

Nom i Cognoms: \_\_\_\_\_

**Justifiqueu totes les respostes!**

1) Disposem d'un microcontrolador PIC18F45K22 funcionant a 4MHz i volem configurar el convertidor analògic digital de 10 bits en el seu punt més òptim de funcionament (temps mínims tot respectant les restriccions del fabricant) (2,5punts)

1.1- Quina seria la configuració dels bits ADCS i ACQT que utilitzaries?

Segons la documentació subministrada ens quedem amb els valors més propers a les especificacions més restrictives del fabricant, es a dir,  $TAD \geq 1\mu\text{seg}$  i  $TACQ \geq 7,45\mu\text{seg}$ .

Per aconseguir un  $TAD \geq 1\mu\text{seg}$  hem de dividir la freqüència de l'oscil·lador entre 4 obtenint un  $TAD = 1\mu\text{seg}$ . Vàlid segons les especificacions. La configuració corresponent de ADCS seria 100.

Seguint el mateix raonament, hem de multiplicar per 8 el TAD obtingut per superar els  $7,45\mu\text{seg}$  de restricció. La configuració corresponent en del ACQT = 100

1.2- Segons la configuració triada, quin seria el temps que trigaria el PIC en obtenir una mostra?

$\text{Temps total} = T_{acq} + 12 TAD = 8\mu\text{seg} + 12 * 1\mu\text{seg} = 20\mu\text{seg}$ .

1.3- Suposant que un cop obtinguda la mostra, el programa triga 10 microsegons en tractar-la, quina seria la freqüència màxima del senyal que podem mostrejar?

$\text{Temps total en adquirir i tractar la mostra} = 10\mu\text{seg} + 20\mu\text{seg} = 30\mu\text{seg}$ .

$T_{\text{mostreig}} = 30\mu\text{seg}$

$\text{Freq. Mostreig} = 1/30\mu\text{s} = 33,3\text{KHz}$

Con que segons el teorema de Nyquist necessitem mostrejar com a mínim al doble de la freqüència del senyal que volem mostrejar, voltra dir que la freqüència màxima del senyal a digitalitzar haurà de ser la meitat de la freqüència de mostreig, es a dir  $16,6\text{KHz}$ .

No obstant, observem que el temps que triguem en adquirir una mostra es superior al temps de tractar-la i per tant seria possible paral·lelitzar el procés. En aquest cas el

$T_{\text{mostreig}} = 20\mu\text{seg}$  i la  $\text{Frq. Mostreig} = 50\text{KHz}$

Obtenint una freqüència màxima del senyal =  $25\text{KHz}$ .

2) Volem mostrejar el senyal analògic provinent d'un sensor de temperatura connectat a un PIC18F45K22. El sensor es comporta de manera LINEAL i dóna tensions entre 0V i 5V corresponents a temperatures d'entre 0 i 100 graus. Per restriccions del problema, necessitem mesurar temperatures entre 20 i 60 graus i volem tenir la màxima resolució possible (2 punts)

2.1 Quines tensions de referència triaries?

Com que el sensor ens dona unes tensions lineals entre 0V i 5V corresponents a una temperatures de entre 0º i 100º voldrà dir que tenim un increment de 1V per cada 20 º.

Com que l'enunciat ens demana que volem màxima resolució, hem de fixar les tensions de referència als valors líndars de la mínima i màxima mesura que volem obtenir, es a dir, entre 20º i 60º, obtenint que  $V_{Ref-} = 1V$  i  $V_{Ref+} = 3V$ .

2.2 Quina es la resolució en graus que tindrem?

$Res = Rang \text{ representat en graus} / \text{subdivisions amb 10 bits} =$

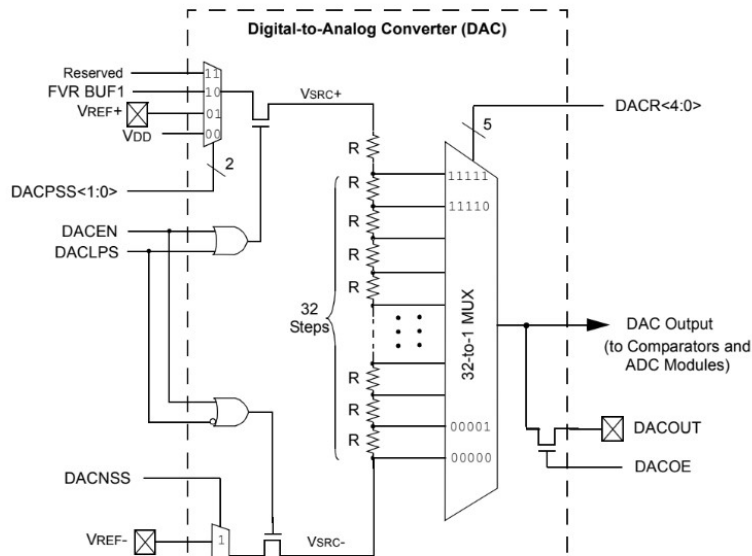
$$60-20 / (2^{10}) =$$

$$40 / 1024 = 0,39 \text{ graus.}$$

3) Necessitem fer servir el **convertidor D/A** del PIC18FK22 a una freqüència elevada i un alumne amic nostre de telecos ens ha recomanat que fem servir la freqüència de clock més alta possible per tal de que el temps de conversió **D/A** es minimitzi. Té raó el nostre amic? (1 punt)

El nostre amic de telecos no te raó. Contràriament al que passa amb el convertidor A/D que requereix una maquina seqüencial per implementar el mètode d'aproximacions successives que depèn de la freqüència de l'oscil·lador principal, el convertidor D/A fa la conversió amb resistències i la velocitat de conversió no depèn de la freqüència de l'oscil·lador.

Nom i Cognoms: \_\_\_\_\_

**22-1: DIGITAL-TO-ANALOG CONVERTER BLOCK DIAGRAM****REGISTER 17-3: ADCON2: A/D CONTROL REGISTER 2**

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	—	ACQT<2:0>			ADCS<2:0>		
bit 7							bit 0

**Legend:**

R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0'  
 -n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown

- bit 7      **ADFM:** A/D Conversion Result Format Select bit  
 1 = Right justified  
 0 = Left justified
- bit 6      **Unimplemented:** Read as '0'
- bit 5-3      **ACQT<2:0>:** A/D Acquisition time select bits. Acquisition time is the duration that the A/D charge holding capacitor remains connected to A/D channel from the instant the GO/DONE bit is set until conversions begins.  
 000 = 0<sup>(1)</sup>  
 001 = 2 T<sub>AD</sub>  
 010 = 4 T<sub>AD</sub>  
 011 = 6 T<sub>AD</sub>  
 100 = 8 T<sub>AD</sub>  
 101 = 12 T<sub>AD</sub>  
 110 = 16 T<sub>AD</sub>  
 111 = 20 T<sub>AD</sub>
- bit 2-0      **ADCS<2:0>:** A/D Conversion Clock Select bits  
 000 = F<sub>osc</sub>/2  
 001 = F<sub>osc</sub>/8  
 010 = F<sub>osc</sub>/32  
 011 = F<sub>RC</sub><sup>(1)</sup> (clock derived from a dedicated internal oscillator = 600 kHz nominal)  
 100 = F<sub>osc</sub>/4  
 101 = F<sub>osc</sub>/16  
 110 = F<sub>osc</sub>/64  
 111 = F<sub>RC</sub><sup>(1)</sup> (clock derived from a dedicated internal oscillator = 600 kHz nominal)

**TABLE 27-22: A/D CONVERSION REQUIREMENTS PIC18(L)F2X/4XK22**

Standard Operating Conditions (unless otherwise stated)							
Operating temperature      Tested at +25°C							
Param. No.	Symbol	Characteristic	Min	Typ	Max	Units	Conditions
130	T <sub>AD</sub>	A/D Clock Period	1	—	25	μs	-40°C to +85°C
			1	—	4	μs	+85°C to +125°C
131	T <sub>CNV</sub>	Conversion Time (not including acquisition time) (Note 1)	11	—	11	T <sub>AD</sub>	
132	T <sub>ACQ</sub>	Acquisition Time (Note 2)	1.4	—	—	μs	V <sub>DD</sub> = 3V, R <sub>S</sub> = 50Ω
135	T <sub>SWC</sub>	Switching Time from Convert → Sample	—	—	(Note 3)		
136	T <sub>DIS</sub>	Discharge Time	1	—	1	T <sub>CY</sub>	

Supposem:  $T_{AD} \geq 1 \mu s$ ,  $T_{ACQ} \geq 7,45 \mu s$ ,  $T_{DIS} = 1 T_{AD}$

4) Indica per cada un dels següents busos, si hi ha algun mecanisme per detectar si està connectat o no l'equip amb que estem comunicant-nos (1,5 punts):

I2C: El Bus I2C reserva un temps de bit, després de treure l'adreça del dispositiu esclau i la ordre d'operació, per rebre un ACK indicant que està present i a disposició.

SPI: El Bus SPI no té cap mecanisme de verificació per detectar si hi ha un esclau present a l'altra banda. Genera clocks i envia/rep "dades", pot enviar les dades a ningú o rebre senyals del bus sense sentit.

1Wire: El 1 Wire comprova a cada transacció si hi ha un esclau connectat. Després del pols de reset deixa la línia a '1' feble perquè l'esclau es carregui elèctricament i indici amb una baixada a 0 la seva presència.

5) Calcula el temps necessari per transmetre 2048 Bytes de dades per línia sèrie a 19200 bps utilitzant bit de paritat (parell), suposant que els enviarem en blocs de 256 Bytes (un Bank de memòria). Després de cada un dels blocs esperem rebre per L/S:

- un Byte ACK (0x0F) que vol dir que s'han rebut tots correctament,
- un Byte ERR (0xAA) que vol dir que hi ha hagut error a la comunicació,
- res si el receptor s'ha desconnectat.

Immediatament després de rebre la resposta enviem el següent bloc o repetim l'anterior si hi ha error. Supposeu que estadísticament falla la transmissió d'1 bit de cada 22000 (2 punts).

Cada Byte a enviar necessitarà 11 bits de transmissió: start+8data+paritat+stop.

Tindrem 8 blocs de 256 Bytes de dades per transmetre, amb un total de  $8 \times 256 \times 11 = 22528$  bits per tant podem assumir que fallarà algun bit i hi haurà una retransmissió d'algun bloc de 256B.

Per tant les dades que circularan en total seran:

$(8 \text{ Blocs} \times 256B + 8 \text{ Respostes ACK}^* + 1 \text{ Bloc} \times 256B + 1 \text{ Resposta ERR}^*) \times 11 \text{bits} = 25443 \text{ bits.}$

25443 bits a 19200 bps= 1,325 segons.

\* A cada bloc hi ha un únic byte de resposta. Cada byte de LS té start, stop, etc.

6) En una prova volem configurar la línia sèrie del PIC18F45K22 amb Fosc=8MHz a 100Kbps. Serà possible obtenir aquesta velocitat en les configuracions assíncrones? (1 punt)

Cal fer la prova per totes aquestes configuracions (SYNC/BRG16/BRGH):

000:  $8\text{MHz}/64(n+1)=100000 \rightarrow n=0,25=0 \rightarrow \text{vel real } 125\text{Kbps}$ , massa error, no serveix.

001/010:  $8\text{MHz}/16(n+1)=100000 \rightarrow n=4 \rightarrow \text{vel real } 100\text{Kbps}$ , perfecte aquestes dues serveixen.

011:  $8\text{MHz}/4(n+1)=100000 \rightarrow n=19 \rightarrow \text{vel real } 100\text{Kbps}$ , perfecte, també serveix.

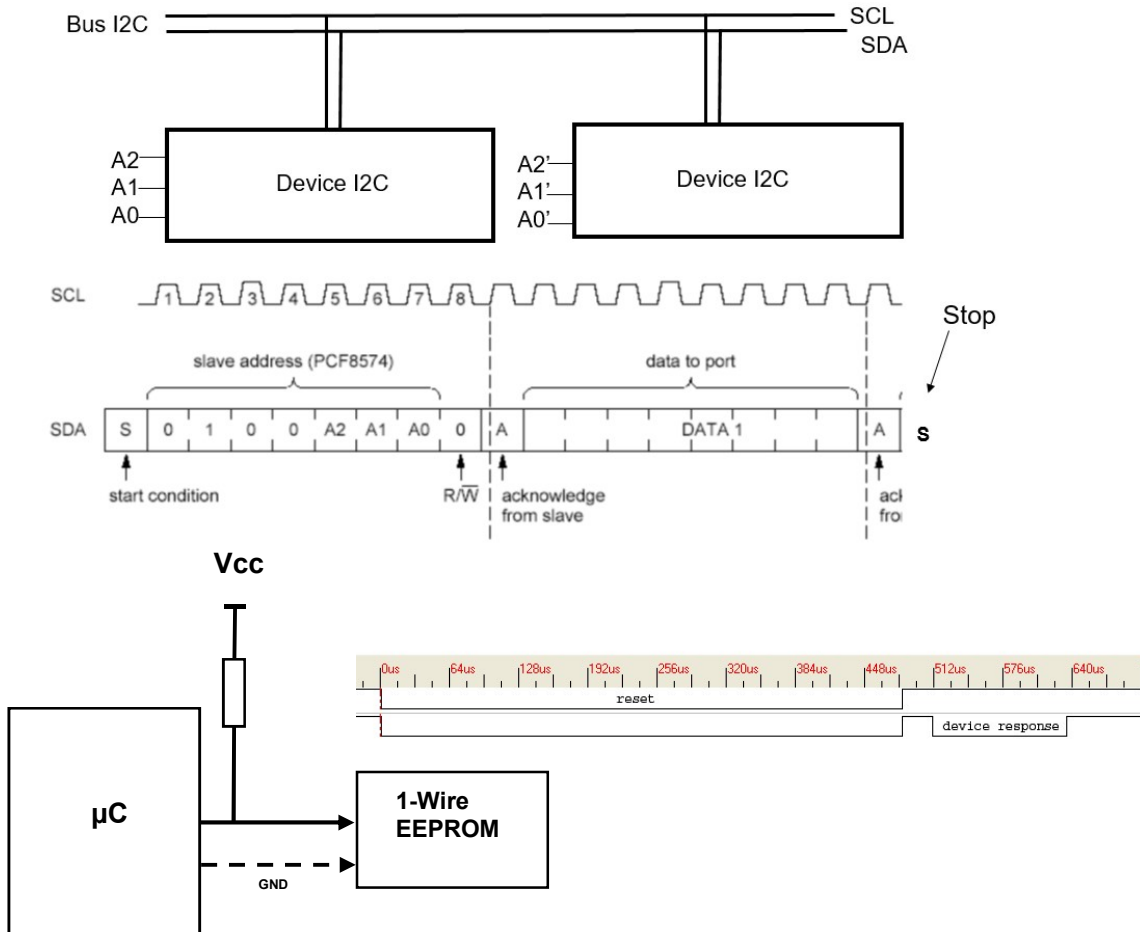
Així doncs veiem que hi ha 3 de 4 configuracions assíncrones on podem treballar a 100Kbps sense error a la generació de la freqüència.

Nom i Cognoms:

TABLE 20-1: BAUD RATE FORMULAS

Configuration Bits			BRG/EUSART Mode	Baud Rate Formula
SYNC	BRG16	BRGH		
0	0	0	8-bit/Asynchronous	$F_{osc}/[64 (n + 1)]$
0	0	1	8-bit/Asynchronous	$F_{osc}/[16 (n + 1)]$
0	1	0	16-bit/Asynchronous	
0	1	1	16-bit/Asynchronous	$F_{osc}/[4 (n + 1)]$
1	0	x	8-bit/Synchronous	
1	1	x	16-bit/Synchronous	

**Legend:** x = Don't care, n = value of SPBRGH:SPBRG register pair



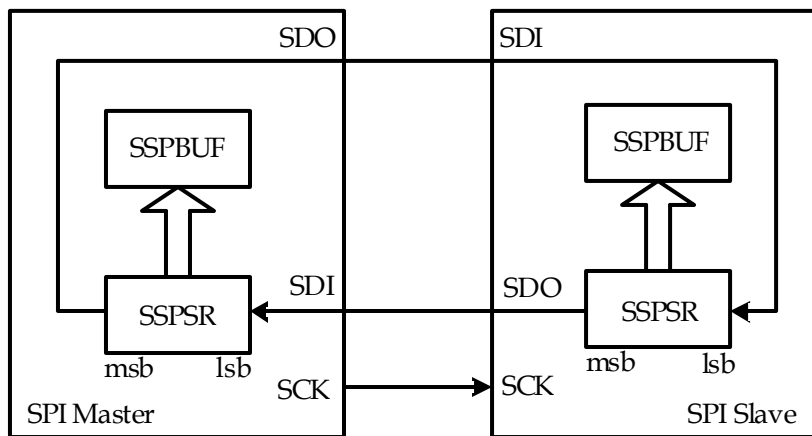


Figure 10.3 Connection between an SPI master and an SPI slave