



Escola Politècnica Superior
d'Enginyeria de Manresa

UNIVERSITAT POLITÈCNICA DE CATALUNYA

PRÀCTICA 2:

SUBMOSTREIG

CALIBRACIÓ DE LA

FREQÜÈNCIA DEL ARDUINO

Sergi Carol Bosch i Enric Lenard Uró

Grau en Enginyeria de Sistemes TIC

Processament Digital del Senyal

Curs 2014-15, Grup 10, G11 de pràctiques

Realització de la pràctica: 08/10/2014

Lliurament del treball: 28/10/2014

Memòria de la Pràctica 2

1 Trobar el clk del Arduino

Tasca 1

En aquest primer apartat intentarem trobar quin és el verdader clock del nostre Arduino. Per fer-ho utilitzarem els alies de la freqüència que entrem a l'Arduino, en aquesta pràctica seran 20 KHz.

Primer de tot compilarem el programa que sens ha facilitat a classe, hi el carregarem a l'Arduino. A continuació ficarem un generador d'ones a 20 KHz. Si el clock de l'Arduino fos perfecte la freqüència de sortida seria gairebé 0 degut a que a 20KHz tindríem un alies. També connectarem un RC a la pota 11 de l'Arduino de valor 8nF i una resistència de 1.2K Ω

Fent aquesta prova vam descobrir que el clock del nostre arduino estava desplaçat 11 Hz . Donant que la freqüència de sortida de l'Arduino era de 19989 Hz, i no els 20 KHz esperats.

Aquí es pot observar la freqüència obtinguda i la seva forma.



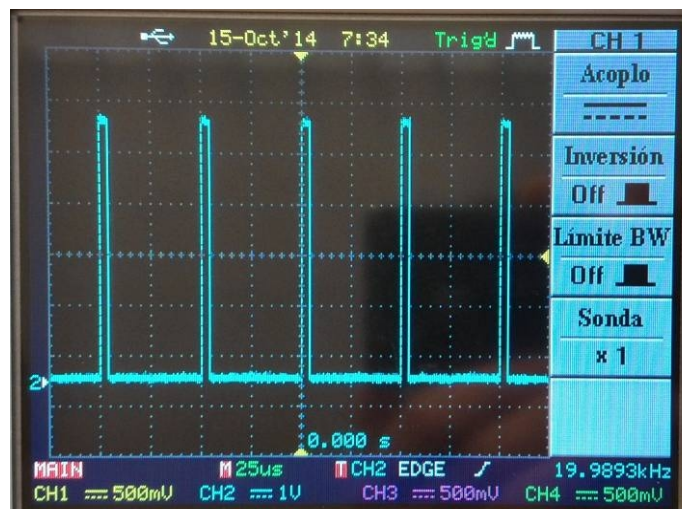
Si calculem el pas per milió ens dona el següent resultat.

$$((19989 - 20000)/20000) * 10^6 = -550$$

Tasca 2

A continuació, farem el mateix experiment anterior però de forma automàtica. Això vol dir que l'Arduino anirà canviant la seva freqüència automàticament per tal de poder veure els canvis al oscil·loscopi sense la necessitat de tenir que canviar el valor del generador d'ones.

Per fer-ho canviarem el timer original, el TIMER 0 , pel TIMER 1, ja que ens serà més fàcil treballar amb aquest. Seguidament, canviem la interrupció del timer que es troba al mòdul **main**, hi fem que cada segon incrementi en 1 el valor de OCR1A, que hem inicialitzat anteriorment a 790 i fem que arribi fins a 810. D'aquesta forma, l'Arduino canvia de forma automàtica el límit del comptador d'interrupció i, per tan, canvia la freqüència de sortida.



En la imatge superior podem veure la sortida de l'Arduino en l'oscil·loscopi. En aquest cas, veiem la freqüència a 19.98 KHz, aquesta freqüència anirà canviant de forma automàtica.

Codi: Main_T2.c

Tasca 3

Per últim, farem un codi que donada una entrada de 20 KHz permeti que l'Arduino mateix calculi la freqüència en que es troba l'altres i per tan mostri pel port sèrie el clock real de l'Arduino de forma automàtica.

Per tal de fer això, el que farem serà crear una variable anomenada "clocks" en la qual hi anirem sumant 1 sempre hi quant ho indiqui la maquina d'estats que hem creat. Aquesta maquina d'estats llegeix el valor del conversor D/A, si aquest valor esta a la meitat (127), amb un possible marge, i tenim l'estat en positiu, sumem 1 a la variable clocks. En el cas que estiguem en estat negatiu i el valor del conversor sigui més gran que 127 més un marge, també li sumarem 1. També especificarem un clock mínim que inicialment valdrà 0xFFFF.

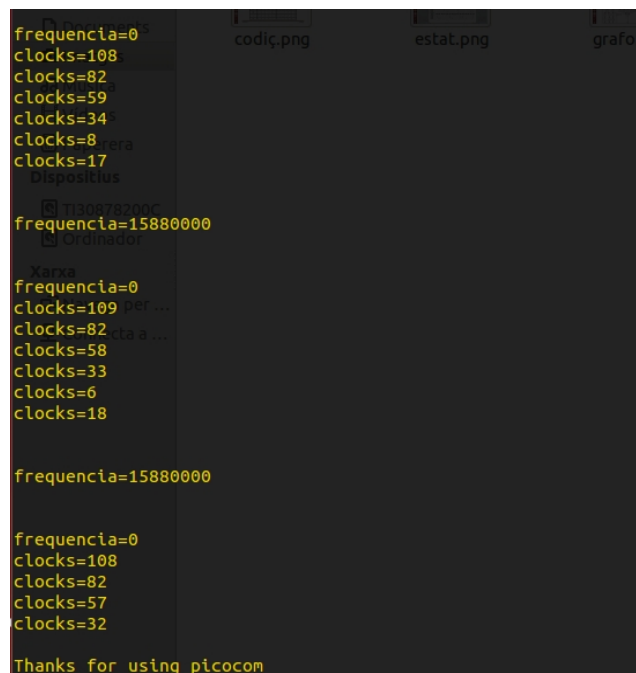
Quant el número de la variable clocks sigui més gran que el clock mínim, llavors voldrà dir que estem a la freqüència mínima i per tan l'ensenyarem per pantalla. Per calcular la freqüència és tan simple com agafar el valor de la freqüència que li entrem a l'Arduino (20000 Hz) i multiplicar-ho per el valor del OCR1A.

A continuació també calcularem la diferencia de freqüència amb l'original. Per fer-ho simplement restem la freqüència obtinguda anteriorment amb la freqüència "real" (16MHz) del clock de l'Arduino.

Un cop realitzats aquets càlculs tornarem al inici, ficant el clock mínim a 0xFFFF i posant el valor de OCR1A a 790.

En el cas de que la variable clocks no superi el valor mínim, assignarem el valor de la variable clocks a el clock mínim i sumarem 1 a OCR1A per tal de canviar de freqüència tal i com hem fet en la tasca anterior.

Finalment aquest és el resultat que obtenim per pantalla.



```
frecuencia=0
clocks=108
clocks=82
clocks=59
clocks=34
clocks=8
clocks=17
Dispositius
[?] trans2000
frecuencia=15880000
[?] trans2000
Xarxa
frecuencia=0
clocks=109
clocks=82
clocks=58
clocks=33
clocks=6
clocks=18

frecuencia=15880000

frecuencia=0
clocks=108
clocks=82
clocks=57
clocks=32
Thanks for using picocom
```

Com a comentari per tal de millorar la dinàmica de treball a l'hora de realitzar la pràctica, simplement mencionar que amb tasques d'aquest estil ajuda molt tenir una guia la qual et permet saber quins són els passos que s'han de seguir en tot moment, i d'aquesta manera, poder aprofitar les seccions de pràctiques amb més eficiència.

Per la resta, creiem que esta tot correcte, ja que veiem a pràctiques allò que hem estudiat a teoria, a la vegada que adquirim coneixements del material que ens envolta, com aquest cas l'Arduino.