllo della vettura (spie di emergenza, tachimetro, contagiri, ecc..) e, in generale, tutti gli eventuali servizi ausiliari.

Questa energia non è fornita dalla batteria che altrimenti si scaricherebbe dopo pochi minuti, bensì dall'alternatore che, costantemente, fornisce potenza elettrica al sistema compensandone i vari assorbimenti. Lo scopo della batteria in questa fase di utilizzo è sia quella di ricaricarsi sfruttando la corrente generata dall'alternatore in modo da ripristinare la sua piena carica, sia sopperire ai picchi di potenza richiesti da bobine e

iniettori mantenendo la tensione il più possibile costante, svolgendo cioè la funzione di un classico condensatore di filtro. Questa funzione risulta essere compromessa nel caso in cui i picchi risultino essere troppo elevati e questo può portare al cattivo funzionamento di molti dispositivi elettronici il cui range di tensione di funzionamento è piuttosto ristretto. Questa problematica presenta comunque una facile soluzione, ovvero l'installazione di una batteria più performante, ciò tuttavia contribuisce sempre ad un'aumento del peso dell'auto; inoltre, il continuo utilizzo del cambio che avviene in questo tipo di competizioni porta a un notevole incremento del valore medio di potenza elettrica richiesto e quindi a un maggiore stress dell'alternatore che è invece studiato per sopportare, oltre ai servizi essenziali, carichi molto modesti.

### 2.1.2 Attuazione con sistema meccanico

Visto che l'attuazione elettrica non è risultata soddisfacente si è pensato di utilizzare qualche sistema meccanico che potesse svolgere tale funzione. Il più semplice è risultato essere l'utilizzo di un pistone meccanico che, mosso da un fluido in pressione, attuasse la leva di comando; il fluido che si è deciso fin da subito di utilizzare è stato l'azoto o, in alternativa, l'aria in quanto facilmente reperibili in pressione, non infiammabili e molto più leggeri rispetto all'utilizzo di fluidi più densi quali, ad esempio gli olii. Questa scelta è in accordo con il regolamento della FORMULA SAE dove, al punto 13.1a, riferendosi a qualsiasi gas compresso utilizzato in vettura come mezzo di azionamento recita:

*13.1a: Il gas d'esercizio dell'impianto deve essere non infiammabile. Ad esempio: aria, azoto, anidride carbonica.*

A questo punto si presentano due tipologie di utilizzo del fluido:

- Montare in vettura un piccolo recipiente d'aria alla pressione di attuazione del pistone pneumatico (medio-bassa) e una pompa che provveda a mantenere costante la pressione dell'impianto;
- Montare in vettura un recipiente più grande ad altissima pressione che permetta un'accumulo d'aria sufficiente a garantire alla vettura di effettuare le cambiate per un'intero endurance<sup>1</sup>.

La prima delle due soluzioni che dava, a differenza della seconda, un'autonomia illimitata nell'utilizzo del cambio, non è risultata essere applicabile in quanto nel mercato non si sono trovati piccoli compressori in grado di assolvere a questa funzione. Si è quindi optato per la seconda alternativa considerando che, rispetto alla prima, nonostante si sia persa la totale autonomia del cambio, il peso e la complessità dell'intero sistema sono diminuite a vantaggio di prestazioni e dell'eventuale possibilità di malfunzionamenti.

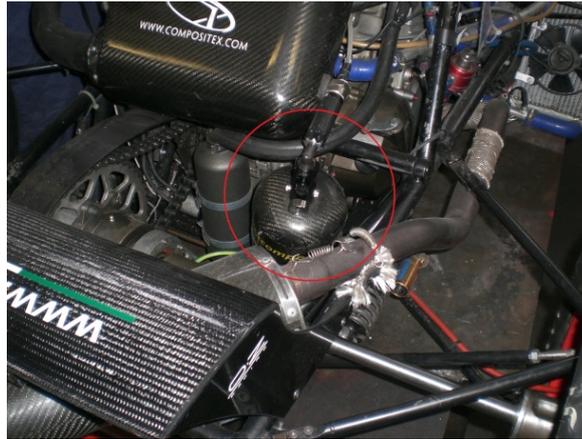
## 2.2 L'attuazione del cambio

L'attuazione del cambio è sostanzialmente composta da 6 componenti:

1. Bombola ad alta pressione
2. Riduttore di pressione
3. Elettrovalvola
4. Pistone pneumatico
5. Switch di comando al volante
6. Centralina di controllo

---

<sup>1</sup>In un'evento Formula SAE l'endurance è la prova più lunga durante la quale non è concesso di intervenire in alcun modo sull'auto se non per aiutare i piloti a scendere e a risalire in vettura durante la fase del cambio pilota, che deve avvenire una volta sola esattamente a metà della prova



**Figura 2.1:** La bombola del cambio montata in vettura all'interno di una protezione in fibra di carbonio

La bombola che è stata utilizzata ha una capacità di 1,5 litri ed è realizzata in alluminio rivestito con materiale composito con una pressione massima ammessa pari a 310 bar; per questo prototipo, tuttavia, si è deciso di fissare la pressione massima ammessa pari a 200 bar. Questa pressione risulta essere troppo elevata per poter comandare direttamente il pistone pneumatico, per cui è necessario collegare all'uscita della bombola un opportuno riduttore con lo scopo di portare la pressione al valore di esercizio, cioè al massimo pari a 7 bar dai 200 di partenza. Per raggiungere tale scopo sono stati utilizzati due riduttori:

- Uno primario con lo scopo di portare la pressione a 31 bar, ma che dispone anche di un rubinetto per dar la possibilità di isolare l'intero impianto, una valvola unidirezionale di sicurezza e un manometro che misura la pressione a monte del riduttore, ovvero la pressione della bombola;
- Poiché la pressione in uscita dal riduttore primario non è la pressione richiesta è necessario metterne un'altro che riduca la pressione da 31 ai 7 bar di progetto.

Una volta che la pressione è quella richiesta l'aria compressa viene collegata ad un'elettrovalvola che, incanalando opportunamente il flusso d'aria, regola i movimenti del pistone pneumatico. L'elettrovalvola dispone un'ingresso e di due uscite, da collegare al pistone pneumatico per farlo muovere nell'una o nell'altra direzione.

L'elettrovalvola dispone di due bobine di eccitazione che a seconda del loro stato ne cambiano il comportamento, in particolare:

**Nessuna bobina eccitata:** L'ingresso dell'elettrovalvola, ovvero il lato collegato alla bombola, è isolato dal pistone pneumatico mentre le due uscite, che sono collegate ai due lati del pistone, sono alla pressione atmosferica in modo da consentire al gas presente nel pistone di defluire all'esterno. In questa situazione quindi la leva del cambio ha la possibilità di posizionarsi nella posizione di neutro e di permanervi stabilmente.

**Una bobina eccitata:** Connette l'ingresso con una delle uscite del pistone, mantenendo l'altra alla pressione atmosferica, ciò causa un movimento del pistone, che imprime una forza sulla leva del cambio. E' da notare che una volta che il sistema è a regime, cioè quanto il pistone ha raggiunto la sua massima espansione non vi è un'ulteriore richiesta di gas per il mantenimento di tale posizione.

**Entrambe le bobine eccitate:** E' una situazione non ammessa nella quale la valvola può risultare instabile e dunque da evitare.

Gli switch a volante non sono altro che il mezzo con cui il pilota segnala di voler cambiare rapporto, mentre la centralina di controllo ha il compito di gestire ed interfacciare le richieste del pilota con il sistema di attuazione.

## 2.3 Il controllo

Il metodo più semplice per pilotare l'intero sistema è quello di collegare direttamente l'elettrovalvola in serie agli interruttori presenti sul volante, tuttavia questo comporta alcune conseguenze:

- C'è il rischio che non vengano rispettati i tempi minimi di spostamento del pistone con il conseguente fallimento nel tentativo di cambiata;
- C'è il rischio di incorrere in ridondanze del sistema e quindi in cambiate multiple indesiderate;
- Non ci sono protezioni contro il rischio di incorrere in fuorigiri indotti;
- Si è costretti ad utilizzare il microinterruttore per comunicare alla centralina il momento in cui iniziare il tempo di *cut-off*, perdendo la possibilità di gestirlo.

Queste motivazioni hanno indotto l'introduzione di un sistema di controllo per gestire l'elettrovalvola in modo da poter regolare con precisione e a piacimento ogni parametro di cambiata, senza richiedere al pilota attenzioni particolari nella gestione degli interruttori a volante.

Tutte queste osservazioni sono volte ad automatizzare la cambiata e ad evitare l'uso della frizione in caso di salita di marcia sfruttando il quick-shifter integrato nella centralina motore, tuttavia non risolvono il problema della necessità dell'utilizzo della frizione in scalata.

### 2.3.1 La scalata

Come spiegato in 1.3.2 per poter scendere di rapporto in fase di frenata senza l'utilizzo della frizione è necessario aumentare temporaneamente l'alimentazione del motore in modo che tra questo e le ruote non vi sia trasferimento di coppia. Per far ciò è quindi necessario agire sull'acceleratore, ovvero aprire la farfalla che regola la quantità d'aria entrante nel motore, operazione facilmente eseguibile in modo automatico nei motori moderni dove tale farfalla è gestita dalla centralina per mezzo di un motore elettrico e non direttamente dal pilota tramite un cavo, come generalmente avviene in motori più datati.

Questa soluzione, nonostante sia facilmente adottabile con la centralina motore di cui disponiamo, non è possibile a causa del regolamento della FORMULA SAE, che nella sezione B8.5 riguardante l'acceleratore e l'attuazione dello stesso recita:

*B8.5.2: L'acceleratore deve essere attuato meccanicamente. Ad esempio: con sistemi a cavi o ad aste. L'uso di controlli elettronici dell'acceleratore (ETC) o di "drive by wire" sono proibiti.*

La regola in questione è oggettivamente estremamente restrittiva riguardo ad un possibile utilizzo o sviluppo di tecnologie in tal senso, ma va ricordato che in primo luogo la FORMULA SAE è una competizione tra studenti universitari, con l'obiettivo principe di formarli e dargli la possibilità di poter applicare le conoscenze che hanno appreso dai libri nel mondo reale. Questo significa che spesso questi studenti non hanno molta esperienza, aspetto che gli dà il vantaggio di avere la mente "libera" da preconcetti, che invece spesso si riscontrano in chi di esperienza ne ha molta, dandogli la possibilità di seguire strade di sviluppo fantasiose ed a prima vista sconsigliabili che nel mondo del lavoro difficilmente potrebbero intraprendere. Per contro, la possibilità di incorrere in errori è molto più facile rispetto a chi ha molta esperienza ed applica soluzioni consolidate ad ogni problematica. Per fare un'esempio, alcune squadre universitarie presenti alle competizioni a cui ha partecipato il RACE-UP hanno deciso di autoprodursi le centraline motore integrando al loro interno anche un sistema di acquisizione dati e di relè/fusibili, in questo modo hanno potuto scegliere le caratteristiche da dare alla propria centralina in funzione unicamente delle proprie esigenze e del regolamento, risparmiando peso e aumentando notevolmente la semplicità e la compattezza dell'intero impianto elettrico.

La FORMULA SAE, tuttavia, dovendo conciliare la libertà di sviluppo con la sicurezza ha sempre scelto, com'è giusto che sia, di dare la priorità alla sicurezza. Poiché la possibilità di avere la piena gestione dell'acceleratore, in caso di guasti o malfunzionamenti, può portare a situazioni di pericolo, ha deciso di istituire la regola B8.5.2. Un'altro esempio di limitazioni allo sviluppo in favore della sicurezza, per ragioni ancora più lampanti, si può ritrovare nell'articolo 7 che parla del sistema frenante, recitando:

*B7.1: L'auto deve essere equipaggiata con un sistema frenante che agisce su tutte e quattro le ruote con un singolo controllo.*

*B7.1.4: Il "Brake by wire" è proibito.*

In relazione alla limitazione della gestione dell'acceleratore si potrebbe contestare questa scelta sostenendo che invece di intervenire a livello regolamentare sull'acceleratore si potrebbe farlo sulla frizione, in tal modo, nel caso si presentino dei problemi, il pilota premendo quest'ultima può comunque rallentare. Tuttavia, in questo caso, ci si dovrebbe affidare al pilota e alla sua capacità di mantenere la concentrazione anche in situazioni critiche, quindi non sicura come la soluzione che è stata adottata, inoltre ciò imporrebbe l'utilizzo di una frizione ad azionamento manuale escludendo quelle ad azionamento automatico (ad esempio quelle centrifughe<sup>2</sup>).

Date queste limitazioni appaiono possibili solo due soluzioni:

- Scalare la marcia senza l'utilizzo della frizione e senza aumentare l'alimentazione;
- Automatizzare anche la frizione.

La prima opzione è stata scartata fin da subito in quanto la durata del cambio si riduce notevolmente, inoltre spesso la marcia ha molte difficoltà nell'innesto, non costituendo di fatto una soluzione affidabile anche per periodi limitati. La seconda opzione è invece più plausibile infatti questa potrebbe essere facilmente attuata tramite un sistema analogo a quello del cambio, cioè collegando la leva di controllo della frizione con un'altro pistone pneumatico pilotato mediante un'altra elettrovalvola alla stessa bombola di accumulo del cambio. Tuttavia a differenza del cambio, che deve semplicemente essere mosso nel più breve tempo possibile, la frizione può essere gestita in maniera analoga solo quando viene staccata, mentre la fase di rilascio deve essere più dolce in modo da garantire che i due alberi raggiungano la corretta velocità di rotazione gradualmente, evitando che il freno motore risulti troppo brusco. Per risolvere questo problema si presentano due soluzioni:

**Controllo meccanico** La soluzione più semplice è quella di montare in parallelo al pistone pneumatico un'ammortizzatore, il quale agisce secondo la funzione 2.1, limitando la velocità di rilascio della frizione. Questo ammortizzatore, tuttavia, deve agire secondo tale funzione soltanto nella fase di rilascio, lasciando la possibilità alla frizione di scattare nel più breve tempo possibile. Sarà poi possibile cambiare le tempistiche di stacco variando il coefficiente  $c$  fino ad ottenere il migliore rilascio possibile.

$$F_d = -cv = -c\dot{x} = -c\frac{dx}{dt} \quad (2.1)$$

**Controllo elettronico** Un'altra soluzione leggermente più complessa per raggiungere il medesimo scopo è quella di controllare il flusso d'aria uscente dal pistone. Per fare questo è sufficiente installare un'ulteriore elettrovalvola agente tra l'uscita del pistone pneumatico e l'ambiente esterno, per poi modularne l'apertura o chiusura a piacere.

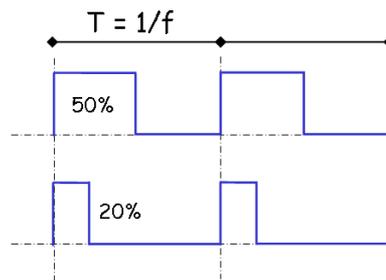
La prima soluzione tuttavia presenta lo svantaggio di essere poco versatile, la risposta del sistema infatti è molto semplice non riuscendo a seguire perfettamente la curva di stacco ideale. Questo problema risulta essere tuttavia poco rilevante in quanto la velocità di stacco della frizione durante la competizione risulta essere sufficientemente elevata da rendere tale non idealità trascurabile. Un problema che invece non è trascurabile riguarda la fase di partenza del veicolo, in tale occasione infatti lo stacco della frizione risulta essere molto più lento rispetto a quanto la vettura è in movimento quindi in questo caso è necessario ricorrere ad un sistema manuale che agisce in parallelo all'ammortizzatore e che il pilota dovrà utilizzare solo in fase di partenza. Questa soluzione risulta tuttavia complessa e poco efficace, richiedendo l'utilizzo di due attuazioni diverse per uno stesso sistema e che, almeno in parte, riporta all'utilizzo di una frizione a controllo manuale, di fatto non risolvendo il problema dell'automatizzazione.

La situazione è diversa invece nel secondo caso, ovvero per il sistema elettronico, infatti l'espansione del pistone in questo caso può, almeno teoricamente, rispettare qualsiasi legge purché si moduli opportunamente l'elettrovalvola. Una modulazione che può risultare efficace

<sup>2</sup>Questo tipo di frizioni ha un funzionamento automatico legato al regime di rotazione raggiunto dal motore o, più in generale, dell'asse su cui ruotano.

è la PWM<sup>3</sup>, che consiste nell'apertura e chiusura dell'elettrovalvola ad una frequenza elevata variandone il duty-cycle<sup>4</sup> in relazione alle esigenze del sistema. Ad esempio con un duty-cycle del 100% la frizione ritornerà della posizione di partenza nel minor tempo possibile, non incontrando nessun impedimento, mentre con un duty-cycle del 0% la frizione risulterà ferma nella posizione in cui si trova. Variando opportunamente il duty-cycle è dunque possibile coprire tutte le condizioni intermedie, e dunque avere uno stacco della frizione che corrisponde perfettamente a quella voluta. In questo caso, tuttavia, risulta esserci un altro inconveniente; questo tipo di controllo, infatti, impone di avere una frequenza di lavoro elevata, impossibile da raggiungere con delle elettrovalvole che hanno risposte in frequenza estremamente basse, a malapena qualche centinaio di hertz. Questo riduce notevolmente l'efficienza di questa modulazione, rendendo il controllo molto meno preciso. In luogo di questo inconveniente risulta ancora meno fattibile la realizzazione di un sistema che sia in grado di modulare la frizione nella fase di partenza in quanto il sistema in questo caso dovrebbe risultare ancora più veloce nella risposta, in modo da evitare pattinamenti delle ruote o scarsa trasmissione di coppia alle stesse. Ciò porterebbe dunque, come nel primo caso, alla realizzazione di un sistema manuale parallelo a quello elettromeccanico da utilizzare esclusivamente per le partenze.

A seguito di queste considerazioni è stato deciso di non automatizzare anche la frizione, in quanto in ogni caso l'attuazione non sarebbe stata totalmente automatica e comunque difficilmente modulabile. Questa soluzione rallenta oggettivamente la scalata di un rapporto, ma lascia al pilota la piena libertà di utilizzare il freno motore al meglio, non appesantisce la vettura con l'utilizzo di due controlli volti a pilotare il medesimo sistema e inoltre non compromette troppo le prestazioni in quanto in fase di scalata non è così importante che la cambiata avvenga in tempi rapidissimi poichè non è il motore a fornire la maggior coppia frenante, ma appunto il sistema frenante.



**Figura 2.2:** Esempio di modulazione PWM

<sup>3</sup>Pulse Width Modulation

<sup>4</sup>Per duty-cycle si intende il rapporto tra la durata in cui un segnale risulta alto e il periodo totale del segnale, e serve a esprimere per quanta porzione di periodo il segnale è a livello alto

## Capitolo 3

# Studio del sistema hardware

In questo capitolo verrà descritta l'architettura della scheda di controllo che è stata realizzata giustificando le varie scelte progettuali.

### 3.1 Gli ingressi

In primo luogo questa scheda dovrà avere la possibilità di rilevare le volontà del pilota di cambiare rapporto, sia per quanto riguarda l'incremento sia il decremento di un rapporto, quindi risultano necessari due ingressi riservati a questo scopo. Questo tipo di informazione è ovviamente di tipo digitale, si può pertanto facilmente ricorrere all'installazione sul volante di due interruttori che il pilota può azionare agevolmente. Questi risultano gli unici due ingressi assolutamente fondamentali per il funzionamento della scheda, tuttavia per garantire la salvaguardia delle componenti meccaniche è opportuno prevedere dei possibili ingressi di sicurezza che abilitano o meno la cambiata in modo da evitare fuorigiri indotti oppure scalate senza l'utilizzo della frizione.

In particolare durante la scalata si presentano due condizioni che necessitano di un'ulteriore conferma alle volontà del pilota:

- Controllare se, quando viene ricevuto il segnale di scalata, la frizione sia già stata premuta, in modo da evitare il danneggiamento del cambio;
- Disabilitare la scalata di marcia nel caso in cui il regime di rotazione del motore risulti superiore alla soglia corrispondente al regime massimo ammesso con il rapporto inferiore.

Per far questo è dunque necessario predisporre altri due ingressi digitali, in particolare nel primo caso è sufficiente installare un'interruttore sulla leva di controllo della frizione che commuti stato quanto questa viene completamente premuta, mentre nel secondo caso la situazione non è così semplice, ma è possibile sfruttare il sistema di acquisizione dati per semplificare le cose. Questo dispositivo infatti dispone di alcune uscite sia digitali che analogiche; inoltre, tramite il collegamento alla centralina motore conosce istantaneamente il valore dei giri motore e della marcia inserita. Si può dunque facilmente riservare un'uscita allo scopo di abilitare o meno una scalata in funzione dei giri motore. Ciononostante, questo tipo di sicurezza ha incontrato il parere negativo dei piloti della squadra, che hanno preferito rinunciare a questa sicurezza mantenendo la libertà di inserire un rapporto inferiore anche ad una velocità del veicolo troppo elevata, gestendo poi la frizione in modo da non incorrere in fuorigiri, avendo in questo modo a disposizione un freno motore maggiore. Per questa ragione si è optato per implementare in questo sistema di controllo solamente il controllo della frizione in fase di scalata.

### 3.2 Le uscite

Le uscite in questione sono tre, due con lo scopo di comandare l'elettrovalvola e una che invia il segnale di inizio del tempo di *cut-off* alla centralina.

L'elettrovalvola funziona con un'alimentazione di 12V e la sua equivalenza elettrica è rappresentabile con una resistenza in serie ad un'induttore. A questo particolare va posta una particolare attenzione in fase di progettazione in quanto se alla parte induttiva sono applicate

brusche variazioni di corrente queste possono causare sovratensioni capaci di danneggiare altri componenti.

Il segnale da inviare alla centralina risulta semplice, deve essere lasciato normalmente in alta impedenza (o volante) e, quando si desidera far partire il tempo di cut-off, va posto a massa. Questa operazione nei kit commerciali viene eseguita da un microinterruttore (vedi Fig.1.1) collegato da un lato alla centralina e dall'altra al telaio (che coincide con la massa del veicolo), mentre in questo caso deve essere effettuato dalla scheda.

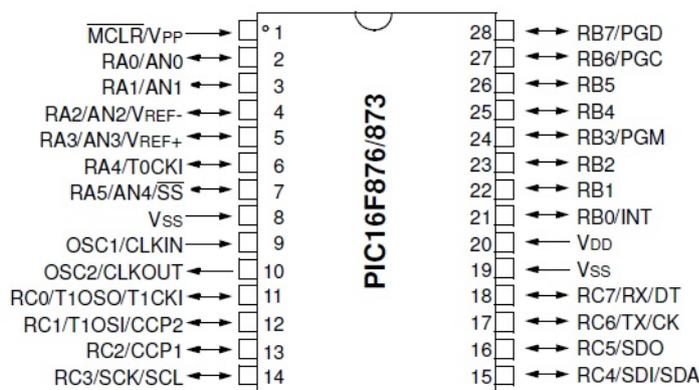
### 3.3 Il microcontrollore

Per effettuare il controllo del cambio è necessario poter variare numerosi parametri, quali ad esempio i tempi di cambiata o il ritardo con cui inviare il segnale di cut-off alla centralina, con il fine di ottimizzare al meglio la cambiata riducendone al massimo i tempi senza però compromettere l'affidabilità della meccanica. Per queste ragioni si è ricorso all'utilizzo di microcontrollore programmabile in modo che sia possibile variare agevolmente questi parametri con rapidità e precisione.

Il microcontrollore che si è deciso di utilizzare è stato il PIC 16F873-A. Questo integrato, come vedremo, risponde largamente alle esigenze sopra esposte, si sarebbe potuto infatti utilizzare un dispositivo meno prestante, ma questa prima scheda è stata pensata per un'uso generico e il più versatile possibile, in modo da rispondere anche ad esigenze che si fossero presentate in corso d'opera.

Il PIC in questione è dotato di 28 piedini, che permettono di avere molti pin riservati a ingressi/uscite; infatti, in questo PIC ci sono ben tre porte riservate a tale scopo, totalmente configurabili via software con la possibilità, variando il contenuto di opportuni registri, di definire ciascun bit di ciascuna porta come ingresso o uscita.

Nello schema sottostante è riportata la piedinatura di tutto il PIC e a fianco di ciascun pin l'acronimo che ne ricorda la funzione.



**Figura 3.1:** La piedinatura del PIC16f873-A

Come si può notare ciascun pin può avere diverse funzioni, che potranno variare a seconda delle impostazioni che si decide di scegliere. Nella tabella sottostante, ripresa dal datasheet, sono descritte le funzioni che può assumere ciascun pin e le caratteristiche hardware.

**Tabella 3.1:** Descrizione della funzione di ciascuno dei pin del PIC16f873.

Pin Name	DIP Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	9	I	ST/CMOS <sup>(3)</sup>	Oscillator crystal input/external clock source input.

*Continua nella prossima pagina*

*Continua dalla pagina precedente*

Pin Name	DIP Pin#	I/O/P Type	Buffer Type	Description
OSC2/CLKOUT	10	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, the OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
$\overline{MCLR}/VPP$	1	I/P	ST	Master Clear (Reset) input or programming voltage input. This pin is an active low RESET to the device. PORTA is a bi-directional I/O port.
RA0/AN0	2	I/O	TTL	RA0 can also be analog input0.
RA1/AN1	3	I/O	TTL	RA1 can also be analog input1.
RA2/AN2/VREF-	4	I/O	TTL	RA2 can also be analog input2 or negative analog reference voltage.
RA3/AN3/VREF+	5	I/O	TTL	RA3 can also be analog input3 or positive analog reference voltage.
RA4/T0CKI	6	I/O	ST	RA4 can also be the clock input to the Timer0 module. Output is open drain type.
RA5/ $\overline{SS}$ /AN4	7	I/O	TTL	RA5 can also be analog input4 or the slave select for the synchronous serial port.
RB0/INT	21	I/O	TTL/ST <sup>(1)</sup>	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0 can also be the external interrupt pin.
RB1	22	I/O	TTL	
RB2	23	I/O	TTL	
RB3/PGM	24	I/O	TTL	RB3 can also be the low voltage programming input.
RB4	25	I/O	TTL	Interrupt-on-change pin.
RB5	26	I/O	TTL	Interrupt-on-change pin.
RB6/PGC	27	I/O	TTL/ST <sup>(2)</sup>	Interrupt-on-change pin or In-Circuit Debugger pin. Serial programming clock.
RB7/PGD	28	I/O	TTL/ST <sup>(2)</sup>	Interrupt-on-change pin or In-Circuit Debugger pin. Serial programming data.
RC0/T1OSO/T1CKI	11	I/O	ST	PORTC is a bi-directional I/O port. RC0 can also be the Timer1 oscillator output or Timer1 clock input.
RC1/T1OSI/CCP2	12	I/O	ST	RC1 can also be the Timer1 oscillator input or Capture2 input/-Compare2 output/PWM2 output.

*Continua nella prossima pagina*

*Continua dalla pagina precedente*

Pin Name	DIP Pin#	I/O/P Type	Buffer Type	Description
RC2/CCP1	13	I/O	ST	RC2 can also be the Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	14	I/O	ST	RC3 can also be the synchronous serial clock input/output for both SPI and I2C modes.
RC4/SDI/SDA	15	I/O	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I2C mode).
RC5/SDO	16	I/O	ST	RC5 can also be the SPI Data Out (SPI mode).
RC6/TX/CK	17	I/O	ST	RC6 can also be the USART Asynchronous Transmit or Synchronous Clock.
RC7/RX/DT	18	I/O	ST	RC7 can also be the USART Asynchronous Receive or Synchronous Data.
VSS	8,19	P	—	Ground reference for logic and I/O pins.
VDD	20	P	—	Positive supply for logic and I/O pins.

*Si conclude dalla pagina precedente*

Legenda:

I = input

O = output

— = Not used

I/O = input/output

TTL = TTL input

P = power

ST = Schmitt Trigger input

DIP Pin#: Rappresenta il numero del pin che assolve a tale funzione nel caso in cui il case sia del tipo DIP (dual in-line package)

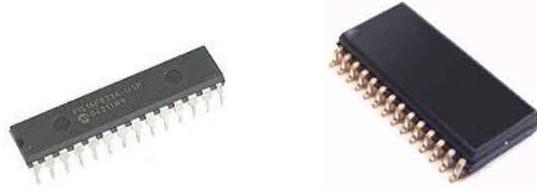
**Nota 1:** Questo buffer è un ingresso a Trigger di Schmitt quando configurato come interrupt esterno.

**Nota 2:** Questo buffer è un ingresso a Trigger di Schmitt quando usato nella modalità di programmazione in serie.

**Nota 3:** Questo buffer è un ingresso a Trigger di Schmitt quando configurato con oscillatore RC, altrimenti è un'ingresso CMOS.

A differenza della tabella dei datasheet non presenta la numerazione dei piedini per il case di tipo SOIC, ma poichè noi utilizzeremo il case di tipo DIP è stata rimossa la colonna in questione, vedi 3.2.

Come si può notare vi sono 3 pin d'alimentazione, l'8 e il 19 vanno collegati a massa e costituiscono il riferimento di massa dell'integrato, mentre il pin 20 è il riferimento per la tensione positiva e va posto a 5V. Oltre a questi 3 pin dedicati esclusivamente all'alimentazione ne sono presenti solo altri 3 che non fanno parte di un banco di ingressi/uscite e sono i pin 1,9 e 10. Il primo di questi tre è il denominato  $\overline{MCLR}/V_{pp}$ , ha la doppia funzione di bit di reset



**Figura 3.2:** Microcontrollore PIC16f873 con case di tipo DIP e SOIC

o di ingresso per la tensione d'alimentazione, mentre gli altri due sono riservati per fornire al PIC il clock di sistema.

### 3.4 Il regolatore di tensione

Come è stato spiegato nel paragrafo precedente il PIC necessita di un'alimentazione a 5V, mentre la tensione disponibile nel veicolo è quella fornita dalla batteria, ovvero 12V, risulta quindi necessario progettare un circuito in grado di abbassare la tensione a 5V. Questo circuito deve essere in grado di mantenere la tensione d'uscita costante al variare di quella d'ingresso e/o del carico in uscita; la tensione d'ingresso, infatti, in un veicolo può variare di qualche Volt nelle condizioni di normale utilizzo e anche di molti Volt in fase di avviamento, quando il motore d'avviamento elettrico assorbe un carico notevole. Per assolvere questa funzione esistono diversi circuiti integrati a tre piedini, estremamente compatti che sono in grado di fornire in uscita una tensione molto stabile. Un'integrato di questo tipo, pensato anche per uso su veicoli è l'LM2940T-5.0; questo dispositivo, infatti, oltre a fornire una tensione d'uscita molto stabile e con basso dropout<sup>1</sup>. E' protetto dai cortocircuiti, dall'inversione di polarità ed è dotato di una protezione termica in caso di surriscaldamento del circuito; inoltre, dispone anche di una protezione durante i transitori di linea, quali ad esempio il superamento momentaneo della massima tensione d'ingresso, che spegne il dispositivo a protezione del circuito e del carico. Questo circuito integrato è prodotto in vari package a seconda del variare di alcune caratteristiche quali tensione o corrente d'uscita. Nel nostro caso è stato montato un dispositivo con package TO220, molto compatto e con metodo di fissaggio PTH, caratteristica importante, in quanto la scheda sarà assemblata manualmente e quindi i dispositivi SMD non risultano adatti a tale scopo.

Questo dispositivo necessita di due condensatori, uno d'ingresso e uno d'uscita. Quello d'ingresso è consigliato nel caso in cui il regolatore sia distante dalla fonte d'alimentazione, com'è nel caso di un veicolo, mentre quello d'uscita deve essere superiore a 22μF per garantire la stabilità della tensione d'uscita. Non è necessario connettere altri dispositivi al circuito integrato per garantirne il funzionamento.

Questa tipologia di dispositivi dissipa, sotto forma di calore, una certa quantità di energia proporzionale alla quantità di potenza che è richiesta in uscita. Nel datasheet si può trovare una formula che lega la potenza massima che è possibile richiedere in uscita in relazione alla temperatura ambiente, alla massima temperatura che può raggiungere la giunzione e al valore della resistenza termica tra giunzione e ambiente.

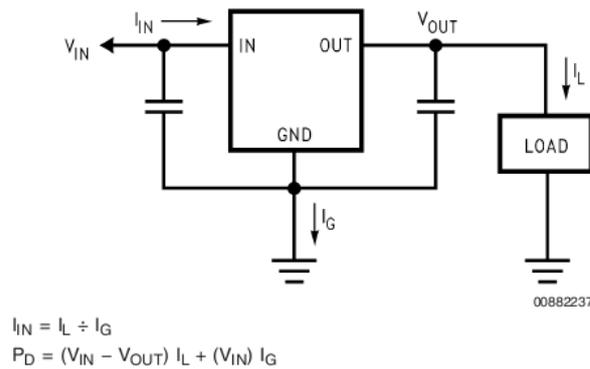
Nella figura 3.3 è rappresentata una schematizzazione del circuito nella sua applicazione tipica con i due condensatori di filtro consigliati nel datasheet. Sono inoltre indicate le correnti circolanti e le formule che legano correnti e tensioni con la potenza dissipata dal circuito. Questa potenza ha un limite massimo, definito come:

$$P_D(max) = \frac{T_R(max)}{\theta_{(JA)}} \quad (3.1)$$

dove  $\theta_{(JA)}$  rappresenta il coefficiente di resistenza termica giunzione-ambiente,  $T_R(max)$  un coefficiente di temperatura e  $P_D$ , appunto, la massima potenza che il circuito deve dissipare. A sua volta  $T_R(max)$  viene definita come:

$$T_R(max) = T_J(max) - T_A(max) \quad (3.2)$$

<sup>1</sup>Differenza di tensione tra l'ingresso e l'uscita del regolatore, ovvero è in grado di fornire 5V stabili in uscita con una tensione d'ingresso di poco superiore



**Figura 3.3:** Schema elettrico dell'utilizzo tipico dell'integrato LM2940

dove  $T_J(max)$  rappresenta la temperatura massima che può raggiungere la giunzione all'interno del dispositivo e  $T_A(max)$  la temperatura dell'ambiente circostante.

Il package TO220 ha un coefficiente  $\theta_{(JA)}$  che consente l'utilizzo del solo LM2940 privo di dissipatori di calore se questo è maggiore di  $53^\circ\text{C}/\text{W}$ . Nel caso questo valore sia inferiore è consigliato l'utilizzo di un dissipatore che, favorendo lo smaltimento di calore, evita che la giunzione raggiunga temperature critiche e che quindi intervenga il blocco termico che interrompe l'erogazione di potenza elettrica.

A questo punto proviamo a calcolare  $\theta_{(JA)}$ . Dai datasheet si può constatare che la massima temperatura che la giunzione può raggiungere è di  $150^\circ\text{C}$ , mentre la massima potenza che il PIC può assorbire è di  $1\text{W}$ . Tuttavia, quest'ultimo dato tiene conto delle peggiori condizioni assolute di funzionamento del PIC, considerando ad esempio tutte le porte in condizione di massimo assorbimento, situazione che nel nostro caso non si può verificare visto che il numero di ingressi e uscite sono solo una parte di quelle disponibili. Ci poniamo tuttavia in tale condizione proprio per garantire il corretto funzionamento nelle condizioni più estreme, ponendoci a temperatura ambiente pari a  $25^\circ\text{C}$  e trovando  $\theta_{(JA)}$ :

$$T_R(max) = T_J(max) - T_A(max) = 150^\circ\text{C} - 25^\circ\text{C} = 125^\circ\text{C} \quad (3.3)$$

$$\theta_{(JA)} = \frac{T_R(max)}{P_D(max)} = \frac{125^\circ\text{C}}{1\text{W}} = 125^\circ\text{C}/\text{W} \quad (3.4)$$

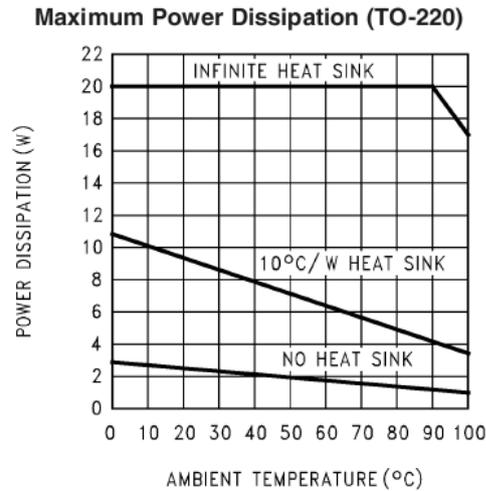
Questo stabilisce che a temperatura ambiente non vi sono problemi di dissipazione del calore; proviamo ora a ricavare la massima temperatura ammessa imponendo la condizione limite di  $\theta_{(JA)}$

$$T_R(max) = \frac{\theta_{(JA)}}{P_D(max)} = \frac{53^\circ\text{C}/\text{W}}{1\text{W}} = 53^\circ\text{C} \quad (3.5)$$

$$T_A(max) = T_J(max) - T_R(max) = 150^\circ\text{C} - 53^\circ\text{C} = 97^\circ\text{C} \quad (3.6)$$

Questo indica che anche nelle condizioni di massimo assorbimento teorico, condizione non raggiungibile in questa applicazione, il regolatore di tensione risulta funzionare in modo stabile con temperature ambiente fino a  $97^\circ\text{C}$ . Considerando che la massima temperatura alla quale l'integrato è in grado di funzionare in modo stabile avendo una temperatura ambiente pari a quella della giunzione (o equivalentemente utilizzando un dissipatore di calore ideale) è di  $125^\circ\text{C}$  il dato ricavato della formula precedente risulta molto buono. Nel grafico 3.4 sono rappresentate delle curve che si possono tracciare al variare della temperatura e della massima potenza dissipabile dall'integrato utilizzando le formule precedenti, questo sia con l'integrato privo di dissipatori, sia con un dissipatore pari a  $10^\circ\text{C}/\text{W}$  e sia con un dissipatore ideale.

Il dato ricavato dalla formula 3.6 risulta da un certo punto di vista confortante in quanto temperature del genere sono non sono raggiungibili normalmente nell'ambiente, tuttavia va considerato che una normale automobile se ferma al sole in una calda giornata estiva all'interno dell'abitacolo può raggiungere temperature anche molto elevate. Queste, in ogni caso, sono condizioni particolari in quanto l'abitacolo chiuso accumula l'energia del sole che filtra attraverso i vetri senza avere poi la possibilità di uscire, quello che avviene in una comune serra. Va dunque posta una minima attenzione al corretto posizionamento della scheda in vettura, ma

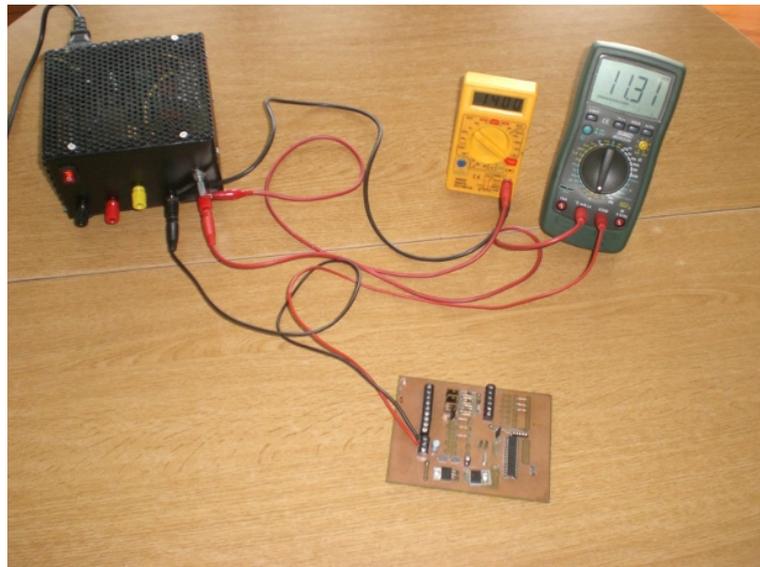


**Figura 3.4:** Curve di dissipazione del calore in funzione di temperatura e potenza dissipata

essendo la monoposto in questione priva di superfici trasparenti o totalmente chiuse il problema risulta essere di facile soluzione, mancando di per se le condizioni necessarie a creare un'effetto serra. Misure sperimentali dell'assorbimento complessivo del circuito dimostrano, tuttavia, che l'assorbimento di potenza dell'intera scheda di controllo in condizione di operatività normale (cioè senza commutazione di ingressi o uscite ed a una temperatura ambientale di circa 24°C) è di gran lunga inferiore alle condizioni limite, con un'assorbimento totale del circuito pari a 11,31mA con un'alimentazione di 14V, tensione simile a quella di un veicolo che normalmente è un pò superiore ai 12V. Questo significa che l'assorbimento di potenza complessivo è pari a

$$P = V * I = 14V * 11,31 * 10^{-3}A \cong 0,16W \quad (3.7)$$

questi dati rilevano che in realtà non vi sono problemi di dissipazione di calore fino a temperature ambientali con valori molto prossimi a 125°C.



**Figura 3.5:** Prova sperimentale del consumo della scheda

### 3.5 L'interfacciamento

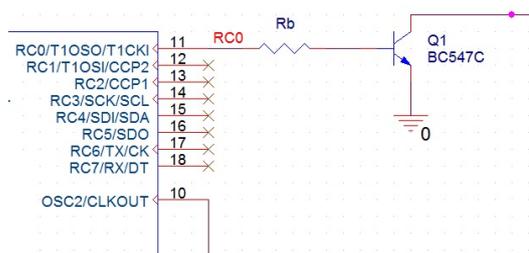
Come esposto in 3.1 e in 3.2 sono necessari tre ingressi e tre uscite, ora ci occuperemo di come interfacciare queste con il PIC iniziando dagli ingressi e ricordando che questi sono

digitali e in particolare costituiti da interruttori. Per poter fornire in ingresso lo stato logico sono possibili due soluzioni, o lasciare l'ingresso normalmente a "0" logico per poi, una volta commutato l'interruttore corrispondente, fornirgli un "1" logico oppure il contrario. Entrambe queste reti prevedono l'utilizzo di una circuiteria esterna in modo da garantire che l'ingresso sia sempre collegato a uno dei due potenziali di riferimento, evitando che questo sia influenzato da interferenze esterne. Queste reti sono chiamate rispettivamente reti di pull-down e pull-up.

Per fortuna per eseguire queste operazioni ci viene in aiuto direttamente il PIC che mette a disposizione, su PORTB, una rete interna di pull-up attivabile via software che garantisce un'ingresso normalmente alto evitando la realizzazione di una rete esterna. Lo stato logico diventerà basso solo su imposizione della circuiteria esterna, si può pertanto pensare di collegare direttamente a 3 pin del PORTB tre interruttori da un lato collegati direttamente al rispettivo pin e dall'alto a massa, cosicché a interruttore chiuso l'ingresso corrispondente risulterà a "0" logico. Questa circuiteria tuttavia presenta il problema che alla commutazione di un'interruttore si possono innescare oscillazioni dell'ingresso, falsando quindi la lettura dello stato logico. Per ovviare a questa problematica è possibile, ad esempio, inserire una rete R-C su ogni linea d'ingresso in modo da rendere meno brusche le variazioni di stato, evitando errori nella lettura dell'ingresso.

Per quanto riguarda le linee d'uscita la cosa non è così semplice, innanzitutto possiamo distinguere le uscite in due gruppi, uno "di potenza" con lo scopo di pilotare l'elettrovalvola che necessita di un certo livello di potenza d'uscita e uno "di segnale", con il compito di inviare il segnale di cut-off alla centralina motore e che richiede un minimo dispendio di energia. Poiché le uscite del PIC hanno potenze d'uscita molto modeste è opportuno che vengano amplificate in modo da evitare il danneggiamento del PIC o l'incapacità dello stesso di pilotare nel modo corretto le uscite. Questo ovviamente è indispensabile nel caso delle uscite atte a pilotare l'elettrovalvola, mentre nel caso dell'uscita di cut-off cioè non è necessario in quanto l'ingresso della centralina motore richiede una modestissima corrente per pilotarne l'ingresso, tuttavia in questo caso si presenta un'altro problema. Il pin della centralina infatti, da specifiche, deve essere lasciato normalmente volante e posto a massa quanto si desidera far partire il tempo di cut-off; forzare una tensione diversa da massa non solo non garantisce il corretto riconoscimento del successivo impulso di cut-off, ma può anche danneggiare la centralina. Per questa ragione anche in questo caso è necessario predisporre un'opportuna logica tra il pin e l'uscita. L'idea più semplice è quella utilizzare un transistor pilotato dal pin del PIC che a sua volta è collegato all'uscita vera e propria.

Nella figura di seguito riporto la schematizzazione dell'uscita "di segnale":

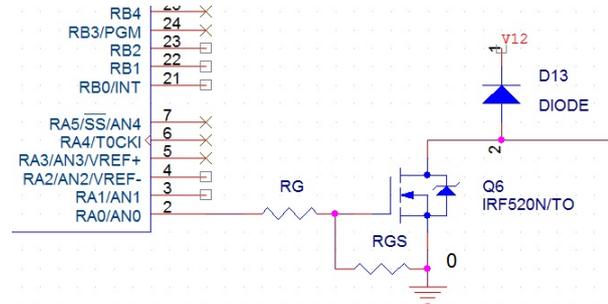


Per questa uscita è stato utilizzato un transistor bipolare di tipo npn, la resistenza posta in serie alla base del transistor serve per limitare la corrente di base, evitando che questa raggiunga valori troppo elevati. Questa resistenza, oltre a limitare la corrente, deve anche garantire che il transistor in questione funzioni nel punto operativo voluto, ovvero in saturazione. Secondo il datasheet questa condizione è soddisfatta se la tensione  $V_{be}$  è compresa tra 0,7 e 0,9V; in questa condizione il guadagno di corrente tra  $I_B$  e  $I_C$  è pari a 20. La corrente minima di collettore per mantenere il transistor in zona di saturazione deve essere almeno di 10mA, mentre invece 100mA è la corrente massima che le uscite del PIC sono in grado di fornire, è quindi necessario imporre una corrente di base compresa tra 0,5 e 5mA. Da queste considerazioni deduciamo che:

$$R_{Min} = \frac{V_{con} - V_{be\min}}{I_{B\max}} = \frac{5V - 0,9V}{5mA} = 820\Omega \quad (3.8)$$

$$R_{MAX} = \frac{V_{con} - V_{beMAX}}{I_{Bmin}} = \frac{5V - 0,7V}{0,5mA} = 8600\Omega \quad (3.9)$$

A questo punto basta scegliere una resistenza che stia all'interno di questo intervallo, possibilmente a buona distanza da entrambi i valori limite, evitando da un lato la zona lineare e dall'altro il breakdown del dispositivo. Un valore consono, considerando che si trovano in commercio sono standardizzati, può essere  $4700\Omega$ . Ora passiamo all'uscita di potenza e cominciamo con l'introdurre lo schema relativo:



Per questa uscita è stato utilizzato un MOSFET di potenza, con caratteristiche adatte a pilotare l'elettrovalvola considerando questa, dal punto di vista elettrico, rappresentabile come un'induttanza in serie ad una resistenza. Non è dato sapere il valore dei singoli componenti in questione in quanto nel datasheet del componente non vengono riportati, ma ci basta sapere che il consumo di potenza è pari a 1W per ogni controllo con un'alimentazione pari a 12V. Con un semplice calcolo è possibile calcolare la corrente assorbita a regime:

$$I = \frac{P}{V} = \frac{1}{12} \cong 0,083A \quad (3.10)$$

Questo calcolo è valido solamente a regime, ma poiché sappiamo dalla fisica dei dispositivi che il massimo valore di corrente in un circuito L-R serie si presenta in condizioni di regime, con la caduta ai capi dell'induttore nulla, questo valore non verrà superato in fase di transitorio, ma verrà invece raggiunto gradualmente. Il transistor BJT precedente non risulta adatto a tale scopo, in primo luogo perché gli 83mA del calcolo precedente risultano molto vicini ai 100 massimi ammessi dal dispositivo, in secondo luogo in quanto la corrente di collettore è proporzionale, come in tutti i BJT, a quella di base. Risulta quindi necessaria una corrente non irrilevante da parte del pin d'uscita del PIC, per questa ragione è preferibile utilizzare un transistor di tipo MOS.

Dallo schema precedente si può notare che il drain del MOS è collegato sia all'elettrovalvola, collegata dall'altro capo ai 12V della batteria, sia ad un diodo, anch'esso connesso all'altro capo a 12V. Questo in quanto, a transistor acceso e in condizioni di regime, se viene aperto il circuito, l'induttanza presente nell'elettrovalvola continua a forzare una corrente nel circuito che se non gestita può provocare un forte innalzamento della tensione ai capi del MOS, con la possibilità di rottura dello stesso. Questo diodo invece limita la tensione del drain a circa 12,7V (12V più la tensione di soglia del diodo) fornendo una via di richiusura della sovracorrente dell'induttanza verso l'alimentazione. Per quanto riguarda invece il diodo posto tra source e drain non si tratta di un componente esterno, ma semplicemente del diodo formato dalla giunzione pn tra bulk e drain e, poiché il bulk è cortocircuitato con il source, il diodo risulta presente tra bulk e drain.

A questo punto possiamo dimensionare le due resistenze rappresentate nello schema. La resistenza di gate serve per limitare la corrente di carica della capacità di gate. In fase di accensione del transistor, infatti, il potenziale di gate si ritrova a massa e se non ci fosse una resistenza a limitare la corrente circolante il PIC si troverebbe a dover erogare un'elevatissimo picco di corrente con il risultato quantomeno di danneggiare lo stadio di uscita della porta in questione. La resistenza da inserire deve quindi garantire che la corrente richiesta, nel caso peggiore, non superi la corrente massima che il PIC può fornire, tuttavia va evitata una resistenza troppo grande per non allungare troppo i tempi di attivazione del transistor. Da un semplice calcolo  $R_{Gmin}$  risulta essere:

$$R_{Gmin} = \frac{V_{RA0ON} - V_{Gmin}}{I_{RA0MAX}} = \frac{5V - 0V}{100mA} = 50\Omega \quad (3.11)$$

A questo punto il margine di scelta è molto ampio, è sufficiente scegliere un valore non prossimo al minimo, ma non troppo grande, nel nostro caso tale resistenza è stata fissata a  $150\Omega$ . A questo punto non rimane che dimensionare la resistenza tra gate e source. Per far comprendere l'utilità di questa resistenza ricordiamo che il gate è di fatto un condensatore, quindi una volta fissato un potenziale per poterlo cambiare è necessario imporvi una certa corrente, diversamente il transistor permane nella stato in cui si trova. Nel nostro caso una volta che il transistor è acceso e risulta necessario spegnerlo la capacità di gate viene scaricata tramite il pin d'uscita del PIC, tuttavia questo è costituito da un'architettura di tipo TTL, che non è in grado di portare il potenziale d'uscita fino a  $0V$ , ma permane un valore minimo di tensione. Questa tensione residua può dar luogo ad un spegnimento solo parziale del transistor di potenza, è dunque opportuno porvi rimedio. La soluzione più semplice è quella di collegare il gate a massa tramite un'opportuna resistenza che scaricherà il condensatore di gate fino allo spegnimento del transistor. Va sottolineato che tale resistenza non può essere di un valore eccessivamente basso, altrimenti, ponendosi in serie alla resistenza di gate, creerà un partitore resistivo quando l'uscita del PIC è posta a "1" logico, limitando la tensione che il gate riuscirà a raggiungere. Risulta quindi opportuno che questa resistenza sia di un valore notevolmente superiore alla  $R_G$  precedentemente fissata a  $150\Omega$ . La resistenza in questione è stata fissata a  $470k\Omega$ , un valore così alto non comprometterà il valore massimo che la tensione di gate potrà raggiungere  $3.12$  pur assicurando un corretto spegnimento del transistor di potenza.

$$V_{GMAX} = V_{A0on} - V_{RG} = V_{A0on} - \frac{V_{A0on}}{R_G + R_{GS}} R_{GS} = 5V - \frac{5V}{470150\Omega} 470000\Omega \cong 4,998V \quad (3.12)$$

La problematica che si è presentata nel caso dei MOS non risulta valida per i BJT in quanto, essendo pilotati in corrente, non soffrono il problema della tensione residua sulla base in quanto questa da origine a una corrente tra base ed emettitore che comporta un'azzeramento della tensione stessa e uno spegnimento totale del transistor.

### 3.6 Lo schema elettrico

A questo punto introduciamo lo schema elettrico completo della scheda 3.2, trattando ciò che non è stato ancora analizzato. Nello schema di riferimento possiamo notare che l'alimentazione dell'elettrovalvola passa per la scheda cambio attraverso due diodi che hanno lo scopo di proteggere l'intera scheda contro inversioni di polarità e di impedire ritorni di corrente verso l'alimentazione. Per limitare al minimo la caduta di tensione ai capi dei diodi si sono scelti due diodi shottky, caratterizzati da una bassa caduta di potenziale in diretta (circa  $0,3V$ ) e da un'alta velocità di commutazione, scongiurando picchi di corrente. Questa tensione rettificata dai diodi viene chiamata  $V_{12}$ , ed è a questa tensione che vanno collegati i diodi di protezione dei transistor di potenza in modo da creare una rete chiusa di ricircolo della corrente, senza disturbare l'intera alimentazione con picchi di tensione e corrente. Sempre riguardo all'alimentazione si può notare che c'è un condensatore  $C7$  posto in corrispondenza del pin 20 d'alimentazione del PIC. Questo ha lo scopo di filtrare ulteriormente la tensione d'alimentazione in prossimità dell'integrato in modo che eventuali componenti oscillatorie che si possono essere generate lungo la pista che porta dal regolatore di tensione al PIC vengano smorzate.

Oltre a questo possiamo notare la rete generatrice della frequenza di clock, con "crystal" che rappresenta il quarzo e i due condensatori  $C5$  e  $C6$ ; la rete è stata realizzata seguendo le specifiche riportate nel datasheet.

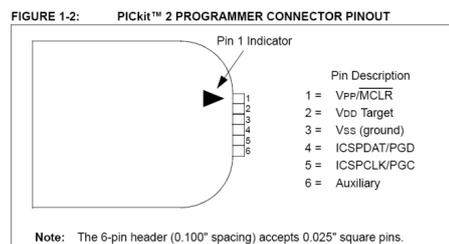
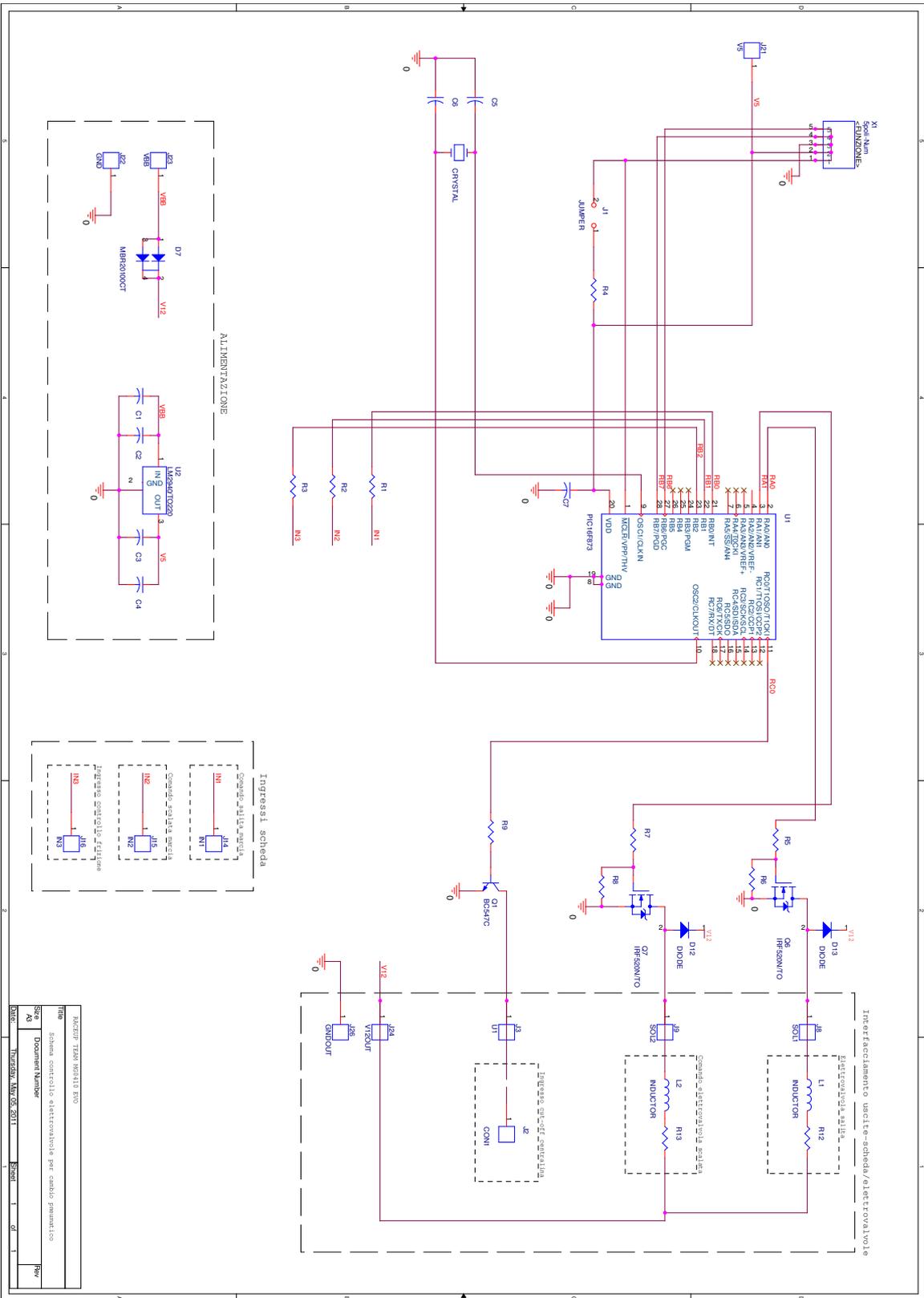


Figura 3.6: Connettore del programmatore con la descrizione di ogni pin

Infine si può notare il connettore X1 che serve per programmare il PIC, a questo va infatti collegato il programmatore che ha il compito di caricare il programma nel microcontrollore. Seguendo le specifiche del programmatore riportate in figura 3.6 si è quindi costruito il connettore X1. Per far questo, tuttavia, si è reso necessario interporre tra il pin 1 del PIC e l'alimentazione un jumper con lo scopo di scollegare l'alimentazione dal pin 1 in fase di programmazione e garantirla invece mentre il PIC sta operando, disabilitando il reset. La resistenza R4 ha lo scopo di prevenire il danneggiamento del PIC nel caso in cui venga collegato il programmatore senza preventivamente aver tolto il jumper, limitando il passaggio di corrente. Tale resistenza è stata fissata ad un valore di  $10k\Omega$ .

Tabella 3.2: Schema elettrico completo della scheda



# Capitolo 4

## Studio del sistema software

In questo capitolo analizzeremo il linguaggio di programmazione, l'algoritmo e il programma di sviluppo del codice che sono stati utilizzati per la realizzazione della parte software della scheda cambio.

### 4.1 Il linguaggio

Un microcontrollore può essere programmato sia con un linguaggio di programmazione ad alto livello sia a basso livello. Questa distinzione si basa sulla vicinanza che ha un linguaggio al modo di operare dell'uomo o della macchina. Più il livello è definito alto più il linguaggio è "vicino" all'uomo, più è basso e più è "vicino" alla macchina.

**Linguaggio macchina** Il linguaggio macchina è l'unico linguaggio che è in grado di comprendere una qualsiasi unità di elaborazione dati e consiste in una sequenza di cifre binarie per ciascuna delle quali è associata una determinata operazione. Questo tipo di linguaggio è stato utilizzato solo agli albori dell'era informatica ma ben presto, con il complicarsi dei programmi, si è reso necessario creare dei linguaggi che semplificassero il compito dei programmatori, evitando che questi si dovessero ricordare le sequenze di bit associate ad ogni istruzione.

**Linguaggi a basso livello** I linguaggi a basso livello sono il tipo di linguaggi più vicini alla macchina dopo il linguaggio macchina ed hanno la caratteristica per cui ciascuna istruzione macchina è associata ad un codice mnemonico semplice da ricordare invece che da un numero, c'è quindi una corrispondenza diretta tra ciascuna istruzione ed una particolare istruzione macchina. Questi linguaggi, essendo molto vicini a quello macchina, danno la possibilità di essere a conoscenza in ogni singolo ciclo macchina del contenuto di ciascun registro. Grazie a questa particolarità è possibile ottimizzare al massimo il codice, consentendo di eseguirlo in tempi molto brevi e senza sprechi di memoria, avendo la libertà di gestirla a piacimento e quindi al meglio. Questi vantaggi hanno per contro un'importante svantaggio, il linguaggio a basso livello infatti è estremamente inadatto allo sviluppo di codice complesso, per eseguire anche operazioni molto semplici è appunto necessario scrivere diverse righe di codice e questo limita notevolmente la complessità che un programma a basso livello può raggiungere.

**Linguaggi ad alto livello** I linguaggi ad alto livello sono molto più simili al modo di pensare della mente umana, quindi consentono di poter scrivere dei programmi con algoritmi molto complessi in poche righe di codice e utilizzando un set di istruzioni estremamente semplici da ricordare in quanto molto più simili, appunto, ad un linguaggio piuttosto che ad una sequenza di numeri. Oltre a questo vantaggio questi linguaggi offrono la possibilità di scrivere del codice che può essere eseguito da qualsiasi processore. Questo è possibile in quanto non è più il programmatore a poter decidere come gestire i vari registri, ma sarà un'apposito programma che convertirà il codice scritto con linguaggio ad alto livello in linguaggio macchina.

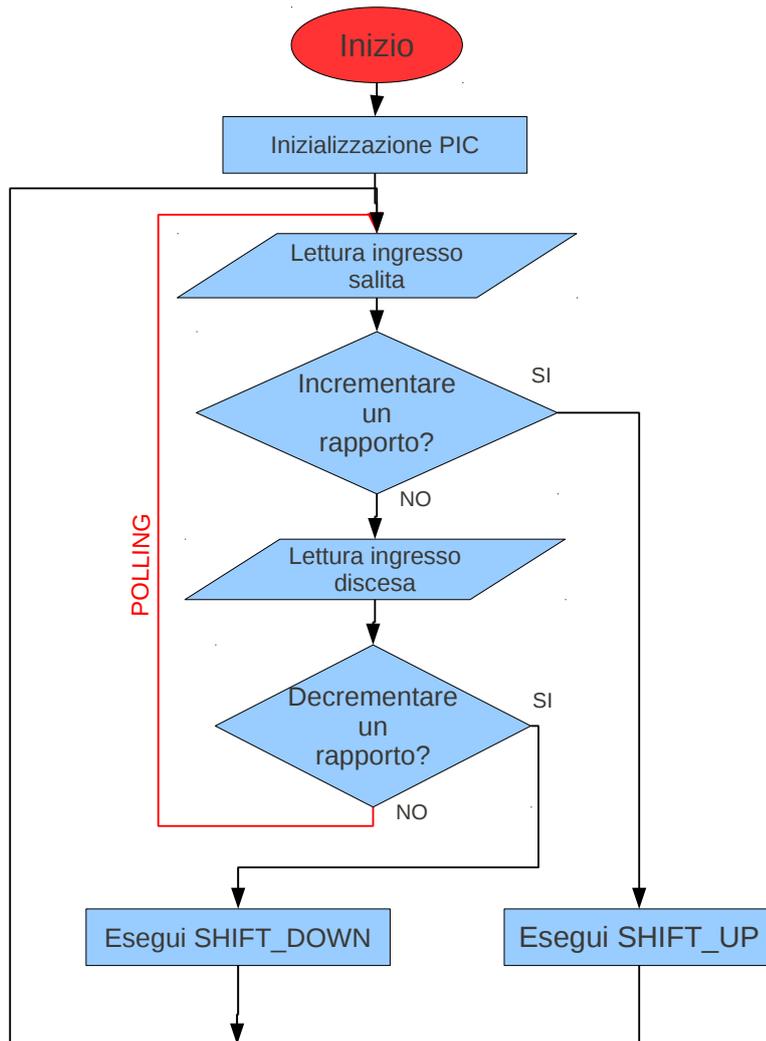
Com'è possibile osservare ciascun livello di linguaggio presenta dei vantaggi e degli svantaggi. Nel caso in questione è stato deciso di utilizzare un linguaggio a basso livello in quanto

l'algoritmo del programma è molto semplice, quindi è possibile sfruttare i vantaggi dei linguaggi a basso livello senza risentire troppo degli svantaggi.

## 4.2 Algoritmo

Prima di qualsiasi altra cosa è indispensabile a questo punto, prima ancora di cominciare a vedere il codice, presentare e descrivere l'algoritmo del programma.

La strada più semplice per far ciò è rappresentarlo tramite un flow-chart<sup>1</sup>:



**Figura 4.1:** Flow-chart generale del programma

Il diagramma di figura 4.1 rappresenta l'algoritmo generale del programma. In prima istanza lo analizzeremo in modo generico e in seguito descriveremo ciascun blocco nel dettaglio, scomponendolo nei vari sottoblocchi.

Per prima cosa si può notare che manca un blocco di fine programma e questo significa che il programma è destinato, visto che non ci sono blocchi “ciechi”, a non terminare mai continuando a ciclare fino a quando non viene tolta l'alimentazione. Un'altra cosa che salta subito all'occhio è il ciclo denominato “polling” e posto in risalto dalla freccia di colore rosso. Questo ciclo risulta infatti essere la condizione in cui si trova normalmente ad operare il nostro PIC, ovvero

<sup>1</sup>I flow-chart o diagrammi a blocchi o diagrammi di flusso costituiscono un linguaggio di modellazione grafico per rappresentare algoritmi. In particolare in campo informatico vengono utilizzati per rappresentare algoritmi software senza doverli relazionare ad alcun linguaggio di programmazione.

un continuo ciclare in attesa che una delle uscite comandate dal pilota subisca una variazione. A seguito di questa prima analisi procediamo descrivendo tutti i blocchi.

### Inizio

Il blocco Inizio è il primo che si incontra analizzando il flow-chart ed è l'unico a non avere frecce entranti, rappresenta infatti la condizione di inizio di esecuzione dell'algoritmo, può essere pertanto associata all'istante in cui il microcontrollore riceve l'alimentazione.

### Inizializzazione PIC

Questo blocco rappresenta tutte quelle operazioni che esegue il microcontrollore prima di poter diventare pienamente operativo, alcune sono implicite ed eseguite automaticamente dal PIC, mentre altre sono impostabili via software.

### Lettura ingresso salita

In questo blocco viene letto l'ingresso che rappresenta la richiesta di salita di marcia, rilevando le volontà del pilota in tal senso.

### Incrementare un rapporto?

Questo blocco rappresenta una scelta che deve effettuare il programma in base alla condizione dell'ingresso "salita di marcia" acquisito nel blocco precedente. Nel caso il programma decida di incrementare il rapporto l'algoritmo procede con il blocco "*Esegui SHIFT\_UP*" mentre nel caso in cui decida di non incrementare il rapporto allora passa al blocco "*Lettura ingresso discesa*"

### Lettura ingresso discesa

Similmente a quanto avviene in "*Lettura ingresso salita*" in questo blocco viene letto un ingresso che stavolta però rappresenta l'eventuale richiesta di discesa di marcia.

### Decrementare un rapporto?

Anche questo blocco è simile al precedente "*Incrementare un rapporto?*" con la differenza che il programma sceglie in base alla condizione rilevata dal blocco "*Lettura ingresso discesa*" e decide se ritornare a "*Lettura ingresso salita*" chiudendo il ciclo di polling oppure a "*Esegui SHIFT\_DOWN*"

### Esegui SHIFT\_UP ed Esegui SHIFT\_DOWN

Questi due blocchi si occupano dell'effettiva esecuzione, dopo aver avuto conferma dai due blocchi di scelta, rispettivamente dell'incremento/decremento del rapporto modificando le uscite del PIC dopo aver controllato che siano rispettate tutte le condizioni di sicurezza. Terminato tale blocco che, come vedremo, non necessariamente modifica le uscite, si riparte dall'esecuzione del blocco "*Lettura ingresso salita*"

Come si può notare l'algoritmo di base è semplice, consiste in un ciclo principale denominato di polling che ripetitivamente continua a controllare gli ingressi comandati al pilota fino a quando uno di questi non viene premuto. Quando si verifica tale condizione, a seconda di quale interruttore viene premuto, si esegue un blocco di istruzioni che verificano la possibilità di eseguire la cambiata e agiscono di conseguenza per poi ritornare nuovamente all'inizio del ciclo di polling in attesa di altre richieste del pilota. Va specificato che i cicli di polling in alcuni casi sono fortemente sconsigliati in quanto il processore resta in questi cicli ad eseguire sempre le stesse istruzioni fino al verificarsi di una condizione specifica, bloccando di fatto l'esecuzione della normale esecuzione del programma e rendendolo insensibile a qualsiasi condizione se non quella che lo fa uscire dal polling. In questo caso, tuttavia, questo non risulta un problema in quanto i due ingressi che condizionano l'intera esecuzione dell'algoritmo sono proprio le

condizioni d'uscita dal polling, inoltre, essendo le operazioni contenute nel ciclo di polling molto limitate, la velocità con il quale il controllore è in grado di reagire a una variazione di un'ingresso risulta estremamente alta.

### 4.2.1 Inizializzazione PIC

Come specificato in precedenza il blocco *“Inizio”* non ha nessuna funzione se non quella di segnalare da dove inizia l'esecuzione dell'algoritmo, quindi cominciamo direttamente dalla descrizione del blocco *“Inizializzazione PIC”*.

Questo blocco può essere suddiviso in due blocchi principali, uno caratterizzato da tutte quelle operazioni che il PIC effettua in maniera automatica non appena viene alimentato e che lo rendono in grado di poter eseguire il programma che vi è caricato. Queste operazioni si differenziano a seconda del PIC in questione, ma in genere vi sono alcune operazioni comuni che li legano tutti, quali ad esempio la gestione delle varie memorie, l'inizializzazione del program counter e di alcuni registri, il caricamento del programma, ecc. Queste operazioni tuttavia non ci interessano in quanto sono indispensabili, non condizionabili e, in ogni caso, non influenti sull'algoritmo.

Passiamo dunque a quelle impostazioni che sono fornite dall'utente per configurare nel miglior modo possibile il PIC in relazione all'uso che se ne deve fare. Iniziamo dunque con il rappresentare graficamente queste operazioni viste come un'ampliamento del blocco *“Inizializzazione PIC”* del flow-chart generale.

Come si può notare in figura 4.2 il flow-chart è composto da una serie di blocchi atti a configurare vari registri in modo opportuno. Poiché non sono presenti blocchi di scelta queste configurazioni sono sempre le stesse per ogni ciclo di funzionamento del PIC, ovvero ogni volta che viene acceso. Una parte di queste istruzioni è contenuta in una subroutine chiamata INIZ, proprio a simboleggiare la fase di inizializzazione del PIC e nel diagramma è posta in risalto dal colore arancione. All'interno di questa subroutine sono presenti le configurazioni dei registri denominati TRISA, TRISB e TRISC che, come vedremo meglio in seguito, configurano gli ingressi e le uscite del PIC i quali, avendo una forte caratteristica comune, sono stati posti in risalto da una colorazione gialla.

### Configurazione `__CONFIG`

I PIC dispongono di un registro di configurazione hardware che viene scritto una sola volta al momento della programmazione e che stabilisce il funzionamento di alcuni circuiti interni come il watch dog timer (wdt) e l'oscillatore di clock; questa operazione è conosciuta anche con il nome di settaggio dei fuses. Nei micro non si tratta naturalmente di fusibili veri i propri, ma semplicemente di impostazioni che possono essere cambiate riprogrammando tale registro. Ogni programma per PIC inizia con una intestazione in cui si dichiara, oltre al tipo di micro usato e al formato di default dei numeri (decimale, esadecimale ecc...), anche la configuration word che ne determinerà il funzionamento (specificata con `__CONFIG`):

---

<code>CP0</code>	<code>CP1</code>	<code>DEBUG</code>	<code>-</code>	<code>CPD</code>	<code>LVP</code>	<code>BODEN</code>	<code>CP1</code>	<code>CP0</code>	<code><math>\overline{PWRITE}</math></code>	<code>WDTE</code>	<code>FOSC1</code>	<code>FOSC0</code>
bit 13											bit 0	

bit 13:12 - 5:4 **CP1:CP0** bits del codice di protezione della memoria programma FLASH

- 11 = codice di protezione off
- 10 = codice protetto da 0F00h a 0FFFh
- 01 = codice protetto da 0800h a 0FFFh
- 00 = codice protetto da 0000h a 0FFFh

bit 11 **DEBUG** Modalità debugger sul circuito

1 = Debugger sul circuito disabilitato, i pin RB6 and RB7 sono are ingressi e uscite ad uso generale.

0 = Debugger sul circuito abilitato, i pin RB6 and RB7 sono dedicati al debugger.

bit 10 **Unimplemented**: Letto come “1”

bit 9 **WRT**: Abilitazione alla scrittura della memoria programma FLASH

1 = La memoria programma può essere scritta con il controllo EECON

0 = La memoria programma non può essere scritta con il controllo EECON

bit 8 **CPD**: Codice di protezione della memoria dati EE

1 = Codice di protezione off

0 = Codice della memoria dati EEPROM protetto

bit 7 **LVP**: bit di abilitazione di programmazione in bassa tensione

1 = il pin RB3/PGM ha funzione PGM, la programmazione in bassa tensione è abilitata

0 = il RB3 è un'I/O digitale, HV su MCLR deve essere utilizzato per la programmazione

bit 6 **BODEN**: bit di abilitazione del Brown-out Reset

1 = BOR abilitato

0 = BOR disabilitato

bit 3 **PWRT $\overline{E}$** : bit di abilitazione del timer Power-up

1 = PWRT disabilitato

0 = PWRT abilitato

bit 2 **WDTE**: bit di abilitazione del Watchdog Timer

1 = WDT abilitato

0 = WDT disabilitato

bit 1-0 **FOSC1:FOSC0**: bits di selezione dell'oscillatore

11 = RC oscillatore

10 = HS oscillatore

01 = XT oscillatore

00 = LP oscillatore

**Note 1:** Il valore non programmato o cancellato della parola di configurazione sono 3FFFh.

**Note 2:** Tutte le coppie CP1:CP0 devono assumere lo stesso valore per abilitare la lista dei codici di protezione.

**Note 3:** L'abilitazione del Brown-out Reset abilita automaticamente il Power-up Timer (PWRT), indipendentemente dal bit  $\overline{PWRT\overline{E}}$ . E' garantiva l'abilitazione del Power-up Timer ogni volta che il Brown-out Reset è abilitato.

A seguito di questa tabella possiamo dire che i codici di protezione e di abilitazione alla modifica del programma non ci interessano in nessun caso, quindi verranno disabilitati, ponendo a 1 i bit 13,12,9,8,5,4. Anche la modalità di debug del circuito non ci interessa e quindi la disabilitiamo ponendo a 1 il bit 11, mentre il bit 10 deve essere posto a 1 in ogni caso. I bit 1 e 0 configurano il tipo di oscillatore scelto, nel nostro caso si tratta di un' oscillatore al quarzo da 4 Mhz, che corrisponde ad un oscillatore di tipo “xt”, quindi il bit 1 andrà posto a 0, mentre il bit 0 va posto a 1. Il bit 7 va posto a 1 in quanto si è deciso di programmare il PIC in modalità a bassa tensione, rinunciando ad RB3 in quanto in ogni caso gli ingressi e le uscite a disposizione sono parecchie. I bit 3 e 6 prevedono il reset del circuito in condizioni

particolari, nel primo caso in fase di accensione, in modo da garantire lo stabilizzarsi della tensione d'alimentazione mantenendo il PIC in reset per un certo intervallo di tempo (circa 76ms). Mentre, nel secondo caso, quando la tensione d'alimentazione risulta inferiore ad una di riferimento (circa 4V) per un intervallo di tempo prefissato (circa 100 $\mu$ s), anche questi bit sono stati abilitati entrambi, ponendoli a 1. Per ultimo resta il bit 2 che è il watchdog timer, un particolare timer che resetta automaticamente il PIC quando raggiunge un determinato valore. L'incremento di questo timer è indipendente da quello dell'esecuzione del programma e serve per poter resettare il PIC nel caso in cui l'elaborazione del programma vada in blocco, questo particolare sistema di reset è stato abilitato ponendo il bit 2 a 1.

### Configurazione OPTION REG

Questo registro completamente leggibile e scrivibile in qualsiasi momento, da la possibilità di configurare alcuni parametri del PIC in particolare inerenti al contatore TMR0, il Watchdog Timer, l'interrupt da esterno e i pull-up della porta B.

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit7							bit0

bit 7 **RBPU**: Bit di abilitazione dei pull-up di PORTB

1 = I pull-up di PORTB sono disabilitati.

0 = I pull-up di PORTB sono abilitati dai singoli valori dei latch delle porte.

bit 6 **INTEDG**: Bit di selezione del taglio dell'interrupt.

1 = Il pin d'interrupt RB0/INT è sensibile ai fronti di salita.

0 = Il pin d'interrupt RB0/INT è sensibile ai fronti di discesa.

bit 5 **T0CS**: Bit di selezione della sorgente del clock TMR0

1 = Transizione del pin RA4/T0CKI

0 = Ciclo di clock interno (CLKOUT)

bit 4 **T0SE**: Bit di selezione del taglio di TMR0

1 = Incremento al fronte di discesa del pin RA4/T0CKI.

0 = Incremento al fronte di salita del pin RA4/T0CKI.

bit 3 **PSA**: Bit di assegnazione prescalato

1 = Assegnazione prescalata al WDT

0 = Assegnazione prescalata dal modulo del Timer0

bit 2-0 **PS2:PS0**: Bits di selezione del rapporto di prescalata

Bit Value	TMR0 Rate	WDT Rate
000	1:2	1:1
001	1:4	1:2
010	1:8	1:4
011	1:16	1:8
100	1:32	1:16
101	1:64	1:32
110	1:128	1:64
111	1:256	1:128

Nel nostro caso il bit 7 è stato posto a 0 in quanto è necessaria l'attivazione dei pull-up interni per rilevare quanto l'ingresso viene posto a massa. Il bit 6 è stato posto a 0, rendendo gli interrupt sensibili ai fronti di discesa, mentre il bit 5 è posto a 0 relazionando quindi TMR0 al clock interno. Il bit 5 è irrilevante in quanto il clock è esterno, ma comunque è stato posto anch'esso a 0. I bit 3,2,1,0 sono invece posti tutti a 1, il bit tre per segnalare che si sta impostando la "prescalata" del Watchdog Timer e i bit 2,1,0 sono invece per ritardare al massimo possibile la scadenza di questo registro, secondo la tabella sopra riportata.

### Configurazione ADCON1

Questo registro consente di impostare i pin delle porte d'ingresso come analogici o digitali. La tabella sottostante è tratta dai data-sheet del PIC, la prima colonna rappresenta le caratteristiche del bit e la seconda il nome, sotto alla tabella è specificato il *bit 0* e il *bit 7* in modo da non incorrere in errori.

U0	R/W-0	U0	U0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	ADCS2	-	-	PCFG3	PCFG2	PCFG1	PCFG0
bit7							bit0

Legenda:

U0 significa "Unimplemented" cioè non un bit non implementato, letto come 0

R = bit leggibile

W = bit scrivibile

-n = bit inizializzato al valore n al reset

'1' = bit posto a 1

'0' = bit posto a 0

'x' = bit sconosciuto

bit 7 **ADFM**: bit di selezione del formato del risultato del convertitore A/D

1 = Giustificato a destra. I 6 bit più significativi del registro ADRESH sono letti come '0'.

0 = Giustificato a sinistra. I 6 bit meno significativi del registro ADRESL sono letti come '0'.

bit 6 **ADCS2**: bit di selezione della velocità di clock di conversione del convertitore analogico digitale

ADCON1	ADCON0	Clock Conversion
0	00	Fosc/2
0	01	Fosc/8
0	10	Fosc/32
0	11	Frc
1	00	Fosc/4
1	01	Fosc/16
1	10	Fosc/64
1	11	Frc

bit 5:4 **Unimplemented**: Letto come "0"

bit 3:0 **PCFG3:PCFG0**: A/D Bit di controllo della configurazione della porta

In questo registro il bit 7 serve per il configurare il formato del risultato del convertitore, il bit 6 regola la velocità di conversione degli ingressi settati come ingressi analogici, i bit 4 e 5 non assumono alcun significato, mentre quelli che vanno dal 0 al 3 configurano la porta. Nella tabella sottostante sono rappresentate le configurazioni degli ingressi associate ad ogni combinazione dei bit 0,1,2,3.

PCFG3:	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	VREF+	VREF-	C/R
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	AN3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	AN3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	AN3	VSS	2/1
011x	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	AN3	AN2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	AN3	AN2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	AN3	AN2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	AN3	AN2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	AN3	AN2	1/2

Legenda: vedi tabella precedente.

**Nota:** In ogni dispositivo di Reset, i pin delle porte che sono dotati di multiplexer con funzionalità analogiche (ANx) sono forzati ad essere ingressi analogici.

I registri ADRESH:ADRESL contengono i 10-bit risultato della conversione A/D. Quando la conversione A/D è completa, il risultato viene caricato in questa coppia di registri, il bit GO/DONE (ADCON0 2) viene posto a 0 e il bit ADIF di flag d'interrupt A/D è posto a 1. Dopo che il modulo A/D è stato configurato come desiderato, il canale selezionato deve essere acquisito prima che la conversione sia iniziata. I canali di ingresso analogici devono avere i corrispondenti bit del registro TRIS selezionati come input.

Nel nostro caso non sono richiesti ingressi analogici, quindi il bit PCFG3 va posto a 0, mentre i bit PCFG2 e PCFG1 a 1, non è invece rilevante lo stato del bit PCFG0, come quello del bit 6.

### Configurazione CMCON

Questo registro gestisce i comparatori presenti nel PIC in base alle proprie esigenze. La seguente tabella di configurazione è tratta dai datasheet del PIC:

R0	R0	U0	U0	R/W-0	R/W-0	R/W-0	R/W-0
C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
bit7				bit0			

bit 7

**C2OUT:** Bit indicatore dell'uscita del Comparator2

Quando C2INV = 0:

1=C2 Vin+ > C2 Vin-

0=C2 Vin+ < C2 Vin-

Quando C2INV = 1:

1=C2 Vin+ < C2 Vin-

$0 = C2 \text{ Vin+} > C2 \text{ Vin-}$   
 bit 6  
**C1OUT:** Bit indicatore dell'uscita del Comparator1  
 Quando C1INV = 0:  
 $1 = C1 \text{ Vin+} > C1 \text{ Vin-}$   
 $0 = C1 \text{ Vin+} < C1 \text{ Vin-}$   
 Quando C1INV = 0:  
 $1 = C1 \text{ Vin+} > C1 \text{ Vin-}$   
 $0 = C1 \text{ Vin+} < C1 \text{ Vin-}$   
 Quando C1INV = 1:  
 $1 = C1 \text{ Vin+} < C1 \text{ Vin-}$   
 $0 = C1 \text{ Vin+} > C1 \text{ Vin-}$   
 bit 5  
**C2INV:** Bit di inversione delle uscite del comparatore 2  
 1 = C2 uscita invertita  
 0 = C2 uscita non invertita  
 bit 4  
**C1INV:** Bit di inversione delle uscite del comparatore 1  
 1 = C1 uscita invertita  
 0 = C1 uscita non invertita  
 bit 3  
**CIS:** Bit di switch d'ingresso del comparatore  
 Quando CM2:CM0:=110:  
 $1 = C1 \text{ Vin-}$  connesso ad RA3/AN3,  $C2 \text{ Vin-}$  connesso ad RA2/AN2  
 $0 = C1 \text{ Vin-}$  connesso ad RA0/AN0,  $C2 \text{ Vin-}$  connesso ad RA1/AN1  
 bit 2:0  
**CM2:CM0:** Bit di configurazione dei comparatori, devi C

Legenda:

U0 significa "Unimplemented" cioè non un bit non implementato, letto come 0

R = bit leggibile

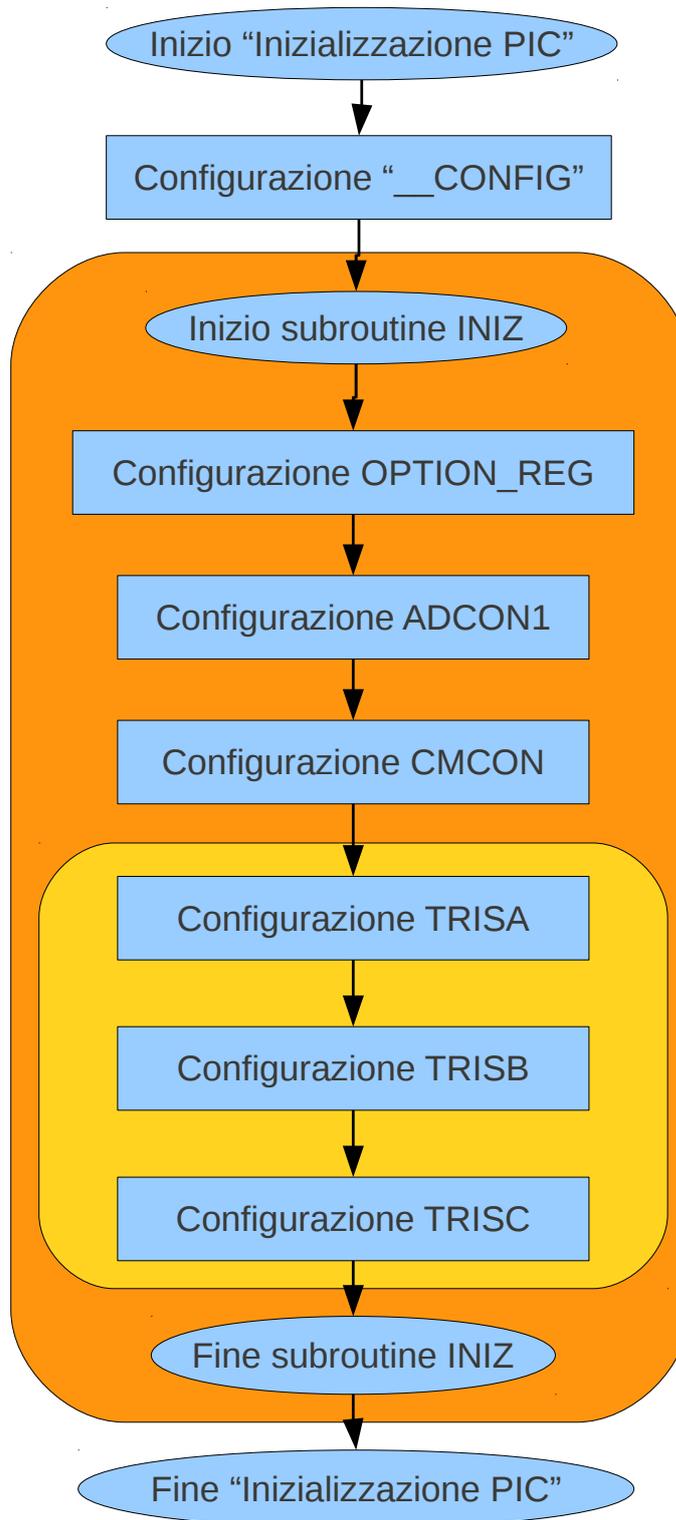
W = bit scrivibile

-n = bit inizializzato al valore n al reset

Nel nostro caso non sono necessari comparatori, quindi è necessario disabilitarli, caricando nel registro il valore 7.

### Configurazione TRISA, TRISB, TRISC

I registri TRISA, TRISB, TRISC consentono di configurare rispettivamente i singoli pin delle porte A, B e C come ingressi o uscite. Per impostare, ad esempio, un pin di PORTA come ingresso è necessario porre a 1 il rispettivo pin del registro TRISA. Ovviamente, mentre nel caso di PORTB e PORTC il numero di bit del rispettivo registro TRIS corrisponde al numero di pin, nel caso di PORTA il numero di pin è pari a 6, mentre i bit di TRISA è pari a 8; i bit 6 e 7 risultano pertanto inutilizzati e non influenti sugli altri pin. Nel nostro caso, ciascun banco è stato settato con la medesima configurazione per tutti i bit, in particolare tutto il banco A e tutto il banco C sono stati settati come uscite mentre il banco B è stato settato come ingresso. In questo modo è stata operata una separazione per banchi tra ingressi, uscite di potenza e uscite di segnale.



**Figura 4.2:** Flow-chart dettagliato di "Inizializzazione PIC"

### 4.2.2 Polling

A differenza del blocco “Inizializzazione PIC” in questo caso non si tratta di un’unico blocco generico che viene scomposto in sottoblocchi specifici, ma di vari blocchi già specifici che verranno analizzati nel complesso in quanto singolarmente assumono poco significato.

Il ciclo di Polling comprendente quattro blocchi che hanno lo scopo di controllare se il pilota desidera incrementare o decrementare un rapporto. Nel caso in cui entrambe le condizioni risultino false il ciclo continuerà a ripetersi fino a quando almeno una condizione risulta vera. Una volta che viene accertata tale situazione, a seconda di quale sia il primo dei due ingressi che è stato variato, viene lanciato il blocco “Esegui SHIFT\_DOWN” o “Esegui SHIFT\_UP”.

#### I registri PORTA, PORTB, PORTC

I valori contenuti nel registro PORT di ciascuna porta equivalgono alla lettura dello stato dei pin nel caso questi sia impostati come ingressi, mentre nel caso siano impostati come uscite variando il valore del bit desiderato vi sarà una corrispondente variazione dell’uscita. Tutte le operazioni di scrittura sono del tipo leggi-modifica-scrivi, dunque la scrittura di una porta implica che i pin della porta siano innanzitutto letti e che successivamente ne venga modificato il valore, con la conseguente variazione dello stato dell’uscita.

### 4.2.3 Esegui SHIFT\_UP

Questo blocco ha il compito di eseguire l’incremento di un rapporto alla richiesta del pilota, tuttavia prima e dopo l’eventuale intervento sulle uscite, effettua dei controlli onde evitare cambiate indesiderate o multiple. In figura 4.3 è possibile vedere i vari sottoblocchi di questa parte dell’algoritmo che di seguito analizzeremo singolarmente.

#### Attendi 2ms

Va innanzitutto ricordato che si arriva in questo blocco solamente quando si è rilevata una variazione dello stato dell’ingresso associato alla salita di marcia. A questo punto in questo blocco si introducono 2ms di ritardo software durante il quale il microcontrollore non varia nessuna uscita.

#### Lettura ingresso salita

Viene ora controllato nuovamente l’ingresso associato alla salita di marcia, questo per controllare se effettivamente la richiesta di cambiata sia stata effettuata dal pilota e non a causa di un disturbo. Infatti se tale ingresso è controllato dal pilota l’impulso di cambiata dura molto difficilmente meno di 100ms, quindi durante i 2ms di attesa questo porta a una stabilizzazione del segnale. Nel caso in cui, invece sia un disturbo a causare la variazione dell’ingresso, durante i 2ms d’attesa questo fa in tempo a ritornare a livello logico alto.

#### L’ingresso salita è a “1”?

Questo blocco di scelta valuta, a seconda dello stato dell’ingresso salita, se procedere nella cambiata o se ritornare al ciclo di polling. Precisamente se risulta a “0” logico si procederà con l’esecuzione della cambiata, mentre se risulta a “1” logico il programma tornerà al ciclo di polling. Come si può notare questa operazione ritarda di fatto la cambiata, tuttavia 2ms sono un ritardo assolutamente irrilevante in quanto non influisce sul tempo di cambiata, ma semplicemente ne ritarda l’inizio di un’intervallo di temporale molto piccolo.

#### Poni a “1” le uscite CUT\_OFF e SOL\_UP

A questo punto si agisce finalmente sulle uscite, sia su quella che comanda l’elettrovalvola sia su quella che invia il segnale di cut\_off alla centralina, ponendole entrambe ad “1”.

### Attendi 160ms

A seguito del cambiamento di stato delle uscite viene atteso un tempo che permette al nuovo rapporto di innestarsi con assoluta certezza. Per questa ragione il tempo fissato è stato di 160ms anche se il tempo effettivo di innesto del rapporto risulta considerevolmente inferiore. Va ricordato che il tempo durante il quale il pistone pneumatico rimane a fine corsa non influisce sui tempi di cambiata, ma semplicemente impedisce la possibilità di effettuare una nuova cambiata, vedi 1.3.

### Poni a “0” le uscite CUT\_OFF e SOL\_UP

Questo blocco riporta a “0” entrambe le uscite e questo ha l’effetto di consentire il ritorno della leva nella posizione di neutro, nonché di riportare l’ingresso della centralina motore in alta impedenza. Va ricordato tuttavia che il tempo di cut-off non dipende da tale uscita, ma essa ne stabilisce soltanto l’istante di inizio, rimandando alla centralina motore il compito di fissarne la durata.

### Attendi 48ms

A questo punto si attende un tempo pari a qualche decina di ms per due ragioni:

- La cambiata, dal punto di vista elettronico è completata, tuttavia prima di poter abilitare una nuova cambiata è necessario attendere il ritorno meccanico della leva cambio nella posizione di neutro.
- Poiché il tempo medio con il quale il pilota tiene premuto l’interruttore d’ingresso è relativamente lungo, quest’ulteriore attesa aumenta le possibilità che una volta terminato tale blocco l’ingresso sia basso. Peraltro, questo non costituisce una limitazione alla possibilità di inserire due rapporti in un breve arco di tempo in quanto questo, nel caso più sfavorevole, è di qualche secondo.

### Lettura ingresso salita e valutazione se è a “0”

A questo punto la cambiata è terminata sia dal punto di vista elettronico che meccanico, tuttavia se l’algoritmo facesse ritorno immediatamente al ciclo di polling principale e il pilota nel frattempo stesse ancora premendo l’interruttore questo porterebbe ad un’ulteriore cambiata indesiderata. Quest’eventualità è tutt’altro che improbabile, se ad esempio il pilota preme l’interruttore per un secondo vi sarebbero cinque cambiata consecutive. Risulta quindi opportuno aggiungere questo ciclo con il fine di evitare situazioni del genere, imponendo che prima di ritornare nel ciclo di polling venga rilasciato l’interruttore e che quindi l’ingresso ritorni a “1” logico.

#### 4.2.4 Esegui SHIFT\_DOWN

Questo blocco, similmente a “Esegui SHIFT\_UP”, ha il compito di modificare le uscite in modo da scalare un rapporto. A differenza dell’incremento di marcia, tuttavia, cambiano il numero di pin d’ingresso e d’uscita. Questi ultimi infatti da due diventano uno in quanto il cutoff non viene più utilizzato, mentre gli ingressi da uno diventano due poiché oltre alla richiesta del pilota di decrementare rapporto il programma deve anche verificare che la frizione risulti staccata prima di abilitare la cambiata 1.3.2. In figura 4.4 si può vedere il flow-chart dettagliato della scalata; di seguito ne analizzeremo le differenze rispetto a quello della salita.

Come si può notare, l’inizio del diagramma a blocchi in questo caso non inizia con un tempo di attesa come nel caso precedente, ma con il controllo della posizione della frizione. Questo per garantire che a inizio ciclo siano rispettate entrambe le condizioni necessarie a decrementare un rapporto, ovvero la richiesta di scalata da parte del pilota e la frizione premuta. Nel caso questa condizione non sia rispettata vi è un ritorno al ciclo di polling.

A seguito di questa prima parte del flow-chart segue una parte molto simile a quella della fase di shift-up, ovvero è presente una fase di attesa di 2ms con a seguito una fase di verifica degli ingressi con lo scopo di confermare o meno la richiesta di cambiata.

Se questa verifica ha un risultato positivo inizia la cambiata vera e propria, con la variazione all'1 logico dello stato dell'uscita, il mantenimento di tale stato per 160ms ed a seguire il ritorno dell'uscita allo 0 logico.

A questo punto la cambiata è stata effettuata e, come nel caso della shift-up, è presente un breve ciclo di attesa con a seguire un'altro ciclo che continua a controllare lo stato dell'ingresso di scalata fino a quanto questo non risulta alto in modo da evitare cambiate multiple. Questo permette di scollegare completamente il tempo con il quale il pilota preme l'interruttore con il tempo di durata dell'impulso di cambiata.

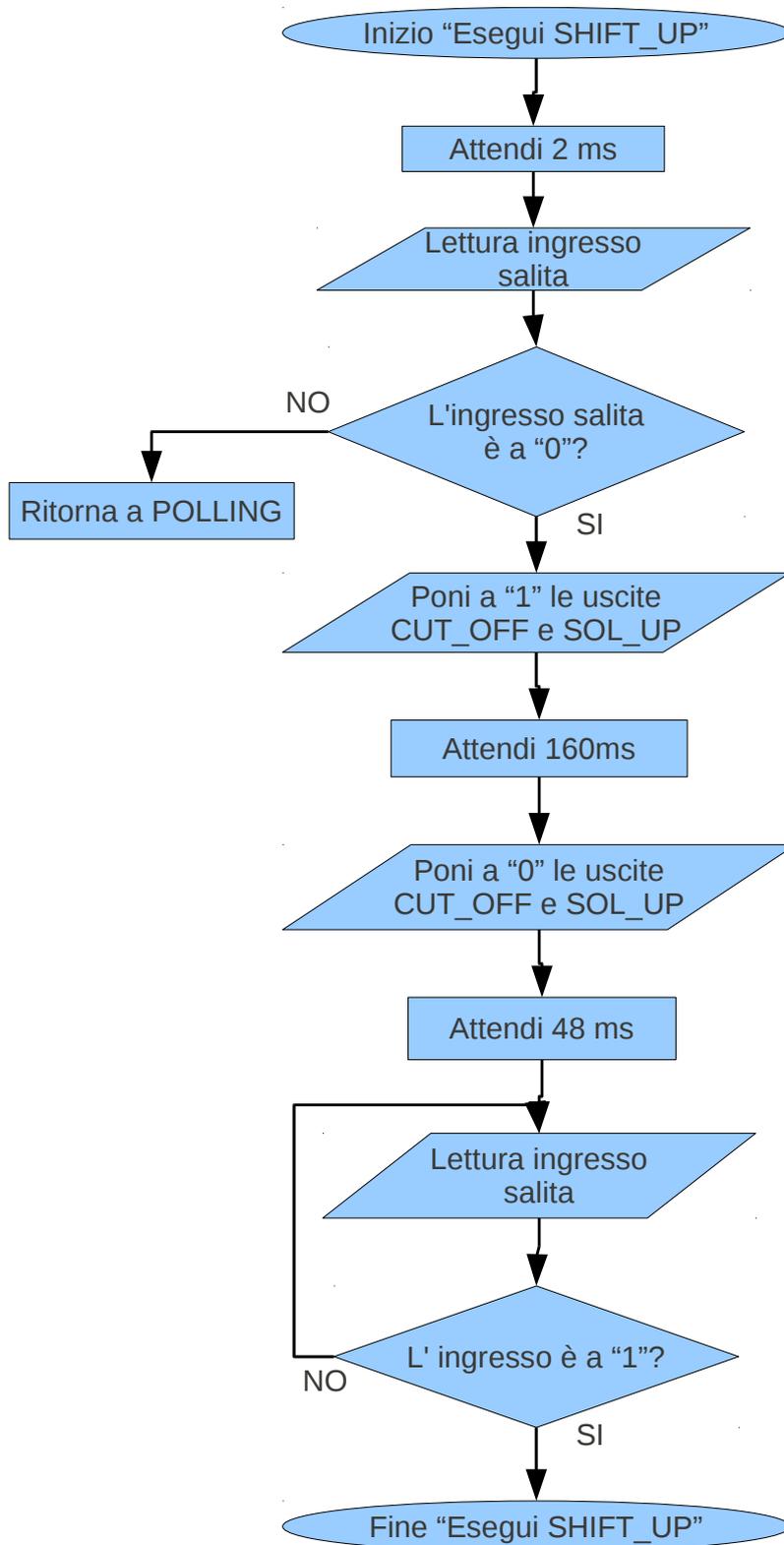
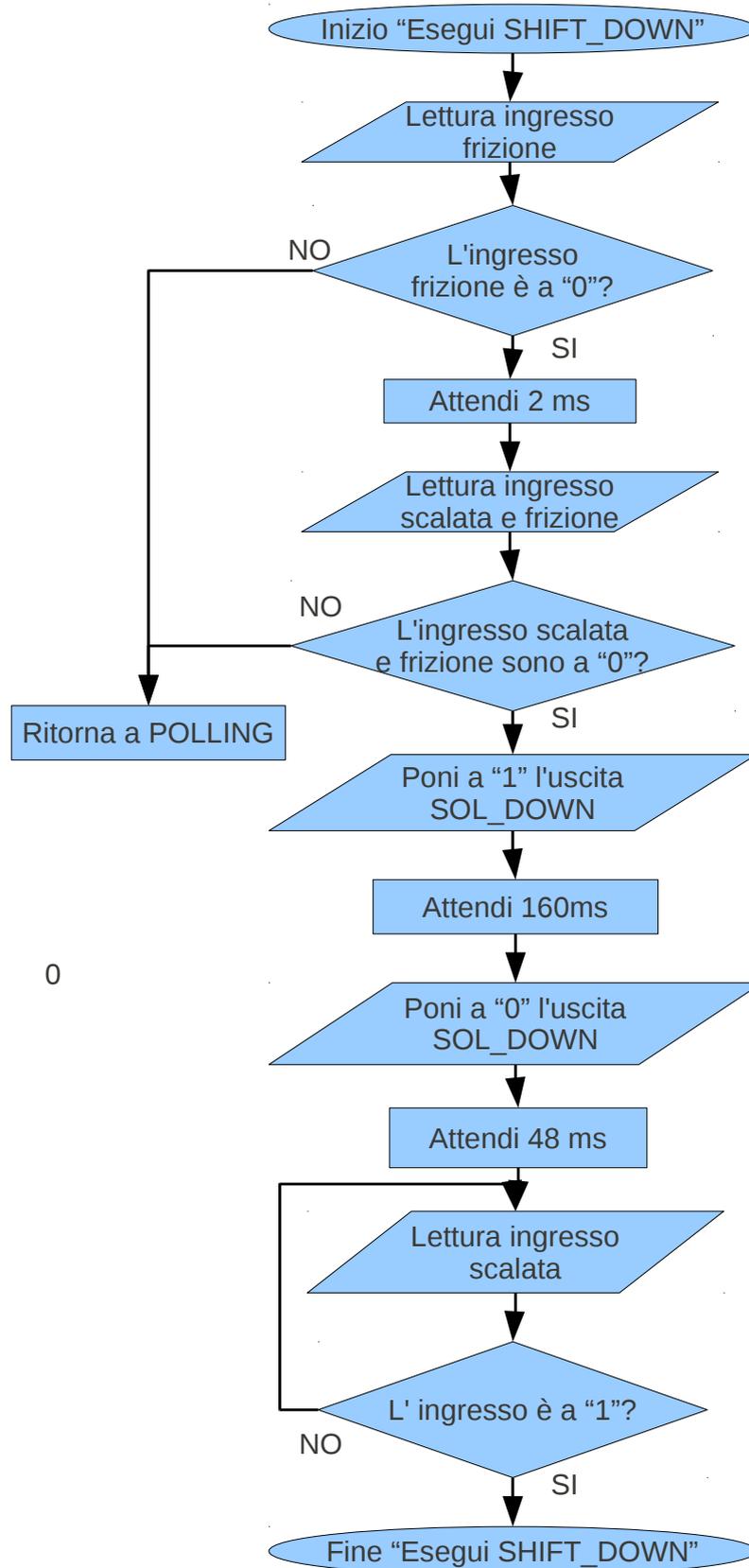


Figura 4.3: Flow-chart dettagliato di "Esegui SHIFT\_UP"

**Figura 4.4:** Flow-chart dettagliato di "Esegui SHIFT\_DOWN"

### 4.3 Il codice:

In questo paragrafo esponiamo il codice assembly che implementa l'algoritmo precedentemente analizzato. A seguire analizzeremo il significato delle istruzioni che non sono ancora state trattate.

---

```
PROCESSOR      16F873
RADIX          DEC
INCLUDE        "P16F873.INC"
ERRORLEVEL     -302
__CONFIG      0x3F7D
```

```
WAIT          EQU      0x71
WAIT1         EQU      0x72
```

```
;*** PORT A ***
```

```
SOL_1        EQU      00h
SOL_2        EQU      01h
SOL_3        EQU      02h
SOL_4        EQU      03h
SOL_5        EQU      04h
SOL_6        EQU      05h
```

```
;*** PORT B ***
```

```
IN_1         EQU      00h
IN_2         EQU      01h
IN_3         EQU      02h
IN_4         EQU      03h
IN_5         EQU      04h
IN_6         EQU      05h
IN_7         EQU      06h
IN_8         EQU      07h
```

```
;*** PORT C ***
```

```
OUT_1        EQU      00h
OUT_2        EQU      01h
OUT_3        EQU      02h
OUT_4        EQU      03h
OUT_5        EQU      04h
OUT_6        EQU      05h
OUT_7        EQU      06h
OUT_8        EQU      07h
```

```
;*** ASSEGNAZIONE PORTE ***
```

```
OUT_UP       EQU      SOL_3
OUT_DOWN     EQU      SOL_2
```

```
IN_UP        EQU      IN_1
IN_DOWN      EQU      IN_2
FRIZ         EQU      IN_3
```

```

CUT_OFF      EQU      OUT_2

OUT_UP_ON    MACRO          ;Comando solenoide salita attivo
                BSF PORTA,OUT_UP
                ENDM
OUT_UP_OFF    MACRO          ;Comando solenoide salita disattivo
                BCF PORTA,OUT_UP
                ENDM

OUT_DOWN_ON   MACRO          ;Comando solenoide scalata attivo
                BSF PORTA,OUT_DOWN
                ENDM
OUT_DOWN_OFF  MACRO          ;Comando solenoide scalata disattivo
                BCF PORTA,OUT_DOWN
                ENDM

CUTOFF_ON     MACRO          ;Comando cutoff attivo
                BSF PORTC,CUT_OFF
                ENDM
CUTOFF_OFF    MACRO          ;Comando cutoff disattivo
                BCF PORTC,CUT_OFF
                ENDM

;*****
;* INIZIO DEL CODICE
;*****
ORG      0x00
CALL    Init

Polling
        CLRWDT
        BTFSS  PORTB,IN_UP
        GOTO   Shift_up
        BTFSS  PORTB,IN_DOWN
        GOTO   Shift_dn
        GOTO   Polling

;*****
;* Init: Inizializzazione dei registri e delle porte
;*****
Init
        CLRWDT
;*** OPTION_REG ***
;*
;* bit value descr.
;* 7  0 PORTB pull-ups are enabled by individual port latch values
;* 6  0 Interrupt on falling edge of RB0/INT pin
;* 5  0 Internal instruction cycle clock (CLKO) for TMR0
;* 4  0 Increment on low-to-high transition on RA4/T0CKI pin for TMR0
;* 3  1 Prescaler is assigned to the WDT
;* 2-0 07h WDT Rate 1:128

        BANKSEL OPTION_REG
        MOVLW  0x0F
MOVWF   OPTION_REG

;*** ADCON1 ***

```

```

;*
;* bit value descr.
;* 7 0 A/D result left justified
;* 6 0 A/D Conversion Clock Select bit
;* 5-4 0
;* 3-0 07h A/D Port Configuration Control bits (All Digital)

```

```

BANKSEL  ADCON1
MOVLW    0x07
MOVWF    ADCON1

```

```

;*** CMCON ***
;*
;* bit value descr.
;* 7 0
;* 6 0
;* 5 0
;* 4 0
;* 3 0
;* 2-0 07h Comparator OFF

```

```

BANKSEL  0x9C ;CMCON
MOVLW    0x07
MOVWF    0x9C ;CMCON

```

```

;*** TRISA ***
;*
;* bit value descr.
;* 5 0 OUT
;* 4 0 OUT
;* 3 0 OUT
;* 2 0 OUT
;* 1 0 OUT
;* 0 0 OUT

```

```

BANKSEL  TRISA
          MOVLW    00000000b
          MOVWF    TRISA

```

```

;*** TRISB ***
;*
;* bit value descr.
;* 7 1 IN
;* 6 1 IN
;* 5 1 IN
;* 4 1 IN
;* 3 1 IN
;* 2 1 IN
;* 1 1 IN
;* 0 1 IN

```

```

          MOVLW    11111111b
          MOVWF    TRISB

```

```

;*** TRISC ***
;*
;* bit value descr.
;* 7 0 OUT

```

```

;* 6 0 OUT
;* 5 0 OUT
;* 4 0 OUT
;* 3 0 OUT
;* 2 0 OUT
;* 1 0 OUT
;* 0 0 OUT

        MOVLW    00000000b
        MOVWF    TRISC

        CLRF     STATUS

        RETURN

;*****
;* Wait: Ciclo di attesa di lms per il valore di W al momento della chiamata @ 4
;*****
Wait
        MOVWF    WAIT

int1
        MOVLW    0xF9          ;Dipende dalla frequenza
        MOVWF    WAIT1

int2
        CLRWDI
        DECFSZ   WAIT1
        GOTO     int2
        DECFSZ   WAIT
        GOTO     int1

RETURN

;*****
;* Shift_up: Pressione del pulsante Shift up
;*****
Shift_up
        MOVLW    0x02 ;ANTI-rimbalzo up
        CALL     Wait
        BTFSC    PORTB,IN_UP
        GOTO     Polling
        CUTOFF_ON
        OUT_UP_ON
        MOVLW    0xA0          ;TEMPO di cambiata up
        CALL     Wait
        OUT_UP_OFF
        CUTOFF_OFF
        MOVLW    0x30
        CALL     Wait
        BTFSC    PORTB,IN_UP
        GOTO     Polling
        CLRWDI
        GOTO     $-3

;*****
;* Shift_dn: Pressione del pulsante Shift down
;*****
Shift_dn
        BTFSC    PORTB,FRIZ ;controllo pre attesa
        GOTO     Polling
        MOVLW    0x02          ;ANTI-rimbalzo down
        CALL     Wait

```

---

```

BTFSC    PORTB, IN_DOWN
GOTO     Polling
BTFSC    PORTB, FRIZ    ;controllo poso attesa
GOTO     Polling
OUT_DOWN_ON
MOVLW    0xA0           ;TEMPO di cambiata down
CALL     Wait
OUT_DOWN_OFF
MOVLW    0x30
CALL     Wait
BTFSC    PORTB, IN_DOWN
GOTO     Polling
CLRWDI
GOTO     $-3

```

---

**PROCESSOR:** E' una direttiva dell'assemblatore che consente di definire per quale microprocessore e' stato scritto il nostro source. Le direttive non sono delle istruzioni mnemoniche che il compilatore traduce nel rispettivo opcode, ma delle semplici indicazioni rivolte all'assemblatore per determinarne il funzionamento durante la compilazione.

**RADIX DEC:** E' una direttiva che serve ad informare l'assemblatore che i numeri riportati senza notazione sono da intendersi come numeri decimali. Ovvero, se intendiamo utilizzare il numero esadecimale 10 (16 decimale) non possiamo scrivere solamente 10 perche' verrebbe interpretato come 10 decimale, va quindi specificata la base di riferimento del numero se differente da quella di riferimento, in questo caso va scritto 0x10 oppure H'10'.

**INCLUDE:** E' un'altra direttiva che comunica all'assemblatore la nostra intenzione di includere nel source un secondo file denominato P16F84A.INC. Il compilatore si limitera' a sostituire la linea contenente la direttiva INCLUDE con il contenuto del file indicato e ad effettuare quindi la compilazione come se fosse anch'esso parte del nostro codice. In particolare il file "PIC16F873.INC" è un file già pronto (fornito assieme all'assemblatore) che contiene tutte le definizioni standard, come ad esempio i nomi associati ai registri.

**ERROLEVEL:** Anche questa è una direttiva e serve ad escludere la segnalazione di alcuni errori di compilazione.

**EQU:** E' anch'essa una direttiva e consente di definire delle costanti simboliche all'interno del source. Quest'operazione risulta molto utile in particolare per nominare le varie locazioni di memoria, in modo tale che sia possibile indirizzarle scrivendo semplicemente il loro nome simbolico, invece di doversi ricordare il numero alla quale ciascuna di esse è associata.

**MACRO:** E' una direttiva che consente di definire nuove istruzioni, o meglio di associare ad una label una sequenza di istruzioni. In fase di compilazione, ogni volta che nel corso del programma viene richiamata questa particolare label viene sostituita con il set di istruzioni definito nella rispettiva macro. Questo è uno strumento molto potente che consente di semplificare il programma, non solo per evitare di ripetere in continuazione blocchi identici di codice, ma anche per poterlo rendere più leggibile e più facilmente sviluppabile. Va tuttavia ricordato che l'uso spropositato di macro va evitato in quanto porta a una moltiplicazione della lunghezza del codice e quindi a una sua grande occupazione di memoria. Per costruire una macro è necessario utilizzare la seguente sintassi:

```

<label> MACRO [<arg>, ..., <arg>]
"codice"
ENDM

```

**CALL k:** Richiama in esecuzione una subroutine memorizzata all'indirizzo k. Il parametro k può essere specificato utilizzando direttamente il valore numerico dell'indirizzo oppure la

relativa label.

**CLRWDT:** Questa istruzione deve essere utilizzata quando programmiamo il PIC con l'opzione Watchdog abilitata (fusibile WDTE). In questa modalità il PIC abilita un timer che, una volta trascorso un determinato tempo, effettua il reset del PIC. Per evitare il reset il nostro programma dovrà eseguire ciclicamente l'istruzione CLRWDT per azzerare il timer prima che raggiunga il suo valore massimo. Se il timer non viene azzerato in tempo, la circuiteria di watchdog (dall'inglese "cane da guardia") interpreterà questo come un blocco del programma in esecuzione effettuando un reset del PIC.

**BTFSS f,b:** Testa il bit b contenuto nel registro all'indirizzo f e salta l'istruzione successiva se questo vale 1.

**BTFSC f,b:** Testa il bit b contenuto nel registro all'indirizzo f e salta l'istruzione successiva se questo vale 0.

**GOTO k:** Determina un salto del programma in esecuzione all'indirizzo k. Il parametro k può essere specificato utilizzando direttamente il valore numerico dell'indirizzo oppure la relativa label.

**BANKSEL k:** Serve per selezionare il registro k, evitando di settare e resettare manualmente il registro RP0, che è adibito alla selezione dei banchi di memoria.

**MOVLW k:** Assegna all'accumulatore W il valore costante k.

**MOVWF f:** Copia nel registro f il valore contenuto nell'accumulatore w.

**CLRF:** Questa istruzione azzerava il valore contenuto nel registro indirizzato dal parametro f.

**RETURN:** Questa istruzione deve essere inserita al termine di ogni subroutine per riprendere l'esecuzione del programma principale da dove era stato interrotto. Per subroutine si intende un blocco di istruzioni che viene eseguito ogni volta che questo viene richiamato per mezzo di una label. La differenza tra una macro ed una subroutine è che quest'ultima occupa uno spazio definito e costante di memoria, mentre una macro viene sostituita in fase di compilazione ogni volta che questa viene richiamata durante il programma.

**DECFSZ f:** Decrementa il valore del registro all'indirizzo f e se il risultato vale zero salta l'istruzione successiva.

### 4.3.1 Analisi

Analizzando il codice sopra esposto si può notare in modo molto chiaro la sua struttura.

Le prime righe di codice servono per definire le impostazioni di base del PIC, a seguire vi sono una serie di righe di codice dedicate ad affidare dei nomi a delle costanti in modo da semplificare le operazioni successive. A seguito di queste operazioni, vi sono tutte le macro che, ricordo, non sono parti di codice che verranno effettivamente inserite nel PIC, ma che saranno sostituite in fase di compilazione ogni volta che vengono richiamate. In particolare le macro che sono state definite sono sei e riguardano l'attivazione e la disattivazione delle uscite di cut-off, salita e discesa di marcia.

A seguito delle macro è presente l'istruzione ORG 0x00 che rappresenta il vettore di reset del nostro programma, ovvero è il punto dove, in caso di reset, ripartirà l'esecuzione del nostro programma. Dopo questa operazione inizierà quindi il programma vero e proprio, del quale nel paragrafo precedente è stato fornito e descritto il flowchart.

Il programma a questo punto comincia con l'inizializzazione dei registri descritti in dettaglio al paragrafo 4.2.1 e che riguardano l'impostazione di ingressi, uscite e dei timer del PIC. Analizzando il codice, come prima istruzione in questa fase del programma si può trovare CLRWDT, che è l'istruzione che azzerava il watchdog timer in modo che questo non raggiunga il valore massimo

e conseguentemente resettiti il PIC. Questa istruzione, che deve essere ripetuta con costanza durante l'esecuzione del PIC, non è stata citata nel flowchart proprio perché non è un'operazione che riguarda l'algoritmo funzionale del programma, ma serve solo nel caso in cui il PIC per qualche ragione vada in blocco, ad esempio a causa di un ciclo infinito; per eventi quindi non contemplati nella normale esecuzione del programma.

**Polling:** Dopo queste operazioni, che potremmo definire preliminari, inizia la vera e propria esecuzione del PIC, con l'inizio del ciclo di polling. A fianco troviamo la riproposizione del codice che esegue questa parte di programma. La prima istruzione rappresenta la label del ciclo, chiamata appunto Polling con a seguire l'istruzione di reset del watchdog timer. Va infatti ricordato che questa parte di programma è quella che viene eseguita ripetitivamente fino all'avvento di una condizione esterna, quindi all'interno di questo ciclo è necessario inserire un'istruzione di reset del watchdog timer. A seguito di questa istruzione c'è l'istruzione `BTFSS PORTB, IN_UP` che controlla lo stato di `IN_UP` della porta B. Nel caso questo sia pari a "0" continua normalmente ad eseguire il programma, mentre se è pari a "1" viene saltata l'istruzione successiva. Ricordo che il pin d'ingresso è tenuto a livello logico alto dal pull-up interno e viene posto a "0" quanto esternamente scatta dell'interruttore di salita di marcia.

Nel caso quest'ultima condizione si presenti allora verrà eseguita l'istruzione seguente, che esegue un salto alla label `Shift_up` mentre nel caso l'ingresso rimanga alto questa istruzione di salto non viene eseguita, passando alla successiva, che in modo simile controlla l'ingresso della scalata di marcia. Se la richiesta di scalata è stata effettuata viene eseguita l'istruzione successiva, ovvero il salto alla label `Shift_dn` altrimenti viene ignorata anche questa e si passa all'istruzione `GOTO Polling` che ricomincia il ciclo.

---

```

Polling
      CLRWDT
      BTFSS  PORTB, IN_UP
      GOTO   Shift_up
      BTFSS  PORTB, IN_DOWN
      GOTO   Shift_dn
      GOTO   Polling

```

---

**Wait:** Nel corso del programma è necessario introdurre vari ritardi. Per far ciò si ricorre ad una subroutine che introduce un ritardo software funzione del contenuto di qualche registro, ad esempio quello dell'accumulatore. In questo codice il ritardo introdotto da questa subroutine nel momento in cui viene richiamata è pari a 1ms moltiplicato per il contenuto presente nell'accumulatore. Sappiamo che la frequenza di clock del nostro PIC è pari a 4MHz e che un'istruzione semplice viene eseguita in 4 cicli di clock, cioè alla frequenza di 1MHz; questo significa che le istruzioni che richiedono un ciclo macchina per essere seguite vengono completate in 1 $\mu$ s. Va ricordato tuttavia che alcune istruzioni richiedono 2 cicli macchina per essere completate, ad esempio l'istruzione `GOTO` o `RETURN`, mentre altre istruzioni quali `DECFSZ` vengono eseguite in un ciclo macchina nel caso non venga rispettata la condizione di salto e in due cicli nel caso venga soddisfatta.

---

```

Wait
      MOVWF  WAIT
int1
      MOVLW  0xF9
      MOVWF  WAIT1
int2
      CLRWDT
      DECFSZ WAIT1
      GOTO   int2
      DECFSZ WAIT
      GOTO   int1
RETURN

```

---

A questo punto possiamo analizzare la subroutine che è stata utilizzata per introdurre tali ritardi. Per prima cosa il contenuto dell'accumulatore viene salvato nel registro denominato `WAIT`, definito nella prima parte del programma, a seguito di quest'istruzione viene caricato in accumulatore e quindi in un'altro registro, denominato `WAIT1`, il valore esadecimale `F9`, corrispondente a 249 in decimale. A seguito di queste prime istruzioni inizia un ciclo composto da 3 istruzioni: la prima resetta il watchdog timer, la seconda decrementa il registro `WAIT1` fino a quanto è uguale a 0 e in quel caso salta l'istruzione successiva e la terza salta alla label `int2`. La prima istruzione viene eseguita sempre in un ciclo macchina, la seconda in uno se il valore di `WAIT1` è diverso da 0 altrimenti in 2 e la terza sempre in due, ciò significa che questo ciclo viene eseguito tutte le volte in 4 cicli macchina tranne nel caso in cui `WAIT1` sia pari a 0, quando viene eseguito in 3 cicli. Il numero di cicli che esegue `int2` sono quindi pari a:  $cicli\_int2 = 248 * 4 + 3 = 995$ . Considerando che ogni ciclo viene eseguito in 1 $\mu$ s ciò significa

che `int2` introduce un ritardo di 995 $\mu$ s. Questo ciclo denominato `int2` è contenuto in un'altro ciclo denominato `int1` che, oltre al tempo di esecuzione di `int1` presenta altre istruzioni: le due di caricamento del registro `WAIT1`, quella di decremento con eventuale salto e quella di salto incondizionato. Queste istruzioni comportano 5 cicli macchina aggiuntivi nel caso in cui `WAIT` sia diverso da 0, mentre soltanto di 4 nel caso in cui `WAIT` sia pari a 0. A seguito di questo ciclo è presente l'istruzione di `RETURN` che termina la subroutine e che viene seguita in due cicli macchina. Nel conteggio del numero di cicli necessari ad eseguire l'intera subroutine va anche considerato che inizialmente deve essere caricato il valore presente nell'accumulatore nel registro `WAIT`, operazione che viene eseguita soltanto una volta.

Da questa analisi è emerso che se per ogni ciclo in cui `WAIT` è diverso da 0 il ciclo viene eseguito esattamente in 1000 cicli macchina, ovvero in 1ms. Nel caso in cui invece `WAIT` risulta pari a 0 viene eseguito in 1002 cicli macchina, ovvero in 1002 $\mu$ s o 1,002ms.

$$T_{ritardo}(us) = 1002 + 1000 * (WAIT - 1) \quad (4.1)$$

Questo significa che ci sono 2 $\mu$ s d'errore nel ciclo di ritardo, ma questi sono costanti al variare del tempo d'attesa, ovvero di `WAIT`. Percentualmente quindi l'errore diminuisce all'aumentare del valore di `WAIT`, anche se questo risulta comunque molto piccolo anche nel caso peggiore, ovvero per un ritardo pari a 1ms. Va precisato che `Wait` introduce ritardi di almeno 1ms, ovvero al momento della chiamata l'accumulatore deve contenere un valore maggiore di 0, pena il non corretto funzionamento della subroutine.

**Shift\_up:** Nel caso in cui venga rilevata la richiesta di eseguire un'incremento del rapporto di marcia si entra in una nuova parte del programma, riproposta qui di fianco. Si può notare che le prime due istruzioni caricano il valore 2 in accumulatore e successivamente richiamano la subroutine `Wait`, esse quindi introducono un ritardo di 2ms, come specificato dall'algoritmo 4.3. Dopo questa attesa viene controllato l'ingresso relativo all'incremento del rapporto e nel caso in cui questo risulti a "1" logico, cioè nel caso in cui la richiesta non risulti più attiva, si passa all'istruzione successiva che rimanda il programma al ciclo di polling. Nel caso invece in cui la condizione di salita di marcia risulti ancora attiva viene saltata l'istruzione successiva, passando alle due macro di attivazione del segnale di cambiata e di `cut_off`. A seguito dell'attivazione delle uscite atte ad incrementare il rapporto inizia un tempo di

---

Shift\_up

```

MOVLW    0x02
CALL     Wait
BTFSC   PORTB, IN_UP
GOTO    Polling
CUTOFF_ON
OUT_UP_ON
MOVLW    0xA0
CALL     Wait
OUT_UP_OFF
CUTOFF_OFF
MOVLW    0x30
CALL     Wait
BTFSC   PORTB, IN_UP
GOTO    Polling
CLRWDT
GOTO    $-3

```

---

attesa pari a 160ms. Come si può osservare il valore che viene inserito nell'accumulatore prima del richiamo della subroutine `Wait` è espresso in esadecimale, quindi pari ad `A0`, corrispondente a 160 in decimale. Una volta terminato il ciclo di attesa vengono abbassate le uscite tramite due macro, operazione che conclude la cambiata e in successione al quale e viene atteso un tempo pari a 48ms, che corrisponde a 30 in esadecimale. Terminato anche questo ciclo di attesa viene controllato l'interruttore di cambiata. Nel caso questo risulti inattivo (a stato logico "1") viene fatto ritorno al ciclo di polling tramite l'istruzione di salto successiva, in caso contrario (stato logico "0") tramite l'istruzione `GOTO $-3` si fa ritorno all'istruzione di controllo dello stato dell'ingresso. Da notare che prima di quest'ultima istruzione è stata inserita l'istruzione di reset del watchdog timer, questo in quanto è presente un un ciclo che non è dato sapere a priori quanto durerà. L'omissione di tale istruzione può infatti comportare il reset del PIC che una volta riavviato, nel caso l'interruttore di cambiata risulti ancora premuto, farà partire un nuovo ciclo di cambiata, il tutto a pochi ms di distanza dalla cambiata precedente.

**Shift\_down:** In ultima analisi, resta da valutare la subroutine che attua la scalata di un rapporto. A differenza del caso precedente, e come specificato nel rispettivo algoritmo 4.4, le prime due istruzioni non sono un tempo d'attesa, ma servono per controllare lo stato dell'ingresso frizione. Se questo non è a "0" logico, cioè se la frizione non risulta staccata, si fa ritorno al ciclo di polling. In caso contrario c'è un ciclo d'attesa di 2ms passato il quale vengono controllati sia l'ingresso frizione sia quello di scalata e se entrambi risultano attivi viene posta ad "1" l'uscita della scalata di marcia, in caso contrario si fa ritorno al ciclo di polling. Se la cambiata viene abilitata c'è un ciclo d'attesa di 160ms, seguito dalla disabilitazione dell'uscita. Dopo questa operazione il codice risulta identico a quello del caso di shift\_up, con l'unica differenza che il ciclo non è relazionato con l'ingresso relativo alla salita di marcia ma, ovviamente, a quello di scalata.

---

```

Shift_dn
    BTFSC    PORTB,FRIZ
    GOTO     Polling
    MOVLW   0x02
    CALL     Wait
    BTFSC    PORTB,IN_DOWN
    GOTO     Polling
    BTFSC    PORTB,FRIZ
    GOTO     Polling
    OUT_DOWN_ON
    MOVLW   0xA0
    CALL     Wait
    OUT_DOWN_OFF
    MOVLW   0x30
    CALL     Wait
    BTFSC    PORTB,IN_DOWN
    GOTO     Polling
    CLRWDI
    GOTO     $-3

```

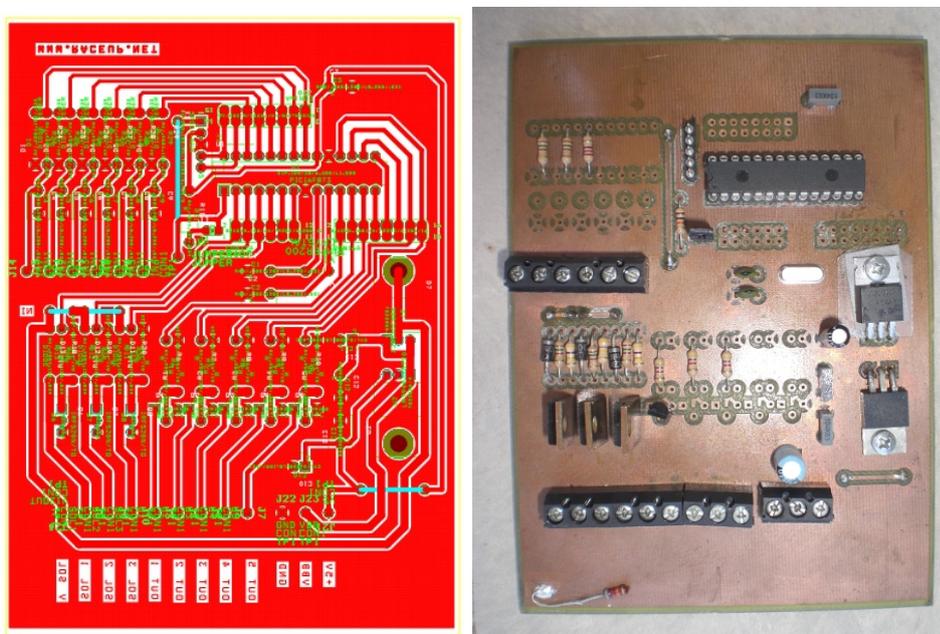
---

## Capitolo 5

# Considerazioni finali

### 5.1 La scheda

Dallo schema elettrico è stato ricavato il file con la disposizione dei componenti della scheda, il quale è riportato in figura 5.1. Il file in questione rappresenta la scheda vista dal lato componenti, nello stesso lato della scheda troviamo le piste di rame rappresentate dal colore azzurro, mentre in rosso sono rappresentate dall'altro lato della scheda. A fianco troviamo invece la scheda già realizzata e con i componenti montati. Possiamo notare che mancano diversi componenti in quanto, come già detto, la scheda in questione è stata pensata per un'uso il più possibile generico, quindi è predisposta per poter collegare anche altri ingressi o uscite, con la possibilità di sfruttare maggiormente le porte del PIC. Oltre a questo dalla figura si nota come i diodi schottky, utilizzati per rettificare la tensione d'ingresso, siano fissati alla scheda previo un'opportuno spessore di materiale plastico che ha lo scopo di isolare la parte metallica del case con la scheda. Mentre infatti la superficie della scheda non coperta da piste ha il potenziale di massa, la superficie metallica del case corrisponde al potenziale del cadoto comune dei diodi, cioè circa 12V.



**Figura 5.1:** File pcb della scheda e scheda realizzata

La vettura della stagione 2010 del Race-UP, L'MG/0510, ha utilizzato fin dai primi test su strada questa scheda per il controllo del cambio ed i risultati sono stati molto incoraggianti. I tempi di cambiata sono infatti risultati molto buoni, tanto che si è deciso di non scindere i tempi di inizio del tempo di cambiata con quello di cut-off. Questo infatti è stato fissato a 43ms di cui soltanto 35 di taglio totale dell'alimentazione e 8 di taglio parziale in modo da garantire

una riaccensione del motore non brusca. Questo dato attesta che l'interruzione dell'erogazione di potenza avviene per un'intervallo estremamente limitato, migliorando notevolmente le prestazioni della vettura.

La rete RC di filtro per gli ingressi, inoltre, è risultata inutile in quanto anche collegando direttamente gli interruttori a volante non si sono presentati mai fenomeni di cambiate multiple. Questo probabilmente sia per merito dei cavi molto lunghi (qualche metro) che di fatto introducono una componente RC, sia per merito del ritardo software previsto nel codice. Per quanto riguarda il posizionamento della scheda cambio essa è stata posta all'interno di un contenitore plastico sotto al sedile del pilota a fianco del serbatoio del carburante, un luogo areato e facilmente accessibile.



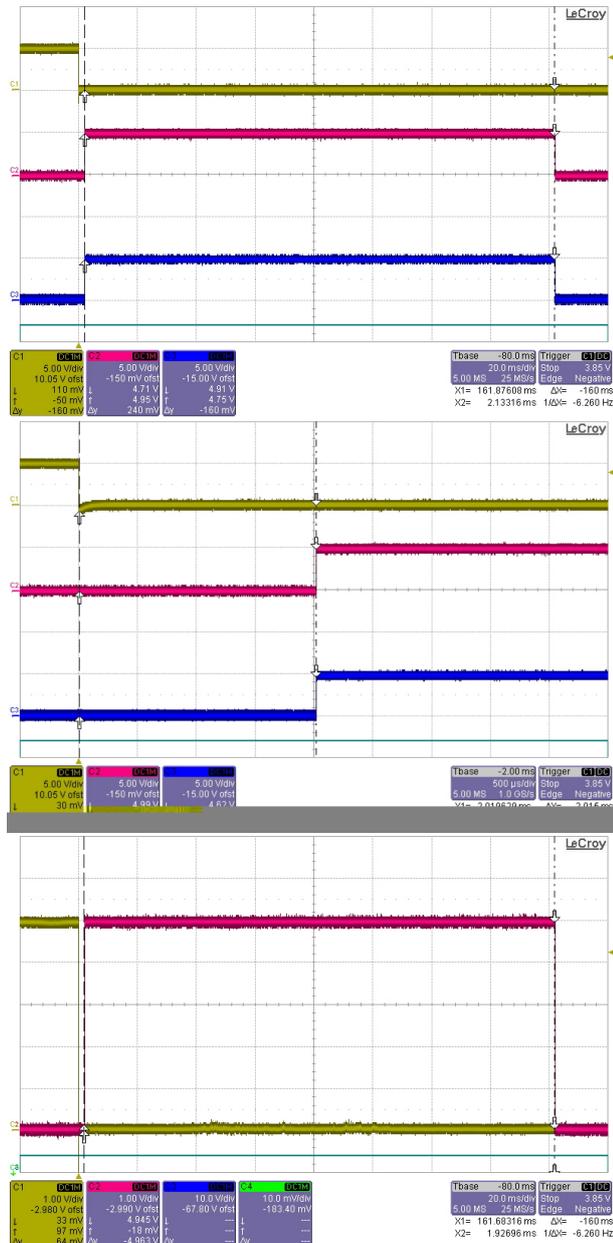
**Figura 5.2:** Posizione della scheda in vettura con e senza carrozzeria

Oltre a verifiche pratiche sono state effettuate anche delle misurazioni per confermare se effettivamente il sistema si comportasse come richiesto. Per far ciò si è utilizzato un oscilloscopio digitale in grado di memorizzare l'intervallo di tempo di interesse. In particolare quest'intervallo deve durare più di 160ms e cominciare nell'istante in cui si preme l'interruttore di salita o discesa del cambio. Nelle figure 5.3 si può notare che le uscite rispettano perfettamente quanto richiesto dal software: in giallo è rappresentato l'ingresso di salita o di scalata (l'ingresso frizione è stato disabilitato per comodità), in rosa l'uscita di potenza per le elettrovalvole ed in blu l'uscita di cut-off presente solo in fase di salita. Si può notare, specialmente nella seconda figura, che l'ingresso presenta solo un leggero undershoot a seguito del quale l'ingresso va rapidamente a regime (pochi  $\mu$ s), molto prima del secondo controllo dell'ingresso effettuato dopo 2ms.

A seguito della fase di test la squadra ha partecipato all'evento di Monza, durante il quale non si sono presentati problemi di sorta per quanto riguarda la scheda in questione. Nei test successivi, tuttavia, si sono cominciati a riscontrare lievi malfunzionamenti quali ad esempio rapide cambiate in successione, il mancato funzionamento della scalata o della salita di marcia o la rottura di alcuni componenti. Tutti questi problemi sono risultati derivanti dal distaccamento delle piste della scheda o addirittura dal troncamento dei piedini di alcuni dispositivi. La causa comune di questi guasti è risultata essere l'enorme stress meccanico al quale è stata sottoposta la scheda nei pochi chilometri percorsi nei test e all'evento di Monza. La vettura infatti risulta essere estremamente rigida sia a causa delle sospensioni, studiate per avere una corsa limitata, sia a causa del motore che non è fissato, come sulle normali vetture, su dei supporti che smorzano gran parte delle vibrazioni, ma direttamente al telaio. Si è quindi cercato di porre rimedio sostituendo i componenti danneggiati e inserendo i nuovi con l'accortezza di fissarli in modo da limitarne il più possibile i movimenti, ripristinando le piste distaccate con dei fili, rinforzando le saldature e posizionando la scheda in un supporto morbido. Queste precauzioni però hanno allungato minimamente il tempo di vita della scheda, che poco dopo ha cominciato a ripresentare gli stessi problemi. Tuttavia questo ha permesso alla squadra di poter partecipare all'evento più importante della stagione, quello di Silverstone, con il cambio funzionante.

Da queste considerazioni è emerso che, nonostante la scheda sia risultata efficace nel controllo in questione, il montaggio della stessa manualmente e non in modo automatizzato, con l'utilizzo di tecniche industriali specifiche per il campo automobilistico, in generale non dà la garanzia del funzionamento della stessa se non per brevi periodi. Un'evoluzione di questa scheda dovrà pertanto tenere conto anche di quest'aspetto in fase di realizzazione, affidandone l'assemblaggio

ad un'azienda specializzata e prevedendo fin da subito il montaggio della stessa su di un supporto in grado di assorbire parte delle vibrazioni.



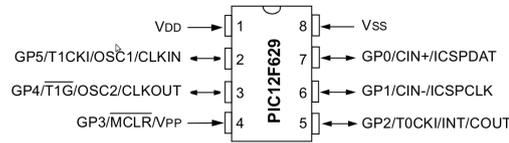
**Figura 5.3:** Misurazioni con l'oscilloscopio: In fase di salita con misurazione del tempo di attivazione delle uscite, In fase di salita con misurazione del tempo di attesa software e in fase di scalata con misurazione del tempo di attivazione delle uscite

## 5.2 Una possibile evoluzione

Questa scheda, come più volte specificato, risulta essere il primo passo per la realizzazione di una gestione del cambio ottimale. Durante il suo utilizzo è emerso che, se il suo funzionamento non risulta compromesso dalle vibrazioni, essa adempie in modo ottimale allo scopo per il quale è stata pensata. Le dimensioni della stessa risultano tuttavia eccessive, molto spazio è infatti utilizzato da ingressi e uscite che non vengono utilizzate, inoltre sono presenti all'uscita di tutte le porte dei fori che permettono di agganciare delle sonde in fase di test ma che di fatto, durante il suo normale funzionamento, occupano un'area notevole senza alcuno scopo. Oltre a modificare la scheda per ridurne le dimensioni si può anche pensare di cambiare il modello del

PIC in quanto quello utilizzato risulta essere fin troppo performante per le necessità effettive di tale scheda.

Un' alternativa possibile è costituita dal PIC16f629, un PIC dotato di soli 8 piedini, a differenza dei 28 del 16f873 utilizzato fino ad ora. Nell'immagine 5.4 è riportata la piedinatura dell'integrato, mentre nella tabella 5.1 la descrizione dettagliata della funzione di ciascun pin.

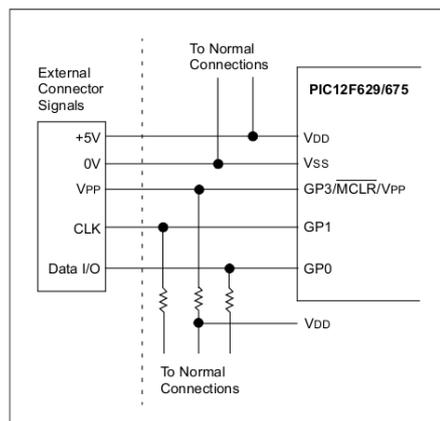


**Figura 5.4:** piedinatura PIC12f629

Ricordando che nel nostro caso devono essere utilizzati 3 ingressi e 3 uscite si può notare che questo integrato risponde alle esigenze del sistema, l'unica porta presente infatti (denominata GPIO) dispone di 6 ingressi/uscite configurabili, non viene dunque lasciato inutilizzato nessun pin, visto che gli altri due rimanenti sono riservati all'alimentazione. A differenza del microcontrollore precedente con questo non è possibile fornire il clock esternamente in quanto i pin che vi sarebbero destinati devono obbligatoriamente essere utilizzati come ingressi/uscite. Si può tuttavia ricorrere al clock interno al PIC che ha una frequenza di 4MHz, ovvero la stessa del PIC16f873 analizzato fino ad ora. Questa particolarità consente di poter riutilizzare buona parte dell'intero codice, risulta infatti necessario cambiare solamente la parte di configurazione del PIC in quanto, ovviamente, un PIC diverso necessita di differenti configurazioni, mentre la parte principale dell'algoritmo rimane totalmente invariata.

Come si può notare in realtà non tutti i 6 pin della porta sono configurabili come ingressi e uscite. Il pin GP3, infatti, può essere configurato solo come ingresso ed è privo di pull-up interno in quanto viene utilizzato anche per la programmazione del PIC. Poiché, come detto, tutti i pin della porta devono assolvere ad una certa funzione è necessario utilizzare tale pin come ingresso, con l'utilizzo di un pull-up esterno. In figura 5.5 è riportata dal datasheet una tipica connessione del PIC al programmatore.

Gli altri pin della porta possono invece essere scelti indifferentemente sia come ingressi che come uscite, inoltre tutti dispongono di una rete interna di pull-up nel caso essi siano configurati come ingressi. Per semplificare l'hardware tuttavia risulta conveniente assegnare come ingressi i pin che si trovano sullo stesso lato del PIC e tutte le uscite dall'altro, perciò GP3, GP4 e GP5 vanno configurate come ingressi, mentre GP0, GP1 e GP2 come uscite.



**Figura 5.5:** Connessione tipica dell'integrato in fase di programmazione

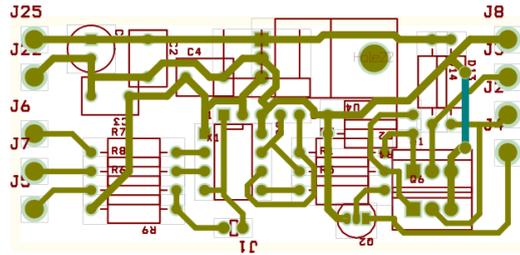
A questo punto quindi la scheda si semplifica notevolmente, il PIC infatti non presenta più pin inutilizzati, inoltre non sono più riservati 3 pin per l'alimentazione, ma due e non sono più presenti pin dedicati esclusivamente al reset del PIC e alla sua programmazione, ma vengono utilizzati i pin a disposizione per svolgere più funzioni. Ciò consente di compattare notevolmente le dimensioni della scheda. Quanto trattato riguarda solamente il PIC, mentre

Tabella 5.1: Descrizione dei pin del PIC12f629

Name	Function	Input Type	Output Type	Description
GP0/AN0/CIN+/ICSPDAT	GP0	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change Comparator Input
	CIN+	AN		
GP1/AN1/CIN-/VREF/ICSPCLK	GP1	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change Comparator Serial programming clock
	CIN- ICSPCLK	AN ST		
GP2/AN2/T0CKI/INT/COUT	GP2	ST	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change TMR0 clock input External InterruptX Comparator output
	T0CKI	ST		
	INT COUT	ST	CMOS	
GP3/ $\overline{MCLR}$ /VPP	GP3	TTL		Input port w/ interrupt-on-change Master Clear
	$\overline{MCLR}$	ST		
GP4/AN3/ $\overline{T1G}$ /OSC2/CLKOUT	GP4	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change TMR1 gate Crystal/resonant Fosc/4 output1
	$\overline{T1G}$	ST		
	OSC2 CLKOUT		XTAL CMOS	
GP5/T1CKI/OSC1/CLKIN	GP5	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change TMR1 clock Crystal/resonator External clock input/RC oscillator connection
	T1CKI	ST		
	OSC1 CLKIN	XTAL ST		
VSS	VSS	Power		Ground reference
VDD	VDD	Power		Positive Supply

si possono considerare anche molti altri aspetti che consentono di semplificare ulteriormente la scheda in modo da ridurre il più possibile le dimensioni fisiche. In tale ottica si può pensare di togliere il filtro RC presente sugli ingressi in quanto, come si è precedentemente spiegato, essi sono risultati inutili. Si può anche pensare di togliere la coppia di diodi schottky, il regolatore di tensione risulta infatti già dotato di una protezione contro l'inversione di polarità, in più le elettrovalvole possono essere alimentate direttamente dalla batteria collegando i diodi contro le sovratensioni presenti ai capi del MOSFET anch'essi direttamente ai 12V della batteria. Lo schema elettrico della scheda con tutti questi accorgimenti è riportato in figura 5.2. Si può notare che è stato aggiunto, rispetto allo schema precedente, solo la rete di pull-up per l'ingresso GP3; è stato invece mantenuto un collegamento esterno di massa collegato ai transistor d'uscita, questo con lo scopo, da un lato, di garantire il corretto riferimento di massa e dall'altro di limitare i disturbi causati dai picchi di corrente che si presentano quanto viene attivato un transistor. Nello schema elettrico si può notare che non sono presenti i collegamenti per l'alimentazione del PIC, questo non in quanto non siano presenti ma in quanto impliciti; ricordo comunque che, come risulta dalla tabella 5.4, il pin 1 va collegato a Vdd (5V) e il pin 8 a massa. Nella figura 5.6 è riportata una schematizzazione di come si potrebbero disporre i componenti in una

scheda realizzata con lo schema elettrico 5.2 e con tecnologia a montaggio non superficiale, di più facile realizzazione. Questa soluzione risulta, a parità della tipologia di componenti usati e di funzioni svolte, estremamente più compatta con dimensioni di circa  $6,5 \times 2,5$ cm, contro i  $10 \times 13,5$ cm della versione precedente con un'area globale utilizzata circa 8 volte inferiore. Questa soluzione peraltro mantiene la caratteristica, comune alla precedente realizzazione, di avere quasi tutte le piste presenti in un solo lato della scheda con un'unica eccezione, come si può notare dalla pista di colore blu. Questo consente di mantenere i costi contenuti in quanto è possibile realizzare il progetto su una scheda con le piste da un solo lato, per poi effettuare successivamente per mezzo di un filo l'unico ponte necessario.



**Figura 5.6:** Esempio di un possibile realizzazione della scheda con il PIC12f629





# Appendice A

## Regolamento SAE: sistema elettrico

### ARTICOLO 11: SISTEMA ELETTRICO

#### B11.1 Interruttore generale

**B11.1.1** Il veicolo deve essere equipaggiato con due (2) interruttori generali. L'attuazione di uno dei due interruttori deve spegnere il motore.

**B11.1.2** Il simbolo internazionale dell'elettricità, che consiste in un lampo rosso su un fondo blu bordato di bianco, deve essere affisso in stretta prossimità di ciascuno di questi interruttori.

**Note:** Si ricorda alle squadre che tutti i cavi dell'alternatore devono essere disabilitati da ciascun interruttore principale per prevenire ogni possibile ritorno di tensione attraverso il circuito delle bobine.

#### B11.2 Interruttore generale primario

B11.2.1 L'interruttore generale primario deve:

- a. Essere allocato alla destra del veicolo (del pilota), in prossimità del Main Hoop, all'altezza della spalla e facilmente azionabile da fuori dell'abitacolo.
- b. Disabilitare la potenza a TUTTI i circuiti elettrici, inclusa la batteria, l'alternatore, le luci, la/e pompa/e benzina, gli iniettori e i controlli elettronici.
- c. Tutta la corrente della batteria deve fluire attraverso questo interruttore.
- d. Deve essere della tipologia a rotazione e ad azione diretta. Ad esempio non può funzionare per mezzo di un relè.

Un'esempio di interruttore che corrisponde ai requisiti è mostrato qui sotto.



B11.2.2 La posizione di "OFF" dell'interruttore generale deve essere chiaramente contrassegnata.

#### B11.3 Interruttore generale in abitacolo

B11.3.1 L'interruttore generale in abitacolo:

- a. Deve essere posizionato in modo da garantire una facile attuazione da parte del pilota in situazione di emergenza o panico.
- b. Deve essere facilmente azionabile da parte del pilota con le cinture allacciate, vicino al volante, e non ostruito dal volante stesso o da altre parti dell'auto. Si suggerisce di posizionarlo nello stesso lato del volante dov'è posizionata la leva delle marce.
- c. Deve essere un'interruttore di emergenza della tipologia "premi/spingi". L'interruttore deve essere installato in modo tale che:
  - i. Dalla posizione di ON, premendolo si disabilita la potenza sugli iniettori, su tutte le pompe benzina.
  - ii. Dalla posizione di OFF, tirandolo si abiliterà la potenza sugli iniettori e sulle pompe benzina. Gli interruttori che necessitano di essere ruotati o di essere ruotati e tirati per abilitare la potenza sono accettati.
- d. Può funzionare attraverso un relè.

Esempi tipici di interruttori che rispettano questi requisiti sono riportati qui sotto



#### B11.4 Batterie

B11.4.1 Tutte le batterie, ad esempio per le forniture di energia a bordo, devono essere ben fissate al telaio.

B11.4.2 Ogni batteria che contiene liquidi situata nell'abitacolo deve essere racchiusa da un contenitore non conduttivo di tipo marino o equivalente.

B11.4.3 I terminali sotto tensione devono essere isolati in tutta la vettura.

## Appendice B

# Regolamento SAE: sistema frenante

### **B7.3 Interruttore di fine corsa dei freni**

B7.3.1 Un'interruttore di fine corsa deve essere installato nel pedale dei freni. Questo interruttore deve essere installato in modo che, nel caso di malfunzionamenti al sistema frenante per cui il pedale del freno percorra più della sua massima corsa, l'interruttore venga attivato con conseguente spegnimento del motore. Questo interruttore deve interrompere l'iniezione e togliere l'alimentazione a tutte le pompe elettriche del carburante.

B7.3.2 L'attuazione ripetuta dello switch non deve ripristinare l'alimentazione a queste componenti, e deve essere posizionato in modo tale che il pilota non possa disattivarlo.

B7.3.3 L'interruttore deve essere implementato con componenti analogici, e non attraverso controllori logici programmabili, unità di controllo motore o altri controllo digitali con funzionamento analogo.

### **B7.4 Luce stop**

B7.4.1 L'auto deve essere equipaggiata con una luce stop rossa di almeno 15W o, equivalentemente, chiaramente visibile dal retro dell'auto. Se è utilizzata una luce stop a LED, essa deve essere chiaramente visibile anche in un'ambiente molto luminoso.

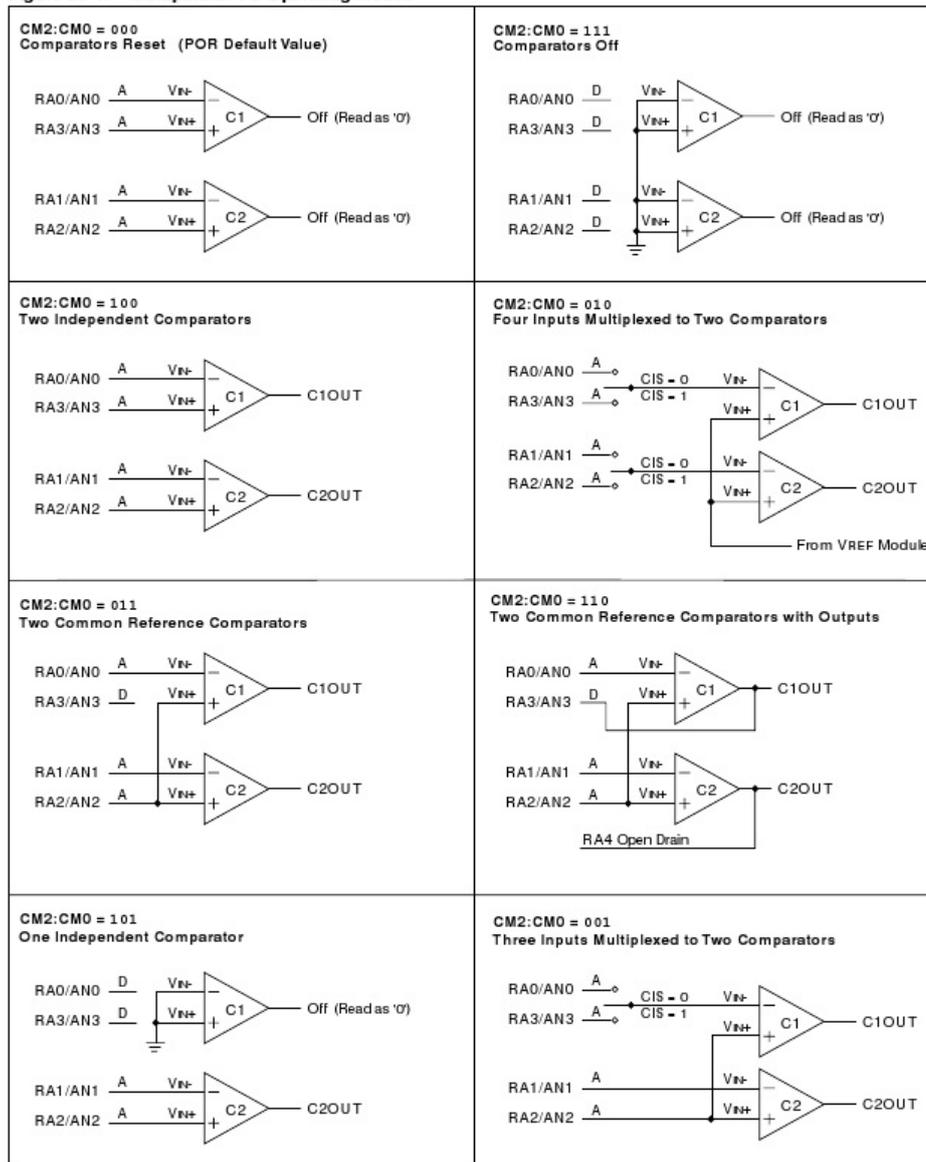
B7.4.2 Questa luce deve essere montata con altezza compresa tra il centraruota e le spalle del pilota e approssimativamente nella linea di mezzzeria della vettura.



# Appendice C

## Controllore PIC: Configurazione dei comparatori

Figure 20-1: Comparator I/O Operating Modes



A = Analog Input, port reads as zeros always.  
D = Digital Input.  
CIS (CMCON<3>) is the Comparator Input Switch.



# Bibliografia

- [1] Datasheet PIC16f873A:  
[ww1.microchip.com/downloads/en/devicedoc/39582b.pdf](http://ww1.microchip.com/downloads/en/devicedoc/39582b.pdf)
- [2] Datasheet PIC12f629:  
[ww1.microchip.com/downloads/en/devicedoc/41190c.pdf](http://ww1.microchip.com/downloads/en/devicedoc/41190c.pdf)
- [3] Datasheet LM2940T-5.0:  
[www.national.com/ds/LM/LM2940.pdf](http://www.national.com/ds/LM/LM2940.pdf)
- [4] Datasheet IRF520N:  
[http://www.datasheetcatalog.com/datasheets\\_pdf/I/R/F/5/IRF520N.shtml](http://www.datasheetcatalog.com/datasheets_pdf/I/R/F/5/IRF520N.shtml)
- [5] Datasheet BC547C:  
[http://www.datasheetcatalog.com/datasheets\\_pdf/B/C/5/4/BC547C.shtml](http://www.datasheetcatalog.com/datasheets_pdf/B/C/5/4/BC547C.shtml)
- [6] Datasheet MBR20100CTK:  
[tec.icbuy.com/product/productDownload/id/21806.html](http://tec.icbuy.com/product/productDownload/id/21806.html)
- [7] Datasheet programmatore KITPIC2  
[ww1.microchip.com/downloads/en/DeviceDoc/51553E.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/51553E.pdf)
- [8] Tutorial alla programmazione dei PIC:  
<http://www.grix.it/areariservata/Picbyexdown.php>  
<http://stor.altervista.org/pic/pic.htm>