

***LaurTec***

# **Ricevitore IR a 4 canali ad autoapprendimento**

**Come attivare un carico con il telecomando della TV**

**Autore :** *Ferrero Vercelli Renato (fer\_ver)*

**ID:** UP0004-IT

## INFORMATIVA

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

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

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

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

## AVVERTENZE

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

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

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

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

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

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

**Indice**

<b>Introduzione</b> .....	4
<b>Specifiche Tecniche</b> .....	5
<b>Applicazioni</b> .....	5
<b>Analisi del progetto</b> .....	6
Cenni sui protocolli dei telecomandi ad infrarossi (IR).....	6
Principio di funzionamento del sistema.....	10
Descrizione del circuito elettrico.....	10
Schema elettrico.....	14
<b>Istruzioni per il montaggio</b> .....	18
<b>Collaudo e messa in funzione</b> .....	21
Prima messa in funzione dopo il montaggio.....	21
Memorizzazione canali.....	22
Funzione test uscite.....	22
Messa in funzione ordinaria.....	22
<b>Analisi del Software</b> .....	23
<b>Analisi finale</b> .....	29
<b>Ringraziamenti</b> .....	30
<b>Bibliografia</b> .....	32
<b>History</b> .....	33

## Introduzione

L'idea di realizzare il ricevitore ad infrarossi (IR) è nata dalla necessità, se così si può definire, di accendere e spegnere una striscia LED di colore bianco che avevo posto, dietro al divano, a fianco di un'altra striscia LED a più colori (RGB) già collocata in precedenza e pilotata da un proprio telecomando.

Normalmente sul divano "giravano" già diversi telecomandi e mi è sembrato esagerato aggiungerne un altro per comandare questa nuova fonte d'illuminazione. Ho quindi pensato di utilizzare un tasto inutilizzato di uno di questi telecomandi per comandare la nuova striscia LED...ma quale tasto?

...e se un giorno avessi voluto cambiare il tasto del telecomando, avrei dovuto riprogrammare il PIC? Ho quindi pensato di realizzare un ricevitore ad IR a autoapprendimento. Durante la progettazione il sistema si è ampliato in quanto ho pensato di variare la luminosità dei LED sempre usando lo stesso tasto, inoltre ho pensato di creare più canali con funzioni diverse. Infatti il progetto finale del ricevitore è composto da 4 canali così funzionanti:

**Canale 1**      Uscita con MOSFET per il controllo di una striscia LED a 12V, 2A max., con regolazione luminosità tramite PWM.

**Canale 2**      Uscita a relay. Lo stato del relay può essere programmato in due modalità :

**Monostabile** : cioè che segue lo stato del pulsante premuto.

**Bistabile**     : cioè che cambia stato ad ogni nuova pressione del tasto abbinato.

La programmazione del modo di funzionamento avviene tramite il Jumper JP2. Anche l'uscita del contatto può essere programmata in due modi:

Se il ponticello SJ1 (che collega il comune del contatto del relay al positivo dei 12V) è inserito, si può prelevare direttamente il positivo dei 12V dal morsetto X5-2, mentre in caso contrario si ha in uscita un contatto pulito, in deviazione. Personalmente ho utilizzato questo canale per alimentare la striscia LED RGB, ed ho quindi eliminato il suo alimentatore dedicato (Consumo LED RGB 1,6 A.)

**Canale 3**      Questo canale funziona come il canale 2, con la differenza che la programmazione avviene tramite il jumper JP3 e che in uscita è solamente disponibile il contatto in deviazione "pulito". Personalmente ho fatto funzionare questo canale in modalità monostabile ed ho collegato il contatto del relay in parallelo ai pulsanti dell'impianto elettrico dell'abitazione che comandano il lampadario del salotto (Detto lampadario è già comandato da un dimmer che ne varia la luminosità).

**Canale 4**      Uscita con Triac per alimentare e regolare la luminosità di una lampadina alogena o incandescenza da 220V 300W max. (lampade elettroniche, se non appositamente progettate per un dimming non possono essere utilizzate).

## Specifiche Tecniche

**Alimentazione** : 12VDC

**Assorbimento** : 5 A max.

**Assorbimento attivo** : 140mA

**Assorbimento da spento** : circa 15mA

**Assorbimento in standby** : 5mA

**Alimentazione Triac**: 220Vac

**Potenza Carico Triac**: 300W max.

**Dimensioni** : 80 x 100 mm

**Peso**: 85g

**Product Number**: UP0004

**Versione PCB** : 1.0



**Nota:**

Il sistema presentato nell'articolo presenta delle parti ad alta tensione. Il progetto è pensato come sistema di sviluppo e nel realizzarlo devono essere prese le dovute precauzioni per un sistema ad alta tensione. Qualora non si abbia la dovuta esperienza montare solo la parte a bassa tensione e utilizzare i relay per lampade a 12V.

## Applicazioni

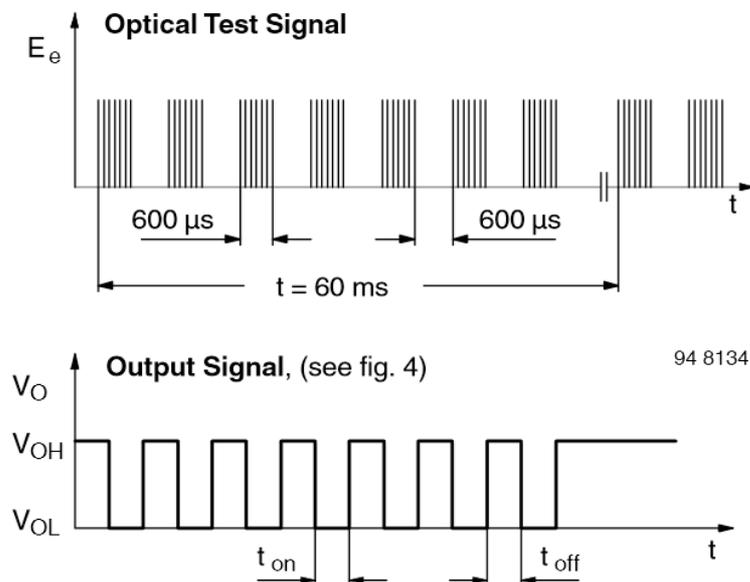
La scheda presentata si può utilizzare in molteplici applicazioni in cui sia necessario o utile un controllo remoto per mezzo di telecomando ad infrarossi.

- Accensione/Spengimento lampade.
- Accensione/Spengimento ventilatori.
- Accensione/Spengimento carichi induttivi/resistivi (tramite relay).
- Accensione/Spengimento carichi resistivi (tramite Triac).
- Azionamento serrande esterne.
- Togliere la tensione di rete al televisore quando non è in uso.
- Regolazione strisce LED o lampadine a 12Vcc.
- Regolazione lampade ad incandescenza a 220Vac.
- Azionare robot e modellini.

## Analisi del progetto

### Cenni sui protocolli dei telecomandi ad infrarossi (IR)

I telecomandi per TV trasmettono i loro comandi inviando una serie di impulsi attraverso un diodo LED ad infrarossi. La scelta degli infrarossi è dettata dal fatto di poter discriminare il segnale utile da eventuali luci presenti all'interno del locale che potrebbero saturare o disturbare il ricevitore. Inoltre, sempre per ovviare ai disturbi ambientali, quali ad esempio la luce del sole che colpisce il ricevitore, il segnale è trasmesso modulandolo per mezzo di una portante, che nella maggioranza dei casi varia da 36 a 40 KHz. In Figura 1 è riportato un esempio di segnale modulato. In particolare si noti che ogni livello 1 contiene la frequenza di modulazione, per cui si può capire che per ottenere tale segnale si può fare un AND (o NAND) logico tra il segnale da modulare e il segnale modulante.



**Figura 1:** Esempio di segnale digitale modulato (prelevato dal datasheet vishay TSOP34836).

L'esempio appena mostrato in Figura rappresenta un segnale di test modulato posto in ingresso al sensore TSOP34836 che permette di demodulare il segnale (Output Signal) fornendo il segnale digitale originale. Il fotodiodo, TSOP34836 è stato scelto perché permette di demodulare il segnale dei telecomandi ad infrarossi operanti a 36KHz fornendo un'ottima reiezione ai disturbi esterni. Il suo contenitore, riportato in Figura 2, rappresenta un filtro ottico selettivo per gli infrarossi, questo permette di ricevere solo i segnali nella banda d'interesse.



**Figura 2:** Fotodiodo TSOP34836 con front end e demodulatore integrato.

Oltre ad avere un filtro ottico e lente di focalizzazione, il TSOP34836 possiede il diodo di ricezione e tutta l'elettronica necessaria per amplificare, filtrare e demodulare il segnale in ingresso (maggiori dettagli sono mostrati in Figura 3). Questo permette di semplificare molto la parte di ricezione del segnale, infatti l'uscita del sensore TSOP34836 può essere collegata direttamente al PIC18F2331. Per maggiori dettagli sul sensore TSOP34836 si rimanda al datasheet in Bibliografia.

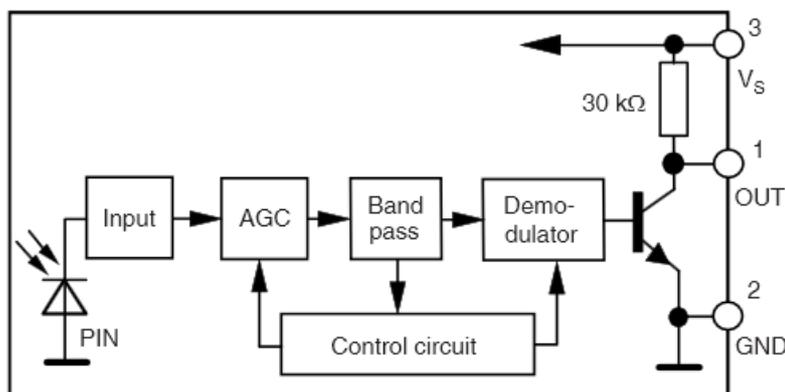


Figura 3: Schema a blocchi del fotodiodo TSOP34836 (prelevato dal datasheet vishay TSOP34836).

La durata ed il numero degli impulsi inviati dai telecomandi ad infrarosso, seguono un protocollo ben definito che varia da marca a marca ed dal modello. Il ricevitore qui descritto è stato progettato per essere principalmente usato con i protocolli SIRC della Sony ed l'RC5 della Philips, ma funziona anche con altri telecomandi che non seguono tali protocolli.

Il treno d'impulsi, o pacchetto che dir si voglia, inviato dal telecomando è composto da più parti che possono essere, in linea generale, così suddivise:

- Burst iniziale.
- Codice tasto premuto.
- Codice d'indirizzo dell'apparecchio (TV , Videoregistratore, lettore DVD, ecc...).

L'invio di un comando con protocollo SIRC, della Sony, può contenere 12, 15 o 20 bit ed è così composto:

- Parte iniziale: Singolo impulso avente un segnale alto della durata di 2,4 ms.
- Parte comando composta da 7 bit.
- Parte indirizzo composta dagli ultimi 5 bit.
- Nel caso di un telecomando a 20 bit vengono ancora aggiunti ulteriori 8 bit come estensione.

Per inviare un bit avente valore 0 viene inviato un segnale avente segnale basso per 0,6ms seguito da uno stato alto di durata analoga cioè di 0,6ms. Un bit di valore 1 invece è composto da un impulso avente segnale basso della durata di 0,6ms e una parte alta avente durata di 1,2ms.

Ne deriva quindi che l'invio di un singolo bit può durare 1,2ms o 1,8ms a seconda del valore del bit stesso, di conseguenza la lunghezza del pacchetto inviato varia da comando a comando.

Se il pulsante viene mantenuto premuto viene continuamente inviato il treno d'impulsi intervallato da un periodo di pausa. Il tempo totale del treno d'impulsi e relativa pausa è di 45ms.

In Figura 4 e Figura 5 sono mostrate le forme d'onda visualizzate sull'oscilloscopio, misurando l'uscita del fotodiodo TSOP34836 ( N.B. il segnale risulta invertito rispetto al segnale inviato in

quanto l'uscita del fotodiode è pilotata da un transistor NPN e quindi a riposo, cioè con valore binario 0, si ha una tensione di +5V derivante dalla resistenza di pull-up da 30Kohm interna).

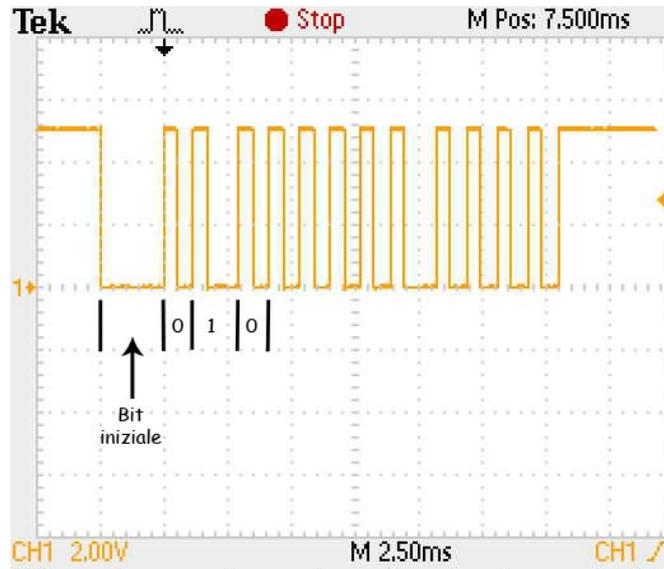


Figura 4: Esempio di ricezione di un comando protocollo Sony (SIRC).



Figura 5: Esempio di ricezione di un treno di impulsi protocollo Sony (SIRC).

L'invio di un comando con protocollo RC5, della Philips, è composto da 14 bit così suddivisi :

- 2 Bit aventi valore uguale ad 1.
- Stato del tasto: 1 Bit che alterna il valore ad ogni nuova pressione del tasto premuto.
- Indirizzo : 5 bit
- Comando : 6 bit

L'invio di ogni singolo bit ha una durata complessiva di 1,778ms composto da una parte bassa di durata pari a 0,889ms (cioè metà impulso) e da una parte alta di pari durata.

Per inviare un bit avente valore binario 1 l'impulso è composto da una parte iniziale bassa seguita da quella alta mentre un impulso composto inizialmente da una parte alta seguita da quella bassa ha valore 0.

In questo modo la durata del pacchetto d'impulsi trasmesso ha sempre la stessa durata, corrispondente a 25ms, indipendentemente dal comando inviato. In Figura 6 e in Figura 20 sono mostrati i dettagli del protocollo, misurando direttamente la tensione di uscita del TSOP34836 sottoposto alla ricezione di un telecomando Philips.

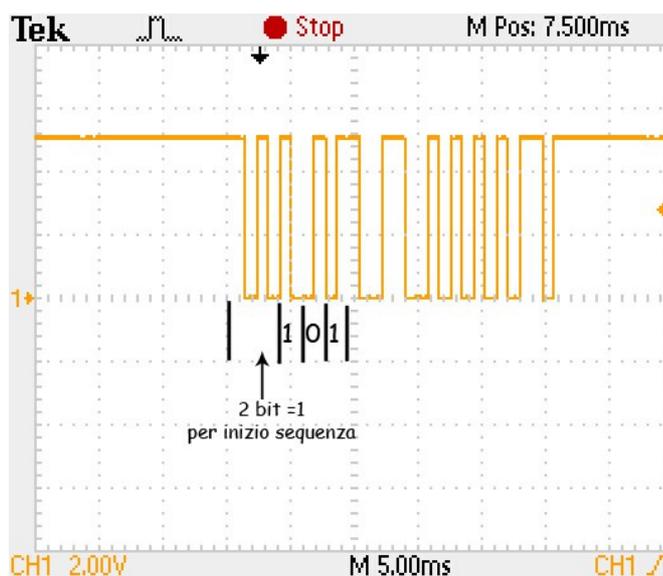


Figura 6: Esempio di ricezione di un comando protocollo Philips (RC5).



Figura 7: Esempio di ricezione di un treno di impulsi protocollo Philips (RC5).

## Principio di funzionamento del sistema

Il sistema qui descritto funziona indipendentemente dal protocollo utilizzato in quanto non rileva il codice composto dai vari bit ma misura, ed in seguito confronta o memorizza, la durata di ogni singolo stato del segnale rilevato. Ne deriva quindi che ogni bit trasmesso, composto da un impulso con uno stato basso ed uno alto, viene suddiviso in due nuove variabili denominate nel firmware Dsi (Durata Stato Impulso), aventi un valore proporzionale alla durata dello stato del segnale rilevato. Quindi, ad esempio, per misurare un pacchetto d'impulsi a 20 bit occorrono 40 variabili Dsi. Ogni singolo canale del ricevitore è progettato per ricevere ed analizzare comandi composti da 60 Dsi e quindi da un massimo di 30 bit. Nel caso in cui venga rilevato un pacchetto inviato con protocollo RC5 in cui il terzo bit varia ad ogni invio di dati, al momento del confronto verranno analizzati solo Dsi dal 6 in poi.

## Descrizione del circuito elettrico

Il circuito elettrico in se stesso non è molto complesso in quanto si tratta di un'apparecchiatura gestita da un microcontrollore PIC18F2331 del quale sono usati 3 pin come ingressi e 6 come uscite. La scelta del microcontrollore è stata dettata dalla necessità di avere una EEPROM interna di 256 byte per memorizzare i codici dei 4 canali, di avere almeno un modulo PWM, ed una memoria di programma di oltre 2KB.

Come ricevitore IR, in un primo momento, ho utilizzato un SFH5110 ma in seguito ho provato e quindi optato per un TSOP34836 in quanto risulta molto più sensibile, permettendo un controllo remoto a distanze maggiori.

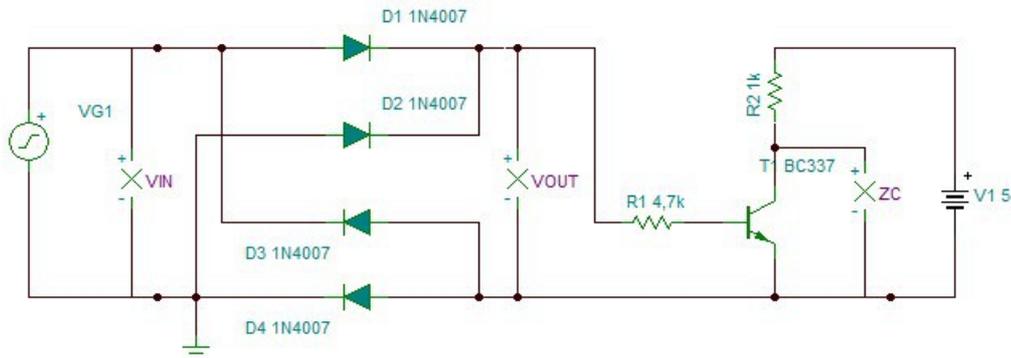
Per l'uscita pilotata dal modulo PWM, ovvero per controllare l'intensità luminosa della striscia LED o lampade da 12V, ho utilizzato un MOSFET IRF3205 in quanto, al momento della conduzione, presenta una resistenza massima tra il Drain e il Source di soli 0,008 ohm e quindi provoca una caduta di tensione ai suoi capi minima, riducendo la potenza dissipata internamente (per tale ragione non è richiesto un dissipatore termico).

Per il funzionamento e la spiegazione dei moduli PWM rimando al tutorial presente sul sito [www.LaurTec.it](http://www.LaurTec.it) (Brief Note BN0011: *Utilizzare il modulo CCP in modalità PWM ed Enhanced PWM*). Il comando delle uscite a relay è invece pilotata da un comune transistor NPN, BC337.

La sezione relativa al controllo dell'intensità luminosa delle lampade ad incandescenza da 220V richiede qualche nota in più. Diversamente dai LED pilotati in corrente continua, e per i quali la modulazione per mezzo del segnale PWM è sufficiente a regolare l'intensità luminosa, ovvero l'energia trasferita nel LED, per le lampade ad incandescenza alimentate in corrente alternata questo non è vero. Sebbene sia possibile scegliere una frequenza ad hoc di un segnale PWM in maniera da regolare l'intensità di una lampada ad incandescenza alimentata in alternata, i risultati potrebbero essere scadenti e si otterrebbe il cosiddetto "flickering" della luce. Per un controllo più accurato quello che normalmente viene fatto è modulare il segnale sinusoidale dell'alimentazione di rete, trasferendo solo una parte della sinusoide stessa. Per fare questo è necessario rilevare la partenza della sinusoide per mezzo della generazione di un segnale di zero crossing e attivare il Triac solo per un determinato tempo.

Un tipico rilevatore zero crossing è riportato in Figura 8. Lo schema è realizzato per mezzo del simulatore TINA. Il generatore in ingresso VG1 (VIN) è una sinusoide da 20Vpp, mentre i 4 diodi rappresentano un ponte raddrizzatore (ponte di Graetz). Il segnale in uscita dal ponte (VOOUT) viene applicato alla base del transistor. L'uscita del sistema, ovvero il collettore del transistor, permette di avere il nostro segnale di zero crossing. Il generatore V1 rappresenta una seconda alimentazione da 5V. Questo circuito simula proprio i 4 diodi dello schema di Figura 13 e il foto accoppiatore. Il diodo LED interno al foto accoppiatore è paragonabile alla giunzione base-emettitore del transistor, e l'uscita

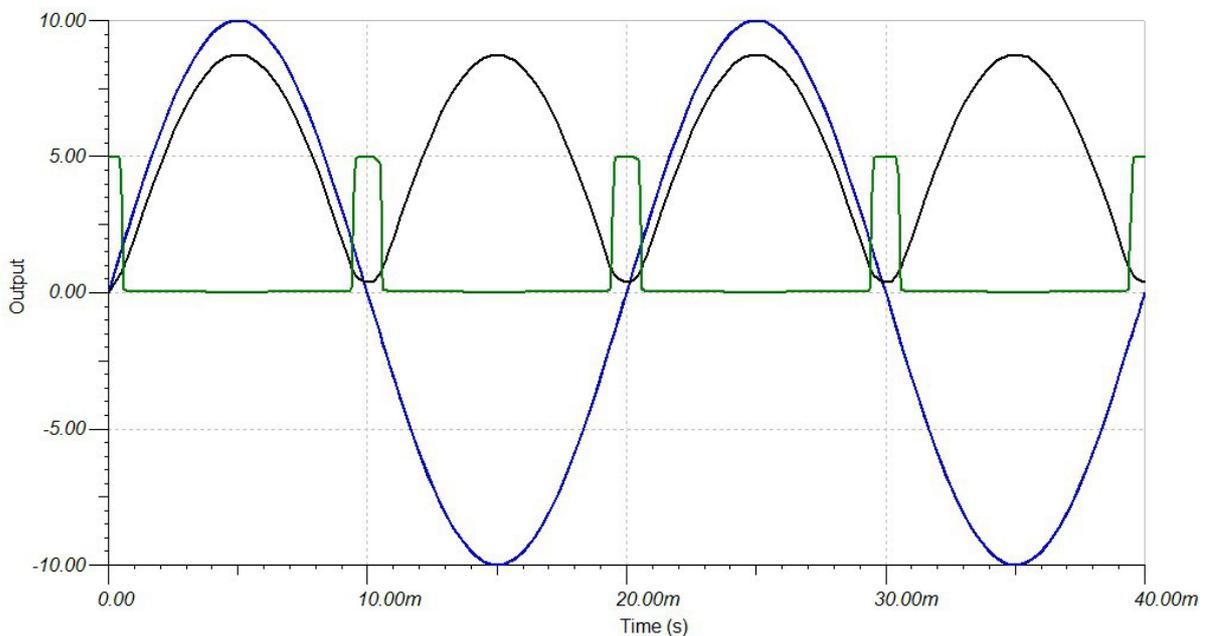
del foto accoppiatore è il collettore del transistor stesso. La doppia alimentazione di rete e del sistema a 5V è modellizzato proprio dai due generatori VG1 e V1. Durante la simulazione però si è posto VG1 pari a 20Vpp 50Hz.



**Figura 8:** Schema di esempio di un circuito Zero Crossing.

Simulando un transiente di 40ms si ottiene il diagramma di Figura 9, dove:

- VIN: Linea Blue
- VOUT: Linea Nera
- ZC: Linea Verde



**Figura 9:** Diagramma risultante dalla simulazione di 40ms.

Dalla simulazione si può vedere che il ponte di diodi, ribalta la semi-onda negativa ottenendo la linea nera. La linea nera differisce dalla linea blue di due cadute di tensione dei diodi (circa 1.4V) e parte non da 0V ma da 0.7V (ovvero il minimo necessario per mandare in conduzione i diodi del ponte). Quando la linea nera è maggiore di circa 0.7V la tensione Vbe sul transistor T1 è sufficiente per mandarlo in conduzione per cui il collettore viene “posto a massa” (il transistor è in saturazione, ovvero è un interruttore chiuso). Quando la tensione scende sotto gli 0.7V il transistor entra in

interdizione e il collettore si trova collegato a 5V per mezzo di R2, per cui nell'intervallo in cui la tensione rimane inferiore a 0.7V si ha un impulso, ovvero il nostro segnale zero crossing. In applicazioni reali in cui si voglia rilevare lo zero crossing della tensione di rete si usa normalmente un trasformatore per abbassare la tensione da 220V a 5V-12V. In Figura 13 invece, il fotoaccoppiatore è alimentato da un circuito RC, che permette comunque di abbassare la tensione di rete. L'impulso ricavato in questo caso risulta però sfasato di quasi 90°, come visibile in Figura 10, rispetto al punto zero reale a causa dello sfasamento introdotto dalla rete RC.

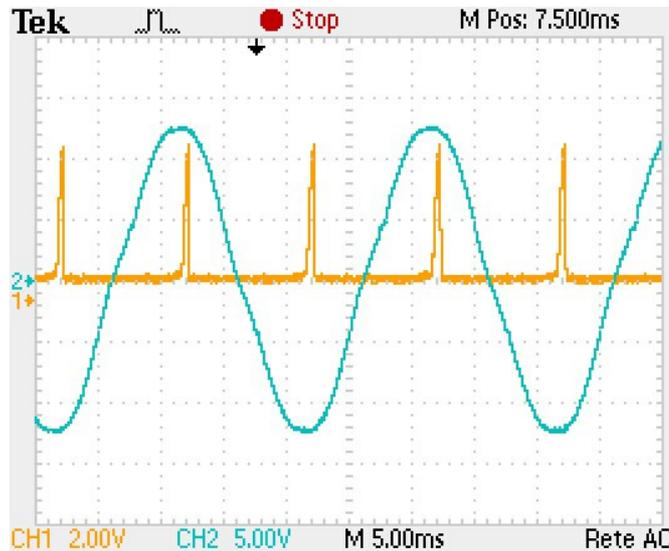


Figura 10: Esempio Uscita del Fotoaccoppiatore 4N33.

Questo sfasamento viene corretto via software e inviato in uscita al piedino 6 del PIC per poter essere misurato come riportato in Figura 11. Maggiori dettagli sulla correzione software sono descritti nel paragrafo relativo al Software.

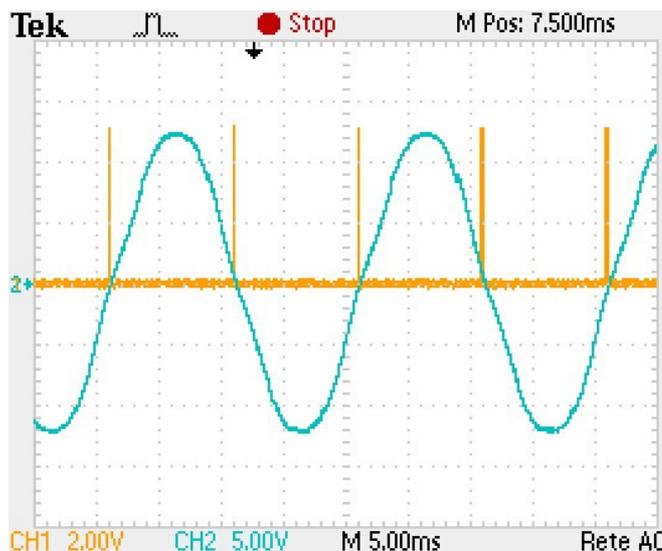
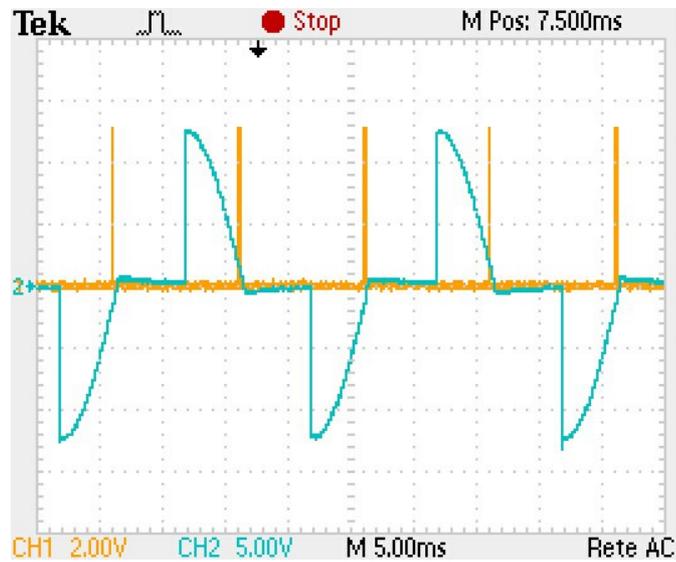


Figura 11: Esempio Uscita del PIC RA4 con segnale zero crossing compensata.

Una volta ottenuto il segnale di zero crossing, è possibile controllare in maniera accurata il trasferimento di energia verso la lampada ad incandescenza, attivando e disattivando il Triac in maniera da far passare solo una porzione dell'onda sinusoidale come riportato in Figura 12.



**Figura 12:** *Tensione di uscita ai capi del Triac.*



## Lista Componenti

### Resistori

**R1** = 1K  $\Omega$  %5 1/4W  
**R2** = 1K  $\Omega$  %5 1/4W  
**R3** = 33K $\Omega$  %5 1/4W  
**R4** = 33K $\Omega$  %5 1/4W  
**R5** = 470 $\Omega$  %5 1/4W  
**R6** = 820 $\Omega$  %5 1/4W  
**R7** = 820 $\Omega$  %5 1/4W  
**R8** = 180 $\Omega$  %5 1/4W  
**R9** = 820 $\Omega$  %5 1/4W  
**R10** = 470 $\Omega$  %5 1/4W  
**R11** = 470 $\Omega$  %5 1/4W  
**R12** = 470 $\Omega$  %5 1/2W  
**R13** = 100 $\Omega$  %5 1/2W  
**R14** = 10K $\Omega$  %5 1/4W  
**R15** = 100 $\Omega$  %5 1/4W  
**R16** = 270 $\Omega$  %5 1/4W  
**R17** = 470 $\Omega$  %5 1/4W  
**R18** = 2,7K $\Omega$  %5 1/4W

### Condensatori

**C1** = 220 $\mu$ F 25V. elettrolitico  
**C2** = 220 $\mu$ F 25V. elettrolitico  
**C3** = 0,1 $\mu$ F  
**C4** = 0,1 $\mu$ F  
**C6** = 0,1 $\mu$ F  
**C7** = 4,7 $\mu$ F  
**C8** = 10nF 600V.  
**C9** = 0,15 $\mu$ F 600V.

### Diodi

**LED1- LED3** = LED 3mm verde  
**LED4** = LED 3mm rosso  
**LED5** = LED 5mm giallo  
**LED5** = LED 5mm rosso

**D1:** 1N4007

**D2:** 1N4007

**B1:** Ponte diodi 1A 800V

### Transistor , Triac e Mosfet

**T1:** BC337

**T2:** BC337

**T3:** 600V 10A

**Q1:** IRF3205

### Circuiti integrati

**IC1** = 7805

**IC2**= PIC18F2331

**OK1** = 4N33

**OK2**= MOC3021

**IR** = TSOP34836

### Fusibili

**F1** = 5 A 5 X 20

**F2**= 3A 5 X 20

### Pulsanti

**S1** = n/a

### Relay

**K1** = 12V. 12A. Finder (Canale3)

**K2** = 12V. 12A. Finder (Canale2)

### Connettori

**X1** : Alimentazione 12V

**X2** : Ingresso ricevitore IR

**X3** : Uscita canale 1.

**X4** : Ingresso 230V e Uscita canale Triac canale 4

**X5** : Uscita canale 2

**X6** : Uscita canale 3

**SL1** : Programmatore PIC

Pin	Nome Pin	Direzione	Nome Linea	Funzione
1	MCLR	Input	RESET	Reset della scheda (non usato).
2	RA0	Output		Uscita Canale 2 (Relè K1).
3	RA1	Output		Uscita Canale 3 (Relè K2).
4	RA2	Output		
5	RA3	Output		Uscita Canale 4 (Triac).
6	RA4	Output		Uscita zero crossing.
7	VDD			+5V.
8	VSS			GND
9	RA7	Output		
10	RA6	Output		
11	RC0			
12	RC1	Output		Uscita Canale 1 (PWM2).
13	RC2	Input		
14	RC3	Input		Ingresso fotoaccoppiatore rilevatore punto zero.
15	RC4	Output		LED Giallo, RUN CPU.
16	RC5	Output		LED Rosso.
17	RC6	Input		
18	RC7	Input		
19	VSS			+ 5V
20	VDD			GND
21	RB0	Input		Jumper JP1 per la selezione del funzionamento Canale 2.
22	RB1	Input		Jumper JP2 per la selezione del funzionamento Canale 3.
23	RB2	Input		
24	RB3	Input		
25	RB4	Input		Pulsante apprendi e/o test uscite.
26	RB5	Input		Ingresso segnale IR.
27	PGC			
28	PGD			

**Tabella 1:** *Tabella riassuntiva delle connessioni del PIC18F2331.*

Connetore	N° MORSETTO	DESCRIZIONE
X1	1	Ingresso Positivo alimentazione +12V.
	2	Ingresso Negativo alimentazione GND.
X2	1	Negativo alimentazione ricevitore IR.
	2	Ingresso segnale IR.
	3	Positivo alimentazione sensore IR.
X3	1	Uscita negativo Canale 1 controllato da PWM 12V. max 2A.
	2	Uscita positivo Canale 1 controllato da PWM 12V. max 2A.
X4	1	Uscita per lampadina 230Vac max 300W.
	2	Ingresso neutro linea 230Vac e comune per lampadina 230Vac.
	3	Ingresso fase linea 230Vac.
X5	1	Uscita contatto NC relay Canale 2 (K1).
	2	Uscita contatto NA relay Canale 2 se inserito SJ1 uscita positivo 12V.
	3	Uscita comune contatto relay Canale 2 (K1).
X6	1	Uscita contatto NC relay Canale 3 (K2).
	2	Uscita contatto NA relay Canale 3 (K2).
	3	Uscita comune contatto relay Canale 3 (K2).

**Tabella 2:** *Tabella riassuntiva morsettiere scheda.*

## Istruzioni per il montaggio

Per questo circuito non vi sono particolari istruzioni per il montaggio, basta seguire le fondamentali nozioni per il montaggio delle schede elettroniche. A tal proposito rimando all'ottimo tutorial presente sul sito [www.LaurTec.it](http://www.LaurTec.it) (*Strumenti e Tecniche per il montaggio di KIT elettronici*). Al fine di agevolare il montaggio è bene far riferimento alla Figura 14 e Figura 15. In Figura 14 si noti la presenza dei 4 ponticelli riportati in rosso.

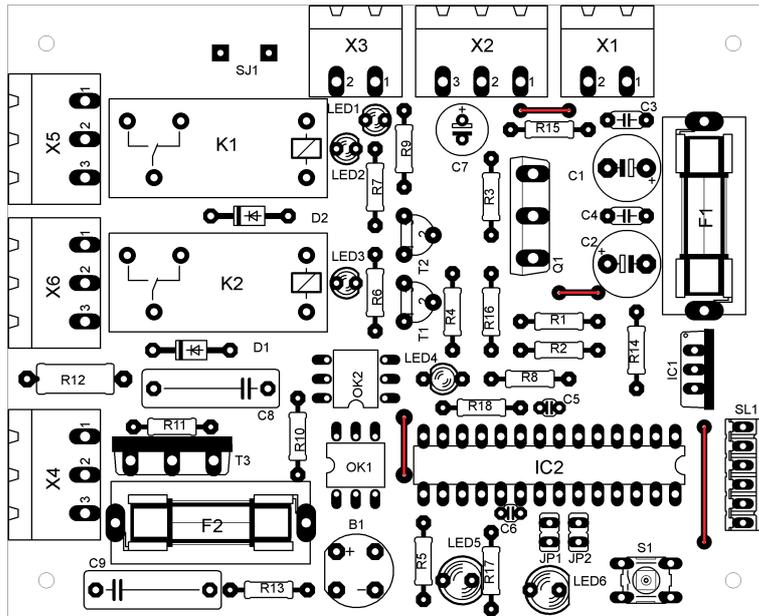


Figura 14: Lato componenti con relativa serigrafia dei componenti.

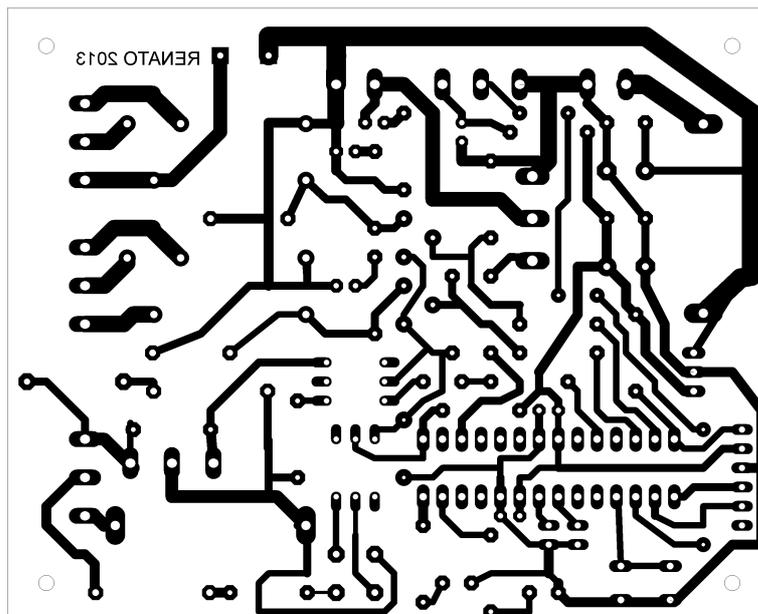


Figura 15: Lato piste del PCB.

A montaggio ultimato la scheda è come riportata in Figura 16.

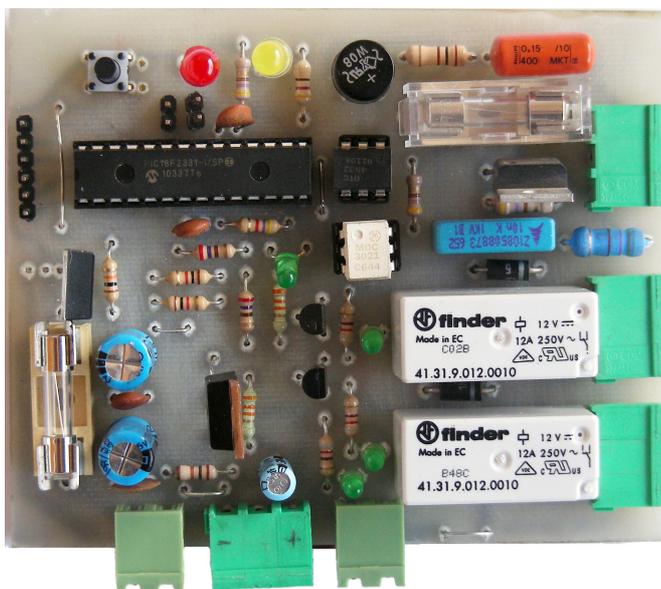


Figura 16: Scheda a montaggio ultimato.

Una volta montato il PCB è necessario montare lo stesso all'interno di una scatola di montaggio al fine di rendere il tutto più robusto e sicuro. Il sistema deve essere montato come riportato in Figura 17.

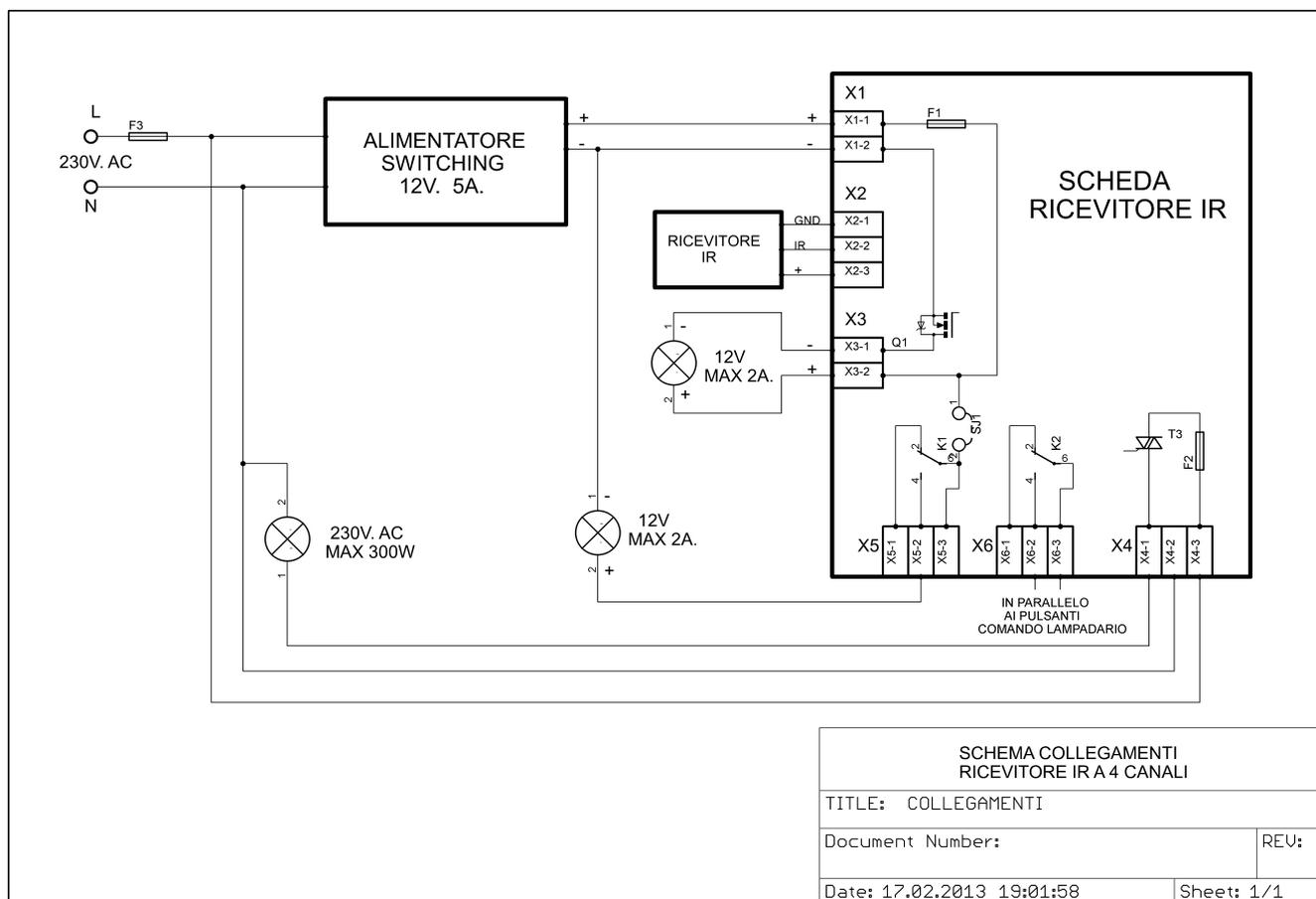
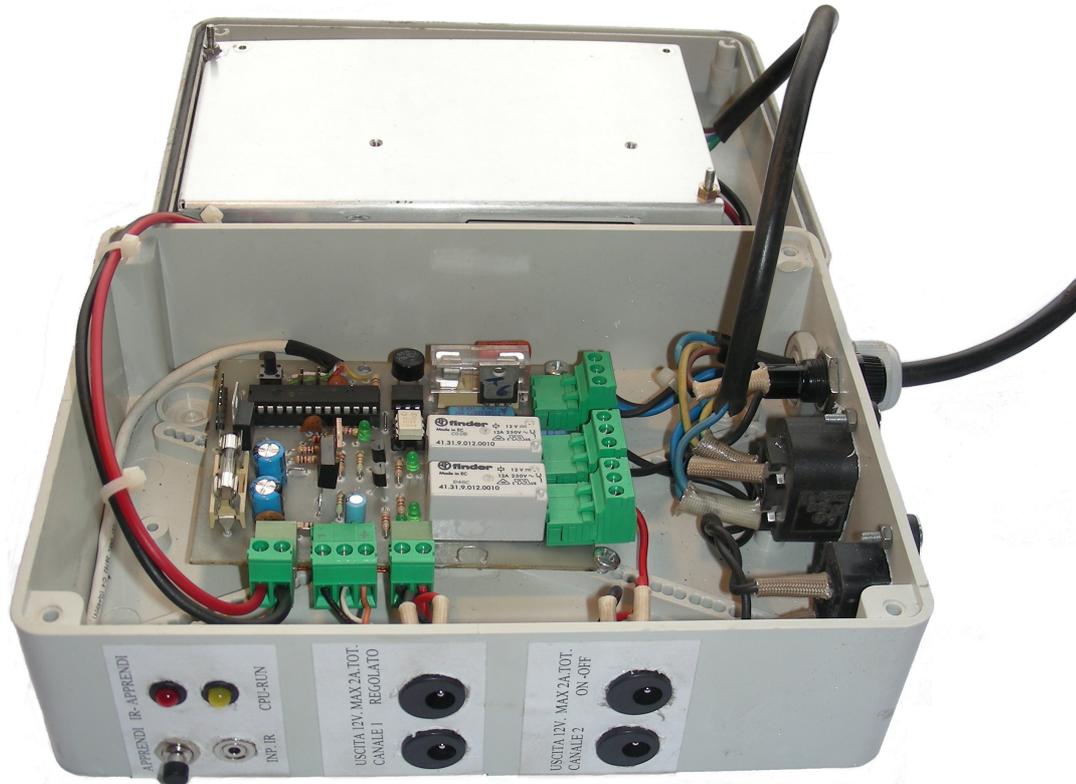


Figura 17: Schema di montaggio del sistema.

Personalmente ho montato il tutto all'interno di una scatola di derivazione per esterni, come riportato in Figura 18 e in Figura 19. In particolare ho incluso il sistema di alimentazione a 12V all'interno della scatola stessa dal momento che ho usato un convertitore AC-DC modulare.



**Figura 18:** Sistema montato all'interno di una scatola di derivazione.



**Figura 19:** Sistema a montaggio ultimato

## Collaudo e messa in funzione



### Nota

Lasciare **sempre disinserita** la morsettiera X4 della tensione di rete 220V.

Prima di procedere alla messa in funzione si raccomanda di pulire bene la base d'appoggio della scheda (che deve essere di materiale isolante) in modo da eliminare eventuali residui di componenti. A montaggio avvenuto, non inserire il PIC nel relativo zoccolo e connettere la morsettiera X1 d'ingresso dell'alimentazione a 12 V e la morsettiera X3 con relativo carico (striscia LED o lampadina 12V 25W max), attenzione non inserire la morsettiera X4 della tensione di rete a 230V. Dare tensione all'alimentatore 12V e misurare con il tester l'uscita del regolatore di tensione a 5V. Misurare anche la presenza di tale tensione tra i piedini 8 (GND) e 7 (+5V.) e tra 19 (GND) e 20 (+5V). A questo punto provare le varie uscite collegate al PIC in modo da rilevare eventuali cortocircuiti o interruzioni del circuito. Questo si può fare con un ponte volante, tramite un pezzettino di cavo rigido, tra la tensione positiva di 5V e i relativi pin dello zoccolo del PIC ai quali è collegato un carico.

I pin in questione sono :

- PIN N° 2 uscita RA0 si deve eccitare il relay del Canale 2 (K1).
- PIN N° 3 uscita RA1 si deve eccitare il relay del Canale 3 (K2).
- PIN N° 16 uscita RC5 si deve accendere il LED rosso (apprendimento).
- PIN N° 12 uscita RC1 deve accendere il LED 1 e la striscia LED (o lampadina a 12V) esterna.
- PIN N° 5 uscita RA3 si deve accendere il LED 4 (uscita Triac).
- PIN N° 15 uscita RC4 si deve accendere il LED giallo (CPU in run).

Se tutto è ok staccare l'alimentazione, attendere qualche secondo in modo che i condensatori si possano scaricare e procedere con il prossimo paragrafo.

## Prima messa in funzione dopo il montaggio

Inserire la morsettiera X2 con il ricevitore IR collegato, ed inserire il PIC nello zoccolo. Aprire il progetto in MPLAB X, collegare il programmatore PICKIT 2/3 alla scheda, ed alimentare il circuito tramite la morsettiera X1.

Scaricare il programma all'interno del PIC, ed in seguito, se non vi sono state segnalazioni d'errore, staccare l'alimentazione a 12 V e togliere il programmatore dalla scheda.

Alimentare il circuito e attendere. Il LED 5 (processore in funzione) deve accendersi per poi spegnersi dopo alcuni secondi e cioè quando il processore andrà in modalità SLEEP.

## Memorizzazione canali

Come già accennato precedentemente il ricevitore è in grado di memorizzare 4 tasti del telecomando TV abbinandoli ad una delle 4 uscite/canali della scheda. Per memorizzare il tasto desiderato sul canale 1 procedere come segue :

- Premere e rilasciare una sola volta il pulsante S1 “Apprendi”. Trascorso qualche secondo dopo il rilascio del pulsante il LED rosso inizierà a lampeggiare indicando che il PIC è pronto a ricevere il segnale IR del tasto del telecomando da memorizzare.
- Premere il tasto del telecomando. Mentre il sistema riceve il segnale IR il LED rosso rimane acceso fisso per poi spegnersi ad apprendimento avvenuto. Se non si preme un tasto del telecomando entro 4 secondi il LED smetterà di lampeggiare indicando che la fase di apprendimento è terminata.

Per memorizzare il secondo canale premere il pulsante due volte consecutive. Per il terzo premere 3 volte e ..indovinate un po’?

...per memorizzare il quarto premere 4 volte consecutive il tasto S1 “Apprendi”.

A questo punto non rimane che premere il tasto del telecomando precedentemente programmato e verificare il funzionamento.

...buona fortuna, e se non funziona...chiedete pure!

Se il funzionamento risulta corretto, togliere l’alimentazione e PRESTANDO LA MASSIMA ATTENZIONE E PRUDENZA ! allacciare la morsettiera X4 inerente l’alimentazione di rete e collegare alla corrispondente uscita la lampada che si desidera pilotare.

Alimentare nuovamente la scheda e collaudare il tutto.

La regolazione di luminosità dei canali 1 e 4 si ottiene mantenendo premuto il tasto del telecomando abbinato al relativo canale. Per accendere o spegnere la lampada premere il tasto solo per qualche istante. L’accensione e lo spegnimento avverranno gradualmente in modo da creare un effetto “soft”.

Per variare la luminosità mantenere premuto il tasto del telecomando. Quando si dà il comando per lo spegnimento viene memorizzato il grado d’intensità luminosa che verrà poi impostato nuovamente al momento di una nuova accensione.

## Funzione test uscite

La scheda dispone di una funzione test uscite. Per attivarla occorre procedere nel seguente modo:

- Togliere l’alimentazione alla scheda e attendere qualche istante.
- Premere, e mantenere premuto, il pulsante S1 “Apprendi” mentre s’inserisce l’alimentazione.
- Quando si accende il LED giallo rilasciare il pulsante.

A questo punto le uscite si attiveranno in sequenza per un secondo circa.

## Messa in funzione ordinaria

Per la messa in funzione ordinaria non ci sono particolari accorgimenti da effettuare.

Basta premere il tasto del telecomando precedentemente memorizzato per attivare la relativa uscita.

## Analisi del Software

Il Firmware è stato scritto in ambiente MPLAB X facendo uso del compilatore XC8 della microchip. Al fine di compilare il codice è necessario installare anche le librerie LaurTec per PIC18 e impostare propriamente i percorsi “include path” nelle opzioni del progetto.

Il programma utilizza i due interrupt presenti all'interno del PIC18F2331. Quello ad alta priorità per rilevare l'arrivo di un segnale IR o la pressione del tasto S1 “Apprendi” mentre quello a bassa priorità, attivato dal Timer 0, per gestire i contatori per la gestione dello Sleep, correggere lo sfasamento del punto zero e per regolare l'attivazione del Triac. Il programma principale inizia impostando le uscite, il Timer 0, il Timer 2, il modulo PWM e le interruzioni.

Il Timer 0 è impostato per lavorare a 8 bit e senza prescaler. Questo vuol dire che attiva l'interrupt a bassa priorità, sul quale è abbinato, ogni 0,128ms.

Infatti lavorando con l'oscillatore interno a 8MHz e ricordando che il tempo d'esecuzione per un'istruzione è di 4 cicli, il Timer viene incrementato ogni  $1/8.000.000 * 4 = 0,5\mu s$

Lavorando a 8 bit il Timer conterà da 0 a 256 e ogni qual volta passerà da 256 a 0 attiverà l'interrupt, cioè ogni  $256 * 0,5 = 0,128ms$ .

Sapendo che la frequenza di rete è di 50Hz e che quindi si ha un ciclo ogni  $1/50 = 0,02$ ” ovvero ogni 20ms, si ottiene che la durata di una singola semionda è pari a 10ms per cui potremo frazionare ogni semionda in  $10/0,128 = 78$  step. Questo ci permetterà di correggere il segnale d'ingresso del fotoaccoppiatore e creare esattamente il punto zero della semionda.( ritardandolo di 27 step tramite la variabile Sfasamento).

Inoltre ci permetterà tramite l'incremento della variabile ContTimer di attivare il Triac in un determinato punto della semionda permettendoci così di regolare la luminosità della lampada ad esso collegata.

Il Timer 0 viene incrementato solo se le uscite sono tutte spente, in particolare quando la variabile ContSleep al superamento del valore di 60.000 (cioè dopo 7,68”) manderà il PIC in modalità Sleep. Lo svolgimento dell'interrupt a bassa priorità è leggermente “lungo” ma nel nostro caso non crea alcun problema.

Il Timer2 viene utilizzato dal PIC per la gestione dell'uscita PWM2 sull'uscita RC1.

Tramite l'istruzione SetDCPWM2 ( ...X..) è possibile regolare la luminosità del carico applicato.

Il valore X può variare da 0 a 1024 in cui con il valore 0 fissa l'uscita sul valore 0, viceversa con il valore 1024 l'uscita è fissa sul valore 1 mentre, con un numero intermezzo, si regolerà il duty cycle del modulo PWM e quindi la luminosità.

L'interruzione ad alta priorità è attivata da un cambio di stato di un ingresso RB4-RB7 per cui viene eseguita in seguito al ricevimento di un segnale IR o alla pressione del tasto S1 “Apprendi”. Al fine di essere il più breve possibile essa attiva solamente i relativi flag.

Il ciclo infinito come indicato successivamente dal diagramma di flusso, è relativamente corto, in quanto dopo aver controllato se sono stati attivati il flag del tasto S1 “Apprendi” o del segnale IR non fa altro che richiamare le relative funzioni; in ultimo controlla se le uscite sono tutte a riposo in modo da mandare la CPU in modalità SLEEP.

Lo svolgimento delle varie funzioni descritte possono essere meglio comprese dal diagramma di flusso riportato in Figura 20.

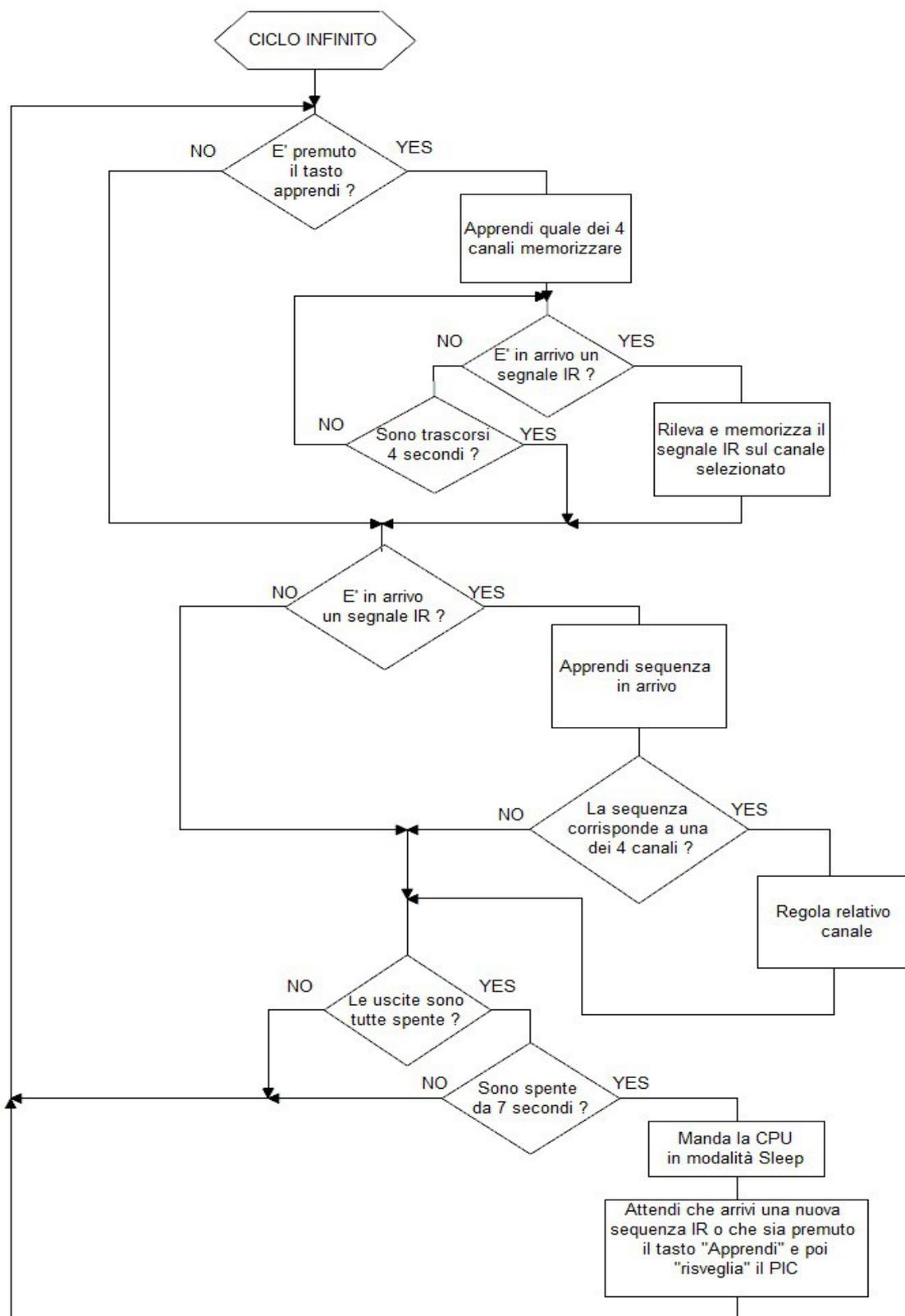


Figura 20: Diagramma di flusso del programma.

L'esecuzione del programma viene svolta dalle seguenti funzioni :

Funzione	Descrizione
ApprendiMemorizzaSequenzaCanale	Memorizza nella EEPROM interna la sequenza Dsi rilevata. L'indirizzo della EEPROM varia a secondo del canale da memorizzare.
RilevaSequenzaIR	Rileva la sequenza IR in arrivo.
ControllaSeEunaSequenza	Verifica se è in arrivo una sequenza IR o se si tratta solo di un singolo impulso IR causato, ad esempio, da eventuali disturbi.
ConfrontaSequenza	Confronta la sequenza ricevuta con quelle memorizzate nella EEPROM.
GestioneUscitaPWM	Gestisce l'accensione e lo spegnimento dell'uscita PWM. Inoltre, in caso di spegnimento, memorizza il grado di luminosità prima di spegnerla in modo da riportarla allo stesso livello ad una successiva attivazione
RegolaLuminositaPWM	Regola la luminosità dell'uscita PWM (Canale 1).
GestioneUscitaTRIAC	Come GestioneUscitaPWM ma per il Canale 4.
RegolaLuminositaTriac	Regola la luminosità dell'uscita a TRIAC (Canale 4)
ContaPremiPulsante	Conta quante volte si preme il tasto "Apprendi" in modo da selezionare il canale da memorizzare.
GestioneTastoTelecMantPremuto	Verifica se il tasto del telecomando viene mantenuto Premuto, ad esempio per variare la luminosità.
TestUscite	Accende e spegne in sequenza tutte 4 le uscite.

La funzione "chiave di questo programma" è quella denominata RilevaSequenzaIR in quanto ad essa è affidato l'incarico di trasformare il codice in arrivo dal telecomando in un array, denominato ContDsiRic[ ], contenente la durata di ogni singolo stato dell'impulso della sequenza che come già detto ho soprannominato *Dsi*. Proprio perché d'importanza rilevante riporto il diagramma di flusso di tale funzione in Figura 21.

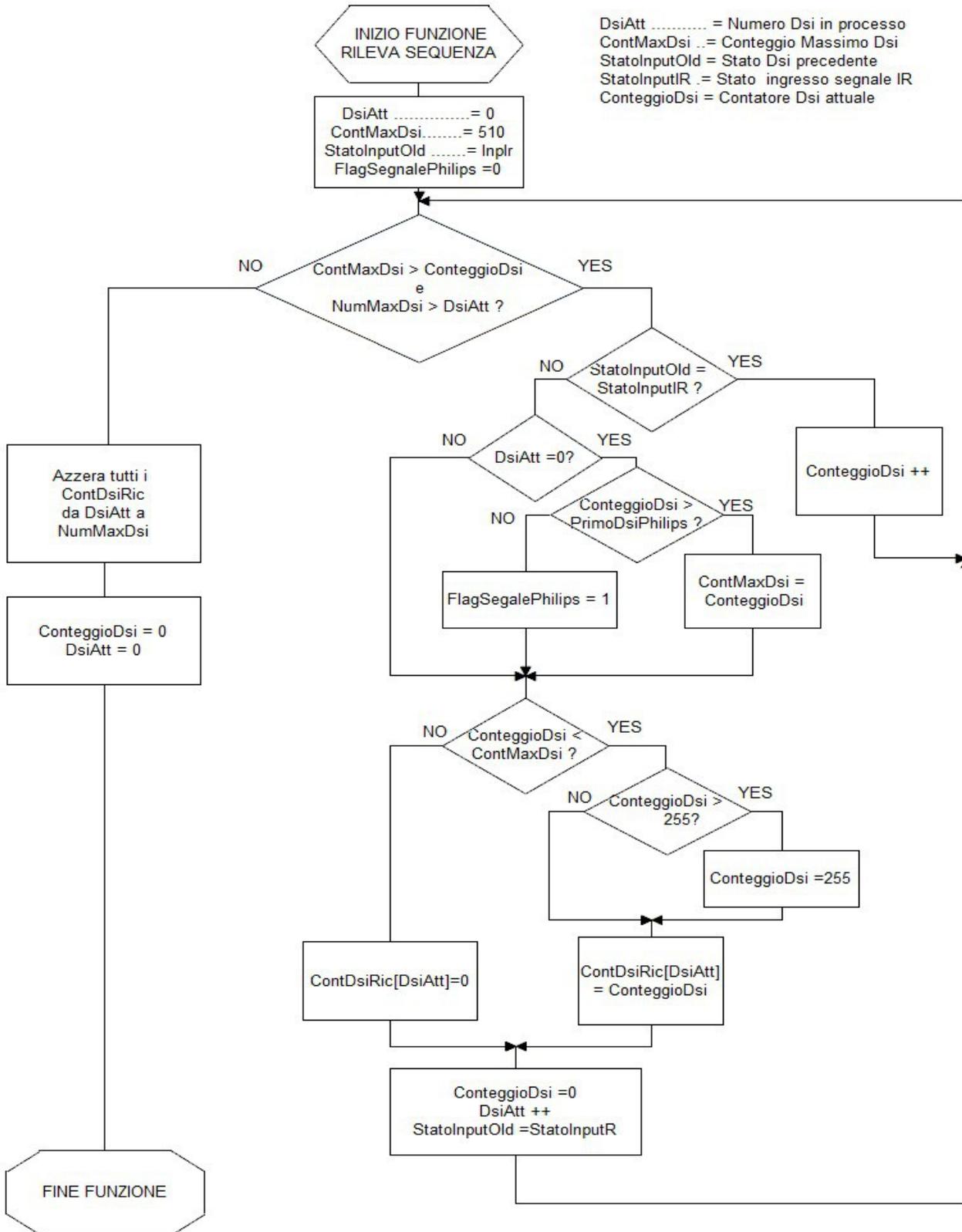


Figura 21: Diagramma di flusso della funzione RilevaSequenzaIR.

Come si evince dal diagramma di flusso, il valore ogni singolo Dsi viene generato ed incrementato non da un Timer interno, bensì dal tempo necessario a svolgere la seguente routine :

```
while ((ContMaxDsi>ConteggioDsi)&& (NumMaxDsi >DsiAtt)){  
    if (StatoInputOld == StatoInputIR){  
        ConteggioDsi ++;  
    }else{  
        //.....;  
    }  
}
```

Siccome questa routine viene interrotta ciclicamente dall'interrupt a bassa priorità, a causa del Timer0, la variabile ConteggioDsi viene incrementata anche nello svolgimento delle istruzioni dell'interrupt stesso.

Di seguito illustro una tabella contenente il valore del *Dsi* rilevata con diversi tipi di telecomandi. Si può notare che il Dsi N°2 e 4 del telecomando Philips riporta due valori in quanto tale trasmissione alterna questi due Dsi ad ogni nuova pressione del tasto premuto.

Per le altre funzioni, ritengo non ci siano particolari rilevanti, e pertanto rimando alla lettura del codice per la comprensione del “lavoro” svolto.

N°	SONY 12bit	SONY 20bit	SAMSUNG	PHILIPS	UNITED	RGB
0	128	130	216	69	255	255
1	21	19	183	31	181	181
2	24	25	22	73/35	23	20
3	22	20	67	34	18	21
4	24	25	21	36/73	22	22
5	20	21	65	34	19	19
6	26	25	24	36	24	25
7	19	20	64	71	19	18
8	24	24	24	72	24	20
9	21	20	19	69	18	21
10	23	24	22	71	23	23
11	23	20	19	35	19	19
12	24	26	23	37	24	24
13	20	21	19	33	18	17
14	25	25	25	37	24	22
15	19	20	19	33	19	20
16	50	25	22	36	25	24
17	20	19	19	34	63	18
18	26	50	23	36	23	21
19	18	20	67	69	66	66
20	24	24	21	37	24	24
21	21	20	66	0	63	63
22	24	51	22	0	24	23
23	23	19	65	0	67	64
24	22	51	25	0	22	24
25	0	19	19	0	65	66
26	0	25	22	0	24	21
27	0	20	19	0	64	18
28	0	50	23	0	23	24
29	0	19	19	0	65	66
30	0	26	24	0	24	21
31	0	20	19	0	64	65
32	0	49	22	0	24	21
33	0	21	19	0	19	65
34	0	49	24	0	23	25
35	0	21	19	0	19	19
36	0	49	22	0	25	20
37	0	21	19	0	18	21
38	0	49	22	0	22	22
39	0	20	67	0	19	66
40	0	50	22	0	23	20
41	0	0	19	0	66	21
42	0	0	24	0	24	23
43	0	0	19	0	19	19
44	0	0	22	0	21	21
45	0	0	20	0	19	20
46	0	0	22	0	25	22
47	0	0	19	0	18	19
48	0	0	25	0	25	25
49	0	0	19	0	63	18
50	0	0	22	0	24	24
51	0	0	65	0	66	6
52	0	0	24	0	23	24
53	0	0	64	0	64	63
54	0	0	24	0	23	23
55	0	0	19	0	66	20
56	0	0	22	0	23	23
57	0	0	65	0	19	63
58	0	0	24	0	24	23
59	0	0	64	0	65	65

Tabella 3: Tabella riassuntiva del valore della variabile Dsi con vari telecomandi.

## Analisi finale

Nel suo insieme questa scheda risulta certamente molto personalizzata ma vuole essere d'aiuto per realizzare altre applicazioni. Potrebbe sicuramente essere realizzata con dimensioni minori, magari con tecnologia SMD ma i miei occhi non lo permettono più.

**Cosa cambierei** Se dovessi rifare lo stampato metterei, al posto del LED rosso, del pulsante e del LED giallo, un connettore per cavo piatto in modo da portare tutte queste periferiche, compreso il jack per l'ingresso del segnale IR, nel luogo più opportuno.

**Curiosità** Durante le prove ho usato dei fotodiodi a 36Khz e a 38Khz e in pratica ho notato solo una leggerissima differenza di funzionamento tra uno e l'altro anche rilevando il segnale di uno stesso telecomando.

**Miglioramenti**

- Rispetto ad una regolazione effettuata con un dimmer tradizionale l'uscita con Triac presenta un leggero tremolio, a bassa luminosità. Questo è dovuto al non sincronismo tra l'ingresso del segnale di "punto zero" e l'attivazione dell'interrupt provocata dal Timer0 che ne rileva il suo ingresso. Si tratta comunque di un leggerissimo tremolio che si nota solamente guardando la lampadina accesa a bassa intensità.
- La sezione a 220V potrebbe richiedere opportuni filtri per limitare disturbi da elettrodomestici.
- Il Software potrebbe essere ottimizzato armonizzando ulteriormente le interruzioni.

Per quanto riguarda la realizzazione del firmware ricordo che non sono un professionista ma solamente un autodidatta che ha imparato a programmare i PIC seguendo il corso di Mauro Laurenti "C18 Step by Step" e quindi ritengo si possa sicuramente migliorare la stesura dello stesso.

Ricordo inoltre che si tratta di un progetto amatoriale e quindi è stato solamente collaudato con i telecomandi che avevo a disposizione in casa, quindi ... penso e ritengo possa funzionare con tutti i telecomandi delle marche sopracitate ma... provare per credere e soprattutto collaudare !

## Ringraziamenti

Vorrei ringraziare Mauro Laurenti per avermi proposto e incoraggiato a scrivere questo articolo. Mi ha fatto tornare indietro di 35 anni..... mi è sembrato di riscrivere un articolo per l'esame di maturità..... speriamo di averlo superato !  
Vorrei anche ringraziare mia moglie che dice sempre : “ Ma cosa stai facendo ?..... Ma chi te lo fa fare ?”..... non ho fatto la spia !!

Grazie a tutti

## Indice Alfabetico

<b>A</b>	NPN.....	8, 10
Alimentazione.....	5	
Alimentazione Triac.....	5	
Apprendi.....	22	
ApprendiMemorizzaSequenzaCanale.....	25	
Assorbimento.....	5	
Assorbimento attivo.....	5	
Assorbimento da spento.....	5	
Assorbimento in standby.....	5	
<b>B</b>		
BC337.....	10	
Bistabile.....	4	
BN0011.....	10	
<b>C</b>		
Canale 1.....	4	
Canale 2.....	4	
Canale 3.....	4	
Canale 4.....	4	
circuito RC.....	12	
ConfrontaSequenza.....	25	
ContaPremiPulsante.....	25	
ConteggioDsi.....	27	
ControllaSeEunaSequenza.....	25	
convertitore AC-DC.....	20	
<b>D</b>		
Dimensioni.....	5	
Dsi.....	10, 25 e segg.	
DVD.....	7	
<b>E</b>		
EEPROM.....	10	
<b>F</b>		
flickering.....	10	
fotoaccoppiatore.....	12	
fotodiode.....	7	
<b>G</b>		
GestioneTastoTelecMantPremuto.....	25	
GestioneUscitaPWM.....	25	
GestioneUscitaTRIAC.....	25	
Graetz.....	10	
<b>I</b>		
IRF3205.....	10	
<b>L</b>		
librerie LaurTec.....	23	
Lista Componenti.....	15	
<b>M</b>		
Monostabile.....	4	
MOSFET.....	4, 10	
MPLAB X.....	21, 23	
<b>N</b>		
<b>P</b>		
PCB.....	19	
Peso.....	5	
Philips.....	7 e segg.	
PIC18.....	23	
PIC18F2331.....	7, 10, 16, 23	
PICKIT 2/3.....	21	
ponte di Graetz.....	10	
Potenza Carico Triac.....	5	
Product Number.....	5	
PWM.....	10	
PWM2.....	16, 23	
<b>R</b>		
RC1.....	23	
RC5.....	8 e segg.	
RC5.....	7	
RegolaLuminositaPWM.....	25	
RegolaLuminositaTriac.....	25	
RESET.....	16	
rete RC.....	12	
RilevaSequenzaIR.....	25	
robot.....	5	
<b>S</b>		
S1.....	22	
S1 "Apprendi.....	22	
SetDCPWM2.....	23	
SIRC.....	7	
SLEEP.....	21	
SMD.....	29	
Sony.....	7	
<b>T</b>		
TestUscite.....	25	
TINA.....	10	
Triac.....	4, 10, 13, 21	
TSOP34836.....	6 e segg., 9 e segg.	
<b>V</b>		
Vbe.....	11	
Versione PCB.....	5	
Videoregistratore.....	7	
<b>X</b>		
X1.....	17	
X2.....	17	
X3.....	17	
X4.....	17	
X5.....	17	
X6.....	17	
<b>Z</b>		
zero crossing.....	10, 12 e segg.	

**Bibliografia**

- [1] [www.LaurTec.it](http://www.LaurTec.it) : Sito dove poter scaricare gli aggiornamenti dell'articolo.
- [2] [www.microchip.com](http://www.microchip.com) : Sito dove scaricare i datasheet del PIC18F2331 e MPLAB-X.
- [3] [www.vishay.com](http://www.vishay.com) : Sito dove scaricare il tasheet del sensore TSOP34836.
- [4] [www.ti.com](http://www.ti.com) : Sito della Texas Instruments dove scaricare il simulatore gratuito TINA.

**History**

<b>Data</b>	<b>Versione</b>	<b>Autore</b>	<b>Revisione</b>	<b>Descrizione Cambiamento</b>
03.05.13	1.0	Ferrero Vercelli Renato	Mauro Laurenti	Versione Originale.