

Apri cancello (versione B) – Progetto di Mattia Ziliani

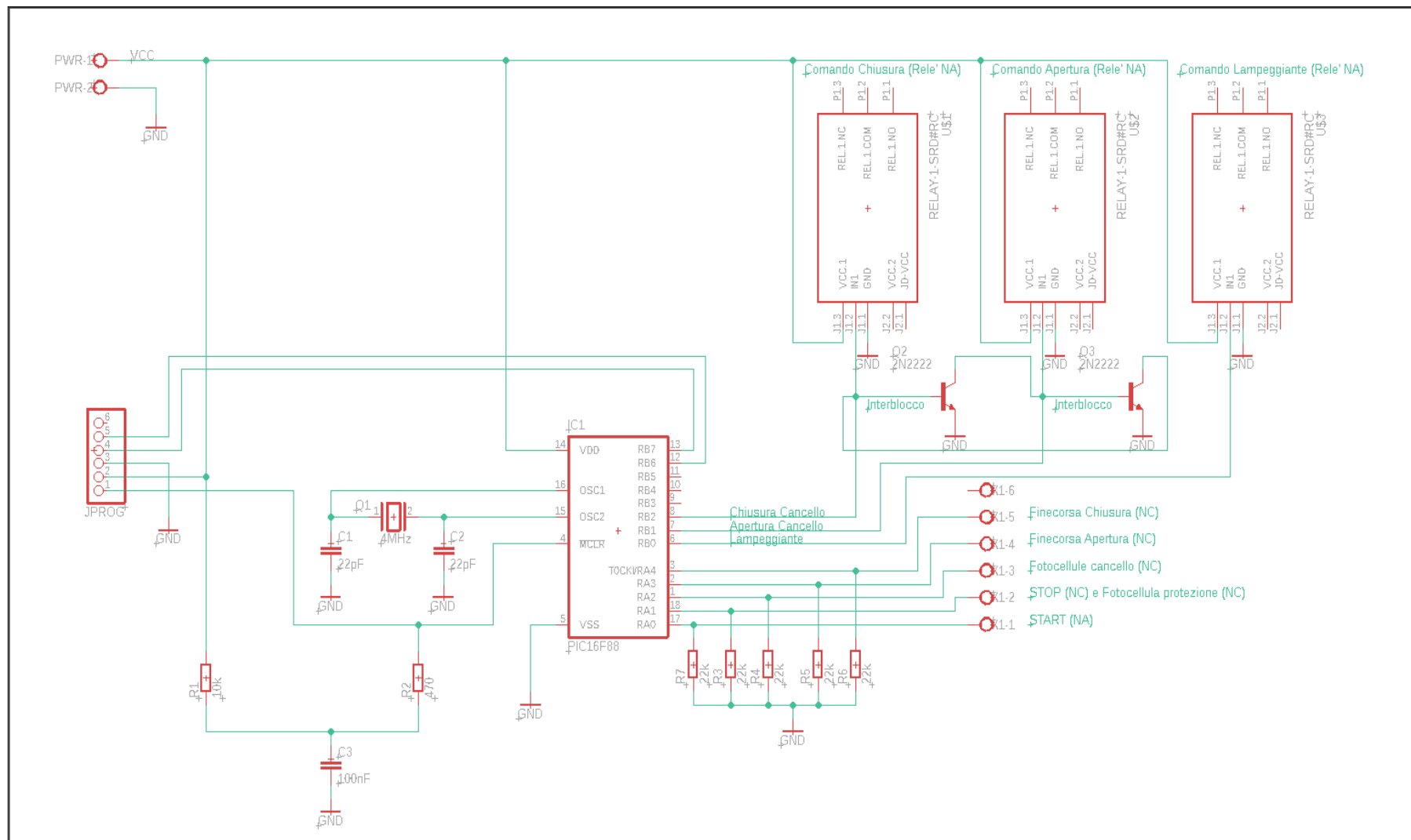
Indice della seguente relazione tecnica:

- [Schema elettrico](#)
- [Specifiche del circuito stampato](#)
- [Descrizione del programma tramite diagramma a stati](#)
- [Codice C del programma per PIC16F88](#)

Una copia della seguente relazione, con i file Eagle e del codice C è disponibile su Github [qui](#).

Apri cancello (versione B) – Progetto di Mattia Ziliani

Schema elettrico:



Apri cancello (versione B) – Progetto di Mattia Ziliani

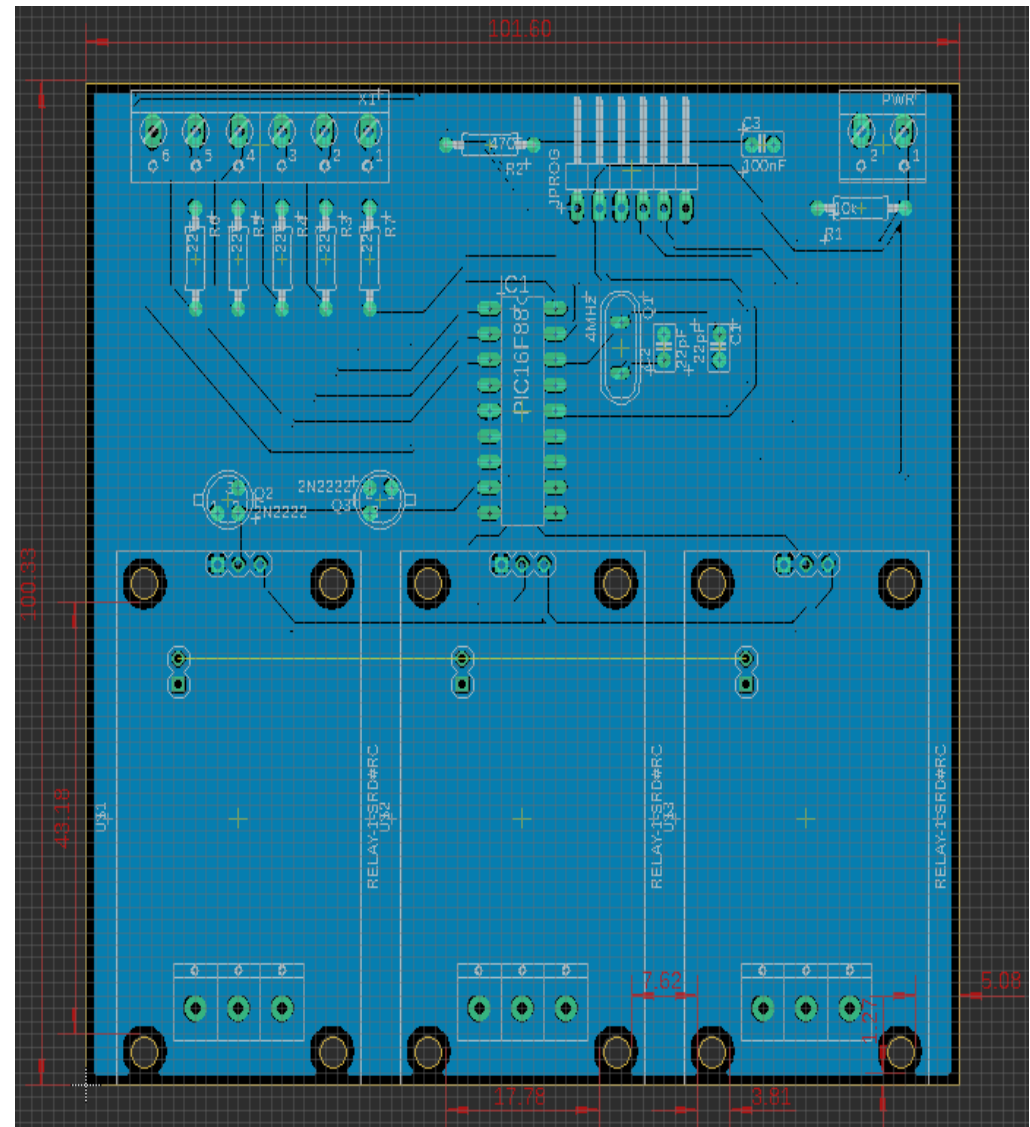
Specifiche del circuito stampato:

Il circuito stampato realizzato usando il programma Autodesk Eagle ha dimensioni di 101,60mm x 100,33mm, presenta 6 fori di montaggio.

I fori di montaggio hanno un diametro di 3,81mm e sono distanziati come nell'immagine, questa disposizione corrisponde ai fori di montaggio dei singoli moduli relè.

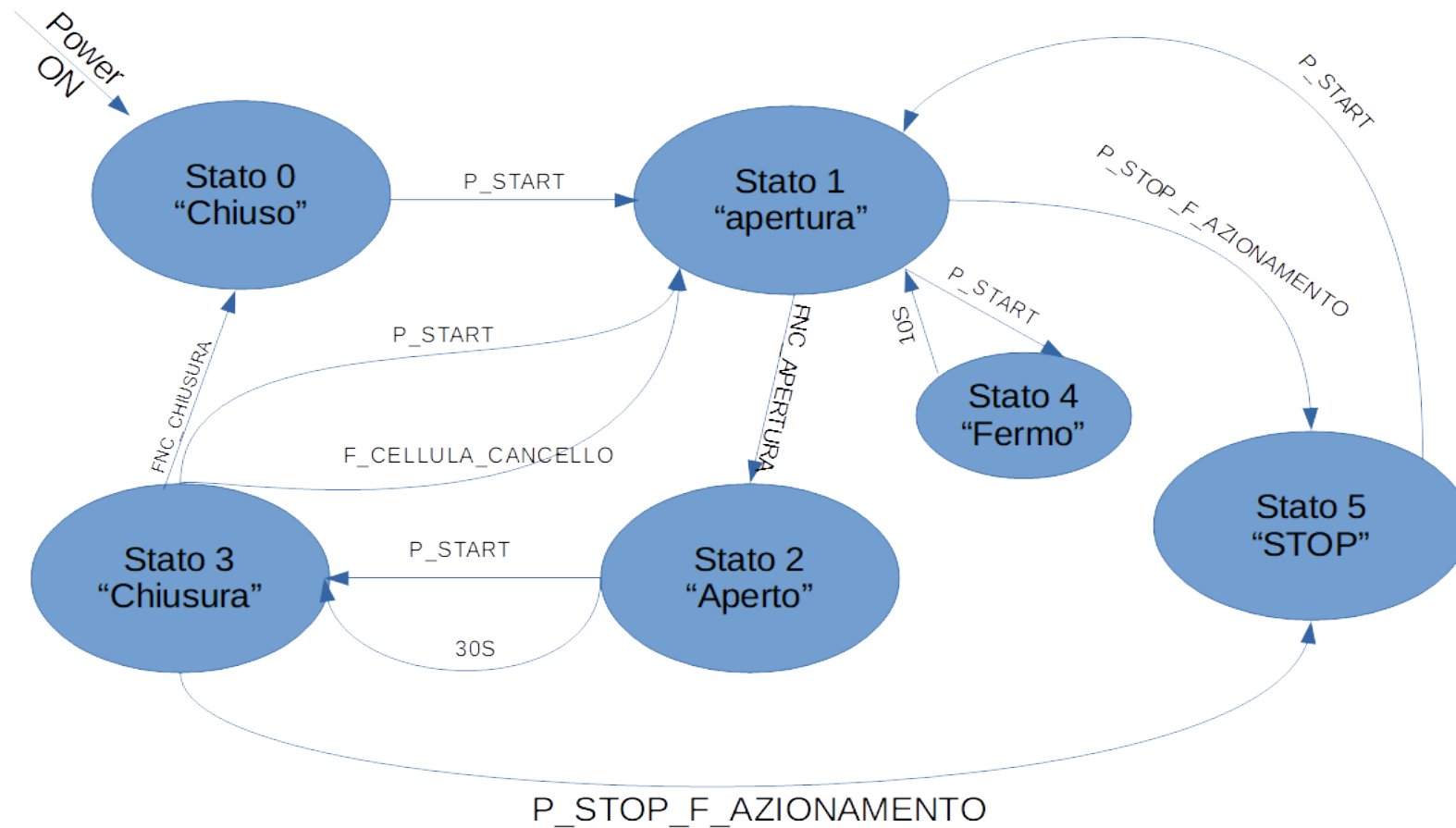
Il circuito stampato presenta una realizzazione a singola faccia.

Le piste di collegamento sono di due dimensioni diverse: 16mil (0,4mm) per i segnali e 32mil (0,8mm) per le alimentazioni.



Apri cancello (versione B) – Progetto di Mattia Ziliani

Descrizione del programma tramite diagramma a stati:



Apri cancello (versione B) – Progetto di Mattia Ziliani

Codice C del programma per PIC16F88:

// ApriCancello_B - Mattia Ziliani

#include <pic.h>

// Configuration bit
__CONFIG(0x3F61);

// Input

#define P_START	RA0	// Pulsante "START" (NA)
#define P_STOP_F_AZIONAMENTO	RA1	// Pulsante "STOP" e Fotocellula zona azionamento (entrambi NC)
#define F_CELLULA_CANCELLO	RA2	// Fotocellula zona cancello (NC)
#define FNC_APERTURA	RA3	// Finecorsa apertura (NC)
#define FNC_CHIUSURA	RA4	// Finecorsa chiusura (NC)

// Output

#define LMP_SEGNALE	RB0	// Controllo Lampeggiante (NA)
#define CMD_APERTURA	RB1	// Controllo motore AC per apertura (NA - CW - Senso orario)
#define CMD_CHIUSURA	RB2	// Controllo motore AC per chiusura (NA - CCW - Senso Antiorario)

// Contatori per timer

unsigned int LMP_TMR0;	// Contatore per tempo Lampeggiante
unsigned int ACT_TMR0;	// Contatore per tempo azioni

// Definizione tempi

#define TEMPO_30_SEC	600
#define TEMPO_10_SEC	200
#define TEMPO_2_SEC	40
#define TEMPO_1_SEC	20

// Variabile di Stato

unsigned char STATO = 0;

void main(void)
{

Apri cancello (versione B) – Progetto di Mattia Ziliani

```
// Inizializzazione
// Tutti gli ingressi digitali
ANSEL = 0;
OPTION_REG = 0;

// 0 = OUTPUT ; 1 = INPUT
TRISA = 0b00011111; // 4 input in banco GPIO A
PORTA = 0;
TRISB = 0b00000000; // 3 output in banco GPIO B
PORTB = 0;

// Abilita Timer0
// -----
// Fosc = 4 MHz
// Fosc/4 = 1,0 MHz
// Prescaler = :256
// F clock ((fosc/4)/Prescaler)= 3906,25 kHz
// T = 1/ f clock = 256 _us
// Periodo = T * 256 = 65,536 ms (intervallo 0 / 256)

T0CS = 0; // clock interno Fosc/4
PSA = 0; // Prescaler assegnato al Timer 0
// Prescaler :256
PS0 = 1;
PS1 = 1;
PS2 = 1;
// Precarica Timer0 in modo da generare interrupt ogni 50 ms (intervallo 60 / 256)
// Periodo = T * 196 = 50,176 ms
TMR0 = 60;

// Attiva interrupt Timer 0
TMR0IE = 1;
// Attiva interrupt globali
GIE = 1;

STATO = 0; //Inizio dallo stato 0, Cancellino chiuso
```

Apri cancello (versione B) – Progetto di Mattia Ziliani

```
//loop
while (1)
{
    //Transizione stati
    switch(STATO)
    {
        case 0: // Cancelli Chiuso
            //Vengono impostati a 0 i due rele' di comando del motore
            CMD_APERTURA = 0;
            CMD_CHIUSURA = 0;

            //Apri il cancello
            if (P_START == 1)
            {
                STATO = 1;                //Passa a stato Apertura cancello
            }
            break;

        case 1: // Apertura cancello
            //Commutazione Lampeggiante segnalazione
            if(LMP_SEGNALAZIONE == 0)
            {
                //Accende Lampeggiante segnalazione
                LMP_SEGNALAZIONE = 1;
                LMP_TMR0 = TEMPO_1_SEC; //Tempo LMP acceso
            }
            else
            {
                //Spegne Lampeggiante segnalazione
                LMP_SEGNALAZIONE = 0;
                LMP_TMR0 = TEMPO_1_SEC; //Tempo LMP spento
            }
    }
}
```

Apri cancello (versione B) – Progetto di Mattia Ziliani

```
//Attiva movimento CW motore AC (apertura cancello)
CMD_APERTURA = 1;
CMD_CHIUSURA = 0;

//blocco cancello per passaggio temporaneo
if (P_START == 1)
{
    STATO = 4;                                //Passa a fermo temporaneo cancello
}

//Stop movimento
if (P_STOP_F_AZIONAMENTO == 0)
{
    STATO = 5;                                //Blocco sistema
}

//Finecorsa cancello aperto attivato
if (FNC_APERTURA == 0)
{
    //Viene fermato l'azionamento del motore AC
    CMD_APERTURA = 0;
    CMD_CHIUSURA = 0;

    LMP_SEGNALAZIONE = 0;                    //Spegni Lampeggiante segnalazione
    STATO = 2;                                //Passa a stato Aperto
}
break;

case 2: // Cancelllo aperto
//Chiudi il cancello
if (P_START == 1)
{
    STATO = 3;                                //Passa a stato Chiusura cancello
}
}
```


Apri cancello (versione B) – Progetto di Mattia Ziliani

```
//Chiusura automatica cancello
if (ACT_TMR0 == 0)
{
    ACT_TMR0 = TEMPO_30_SEC;          // Tempo Chiusura automatica
    STATO = 3;
}
break;

case 3: // Chiusura cancello
//Commutazione Lampeggiante segnalazione
if(LMP_SEGNALAZIONE == 0)
{
    //Accende Lampeggiante segnalazione
    LMP_SEGNALAZIONE = 1;
    LMP_TMR0 = TEMPO_1_SEC;          //Tempo LMP acceso
}
else
{
    //Spegne Lampeggiante segnalazione
    LMP_SEGNALAZIONE = 0;
    LMP_TMR0 = TEMPO_1_SEC;          //Tempo LMP spento
}

//Attiva movimento CCW motore AC (chiusura cancello)
CMD_APERTURA = 0;
CMD_CHIUSURA = 1;

//Ri-apertura cancello
if (P_START == 1 || F_CELLULA_CANCELLO == 0)
{
    //Viene fermato l'azionamento del motore AC
    CMD_APERTURA = 0;
    CMD_CHIUSURA = 0;
}
```

Apri cancello (versione B) – Progetto di Mattia Ziliani

```
//Inverti movimento
if (ACT_TMR0 == 0)
{
    ACT_TMR0 = TEMPO_2_SEC;          //Tempo Inversione
    STATO = 1;                       //Passa a stato Apertura cancello
}

//Stop movimento
if (P_STOP_F_AZIONAMENTO == 0)
{
    STATO = 5;                       //Blocco sistema
}

//Finecorsa cancello chiuso attivato
if (FNC_CHIUSURA == 0)
{
    //Viene fermato l'azionamento del motore AC
    CMD_APERTURA = 0;
    CMD_CHIUSURA = 0;

    LMP_SEGNALAZIONE = 0;           //Spegni Lampeggiante segnalazione
    STATO = 0;                     //Passa a stato Chiuso
}
break;

case 4: // Fermo temporaneo
//Viene fermato l'azionamento del motore AC
CMD_APERTURA = 0;
CMD_CHIUSURA = 0;

//Aspetta
if (ACT_TMR0 == 0)
{
    ACT_TMR0 = TEMPO_10_SEC;        //Tempo Passaggio temporaneo
    STATO = 1;                     //Passa a stato Apertura cancello
}
```

Apri cancello (versione B) – Progetto di Mattia Ziliani

```
    }
    break;

case 5: // STOP
    //Viene fermato l'azionamento del motore AC
    CMD_APERTURA = 0;
    CMD_CHIUSURA = 0;

    //Accende Lampeggiante segnalazione
    LMP_SEGNALAZIONE = 1;

    //Ri-apertura cancello
    if (P_START == 1)
    {
        STATO = 1;
        //Passa a stato Apertura cancello
    }
    break;

default:
    STATO = 0;
    break;
}

}

}

//funzione per uso interrupt
void interrupt irq()
{
    // Timer 0
    if (TMR0IF == 1)
    {
        // Flag Interrupt Timer0
        TMR0IF = 0;
    }
}
```

Apri cancello (versione B) – Progetto di Mattia Ziliani

```
// Precarica Timer 0 in modo da generare interrupt ogni 50 ms (intervallo 60 / 256)
```

```
// Periodo =  $T * 196 = 50,176$  ms
```

```
TMR0 = 60;
```

```
// Decremento variabile contatori
```

```
if (LMP_TMR0 > 0)
```

```
    LMP_TMR0--;
```

```
if (ACT_TMR0 > 0)
```

```
    ACT_TMR0--;
```

```
}
```

```
}
```