

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

1) (1p.) Un enginyer de telecomunicacions presenta el següent codi per a fer una transmissió SPI per bit-banging:

```
char SPI_master (char byte)
{
    unsigned char bit;

    for (bit = 0; bit < 8; bit++)
    {
        if (byte & 0x80)
            SETMOSI();
        else
            CLRMOSI();
        byte <<= 1;

        espera(T/2);
        SETCLK(); // forcem flanc de rellotge

        byte |= READMISO();

        espera(T/2);
        CLRCLK();
    }
    return byte;
}

char SPI_slave (char byte)
{
    unsigned char bit;

    for (bit = 0; bit < 8; bit++)
    {
        While( !READCLK() ); // esperem flanc
        if (byte & 0x80)
            SETMISO();
        else
            CLRMISO();
        byte <<= 1;

        byte |= READMOSI();
    }
    return byte;
}
```

Raoneu si el codi és correcte, i en cas contrari, indiqueu l'error i proposeu una solució.

2) (4p.) Amb quins valors programaríeu els següents bits de control per a realitzar una conversió A/D amb el PIC18F4550 respectant les especificacions del fabricant? (Dades:  $F_{osc} = 48\text{MHz}$ ,  $T_{ADmin} = 0,8\mu\text{s}$ ,  $T_{ACQmin} = 2,54\mu\text{s}$ .)

a)

ADCS2:ADCS0 = .

b)

ACQT2:ACQT0 = .

b) Quin serà el temps total d'una conversió? (Temps d'adquisició + temps de conversió)

c) Si connectem 10 senyals analògics, de la mateixa freqüència, a 10 canals d'entrada del micro, amb una estratègia com aquesta:

```

.....
for (i=0 ; i<10; (i++)%10)
{
    .....
    processa(dada);
    dada = conversio_AD_canal(i);
    .....
}
.....

```

Quina pot ser la freqüència màxima de qualsevol dels senyals d'entrada per respectar el criteri de Nyquist? (negligiu el temps de procés de cada dada)

**ACQT2:ACQT0:** A/D Acquisition Time Select bits

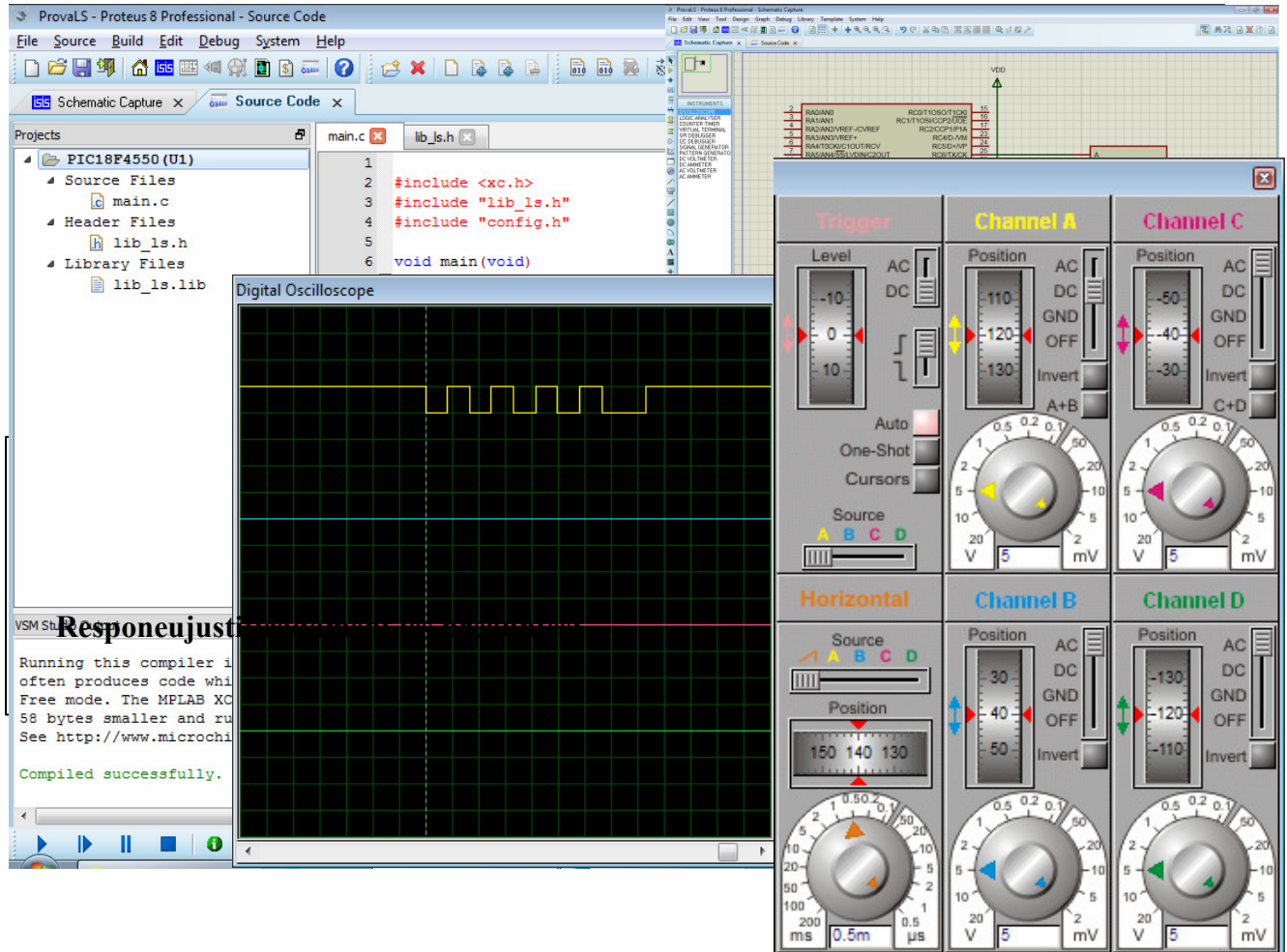
111 = 20 T<sub>AD</sub>  
 110 = 16 T<sub>AD</sub>  
 101 = 12 T<sub>AD</sub>  
 100 = 8 T<sub>AD</sub>  
 011 = 6 T<sub>AD</sub>  
 010 = 4 T<sub>AD</sub>  
 001 = 2 T<sub>AD</sub>  
 000 = 0 T<sub>AD</sub><sup>(1)</sup>

**ADCS2:ADCS0:** A/D Conversion Clock Select bits

111 = F<sub>RC</sub> (clock derived from A/D RC oscillator)<sup>(1)</sup>  
 110 = F<sub>osc</sub>/64  
 101 = F<sub>osc</sub>/16  
 100 = F<sub>osc</sub>/4  
 011 = F<sub>RC</sub> (clock derived from A/D RC oscillator)<sup>(1)</sup>  
 010 = F<sub>osc</sub>/32  
 001 = F<sub>osc</sub>/8  
 000 = F<sub>osc</sub>/2

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

3) (2,5p.) Trobem una llibreria precompilada de línia sèrie pel PIC18F4550 (ls\_lib.lib i ls\_lib.h). Decidim provar-la amb el Proteus fent un programa que periòdicament envia el caràcter 0x55 (es sol fer servir ja que alterna 1 i 0 en codificació binària). Observeu les captures de pantalla amb el resultat de la simulació:

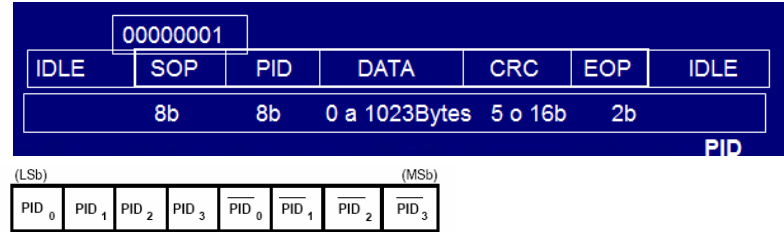
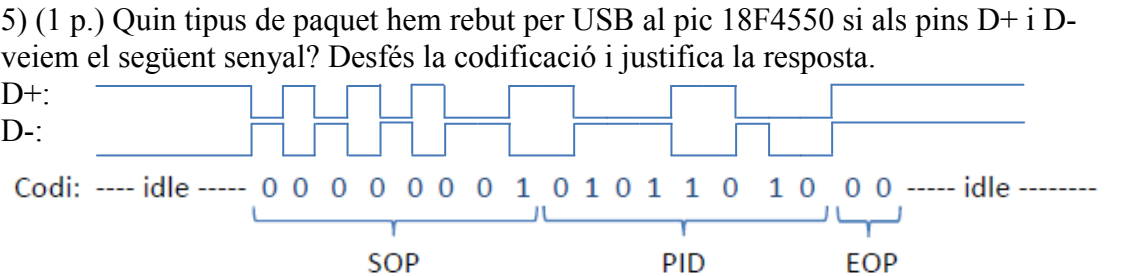


Responen justificadament les preguntes:

- A quina de les velocitats de transmissió està configurat el PIC?
- S'està fent servir bit de paritat?
- Si en comptes d'enviar un byte 0x55 enviéssim un 0x33, quina forma tindria el senyal vist a l'oscil·loscopi?

Sortiria: ...11(idle) 0 11 00 11 00 0(P) 1 1111... (idle)

4) (1,5p.) Volem fer un programa que envii per línia sèrie totes les mostres obtingudes pel convertor analògic digital i amb tots els seus bits.  
 Si el convertor A/D ens dona una mostra de 10 bits cada 250 microsegons, quina serà la velocitat mínima a que cal configurar la línia sèrie? Negligiu els temps d'atenció a interrupcions, manegament de buffers, etc.



PID Type	PID Name	PID[3:0]*	Description
Token	OUT	0001B	Address + endpoint number in host-to-function transaction
	IN	1001B	Address + endpoint number in function-to-host transaction
	SOF	0101B	Start-of-Frame marker and frame number
	SETUP	1101B	Address + endpoint number in host-to-function transaction for SETUP to a control pipe
Data	DATA0	0011B	Data packet PID even
	DATA1	1011B	Data packet PID odd
Handshake	ACK	0010B	Receiver accepts error-free data packet
	NAK	1010B	Rx device cannot accept data or Tx device cannot send data
	STALL	1110B	Endpoint is halted or a control pipe request is not supported.
Special	PRE	1100B	Host-issued preamble. Enables downstream bus traffic to low-speed devices.