

Microprocesoare și microcontrolere

Proiect

Adascalitei Teodora-Florentina

Grupa:5302

1.Introducere:

Enunt: Tema3

Sa se genereze pe un pin notele (min. 16 note) dintr-o melodie. (se poate alege sa se cante imnul RO). Indicație: frecv. tact se va alege nu de 4 MHz! (Frecventa=16Mhz)La apasarea unui buton (sau a unor butoane) se schimba ritmul melodiei (minim 2 ritmuri).

Descriere:

Pentru proiectul meu am ales muzica "La multi ani". Fiecare notă muzicală se generează folosind PWM cu un factor de umplere de 50%, unde semnalul trece rapid între starea ON și OFF. Acest factor de umplere reprezintă procentul de timp în care semnalul este ON într-un ciclu. Frecvența semnalului PWM este ajustată pentru a corespunde frecvenței notei dorite. Sunetul este obținut prin repetarea semnalului PWM de mai multe ori, în funcție de durata dorită, atunci când un buton este apăsat.

Tabel cu notele muzicale, frecventele notelor si perioada lor:

Nota muzicala	Frecventa notei(Hz)	Perioada notei(us)
DO	261.6~262	3816
RE	293.7~294	3401
MI	329.6~330	3030
FA	349.2~349	2865

Calcul pentru calcularea notei si realizarea repetarii unei note:

2.Cod c:

```
#include<htc.h>

#define _XTAL_FREQ 16000000

void main(void)

{

unsigned i;

TRISB=0b00000001;//rb0 pin de intrare;rb3 pin de iesire

ANSELH=0;

ANSEL=0;

while(1)

{

if(RB0==1)//buton apasat

{

//DO(2sec)+DO(1sec)=3*262=786 repetari

for(i=0;i<786;i++)

{

RB3=1; __delay_us(1908);

RB3=0; __delay_us(1908);

}

//RE-1sec=294 repetari

for(i=0;i<294;i++){

RB3=1; __delay_us(1700);

RB3=0; __delay_us(1700);

}

//DO(1sec)=262 repetari

for(i=0;i<262;i++)

{

RB3=1; __delay_us(1908);

RB3=0; __delay_us(1908);

}

//RE-1sec=294 repetari
```

```

for(i=0;i<294;i++){
RB3=1; __delay_us(1700);
RB3=0; __delay_us(1700);
}
//MI(2sec)+Mi(1sec)=Mi(3sec)=3*330=990repetari
for(i=0;i<990;i++){
RB3=1; __delay_us(1515);
RB3=0; __delay_us(1515);
}
//RE-1sec=294 repetari
for(i=0;i<294;i++){
RB3=1; __delay_us(1700);
RB3=0; __delay_us(1700);
}
//Mi(1sec)=330repetari
for(i=0;i<330;i++){
RB3=1; __delay_us(1515);
RB3=0; __delay_us(1515);
}
//FA(1sec)=349repetari
for(i=0;i<174;i++){
RB3=1; __delay_us(1432);
RB3=0; __delay_us(1432);
}
//MI(2sec)+Mi(1sec)=Mi(3sec)=3*330=990repetari
for(i=0;i<990;i++){
RB3=1; __delay_us(1515);
RB3=0; __delay_us(1515);
}
//RE-1sec=294 repetari
for(i=0;i<294;i++){

```

```

RB3=1; __delay_us(1700);
RB3=0; __delay_us(1700);
}
//DO(1sec)=262 repetari
for(i=0;i<262;i++)
{
    RB3=1; __delay_us(1908);
    RB3=0; __delay_us(1908);
}
//RE-1sec=294 repetari
for(i=0;i<294;i++){
    RB3=1; __delay_us(1700);
    RB3=0; __delay_us(1700);
}
//MI(2sec)=660repetari
for(i=0;i<990;i++){
    RB3=1; __delay_us(1515);
    RB3=0; __delay_us(1515);
}
}else RB3=0;__delay_ms(250);
}
}

```

3.Pseudocod:

Start

Configurare(initializare registri)

Cat timp 1 executa

Daca RB0=1 atunci

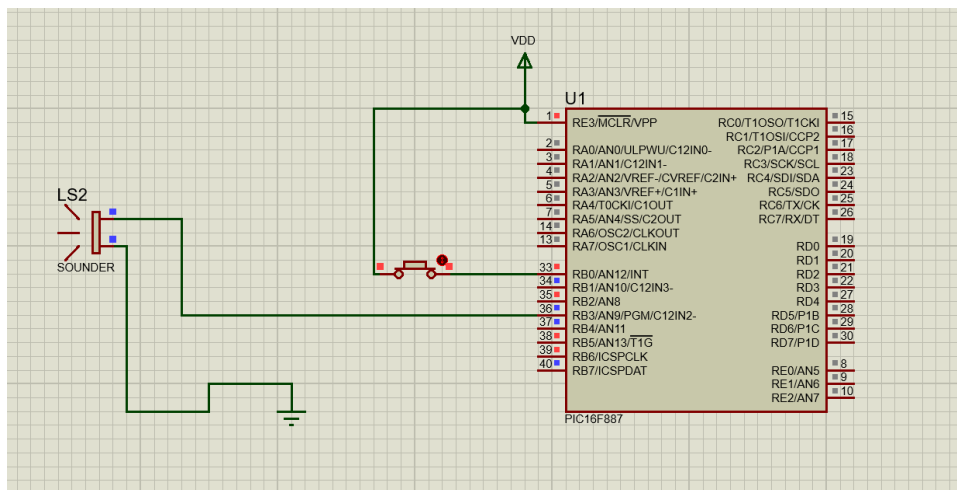
“se vor genera notele muzicale”

Altfel

“nu se vor genera notele muzicale”

stop

4.Schema electrica:



5.Bucla delay in asm:

```
#include p16f887.inc
```

```
i equ 0x20
```

```
j equ 0x21
```

```
main:
```

```
bcf STATUS,RP1
```

```
BSF STATUS,RP0
```

```
movlw 0xFF
```

```
    movwf TRISB
```

```
bcf STATUS,RP1
```

```
Bcf STATUS,RP0
```

```
loop:
```

```
bsf PORTB,0
```

```
CALL delay_1.7ms
```

```
bcf PORTB,0
```

```
CALL delay_1700us
```

```
goto loop
```

```
delay_34ms:
```

```
movlw d'2710'
movwf i
loop_34ms:
nop
nop
decfsz i,1
goto loop_34ms
return
delay_1.7ms:
movlw d'17'
movwf j
loop_1.7ms:
nop
nop
nop
nop
call delay_34ms
decfsz j,1
goto loop_1.7ms
return
delay_1700us:
movlw d'17'
movwf j
loop_1700us:
nop
nop
nop
nop
call delay_34ms
decfsz j,1
goto loop_1700us
```

return

end