

1. La respuesta corta es que la afirmación es cierta. Teniendo en cuenta que la frecuencia de una señal periódica determina el número de repeticiones en un segundo (s^{-1} , Hz) de la misma, se puede, mediante el contador, medir durante un tiempo determinado el número de repeticiones de la señal.

Por ejemplo: sea $x(t)$ una señal periódica de frecuencia $f = 5$ Hz. Si se cuenta el número de veces que se repite la señal en un segundo, obtenemos que el intervalo entre pulsación y pulsación es de $1/5$ s, lo que se conoce como el período (T) de la señal, que cumple que $T = 1/f$ y, efectivamente, se mide en segundos.

2. El programa se mantendrá en el bucle mientras el convertidor AD esté realizando una conversión. Solo en el momento en que termine, el bit $PIR1.ADIF$ tendrá el valor 1, y forzará la salida del bucle.

3.

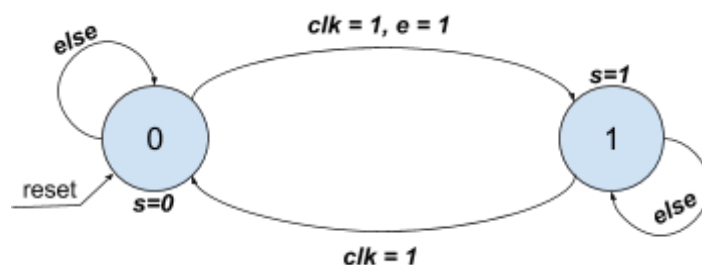
i. -

```
ii. void interrupt(){
    if(INTCON.TMR0IF == 1){
        PORTC.B0 = (PORTA.B0 == 1)? 0 : PORTC.B0;
        PORTC.B0 = (PORTA.B1 == 0)? 1 : PORTC.B0;
    }
    TMR0H = (15536 >> 8);
    TMR0L = 15536;
    INTCON.TMR0IF = 0;
}

void main() {
    ADCON1 = 0x07;
    TRISA.B0 = 1;
    TRISA.B1 = 1;
    TRISC.B0 = 0;
    TOCON = 0x82;
    RCON.IPEN = 0;
    INTCON = 0x2C;
    TMR0H = (15536 >> 8);
    TMR0L = 15536;
    INTCON.GIE = 1;

    PORTC.B0 = 0;
    while(1);
}
```

- iii. Diagrama de flujo de estados:



iv.

```

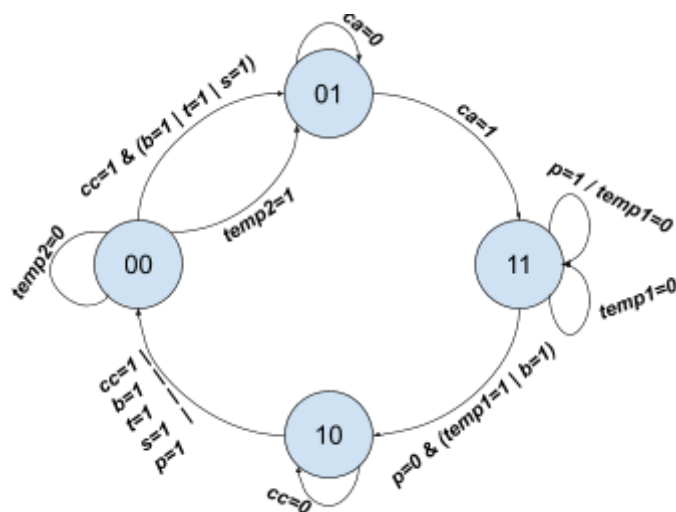
char estado = 0;
void interrupt(){
    if(INTCON.TMR0IF == 1){
        if(estado){
            estado = 0;
        }else{
            estado = PORTA.B0;
        }
        PORTC.B0 = estado;
    }
    TMR0H = (7768 >> 8);
    TMR0L = 7768;
    INTCON.TMR0IF = 0;
}

void main() {
    ADCON1 = 0x07;
    TRISA.B0 = 1;
    TRISC.B0 = 0;
    T0CON = 0x82;
    RCON.IPEN = 0;
    INTCON = 0x2C;
    TMR0H = (7768 >> 8);
    TMR0L = 7768;
    INTCON.GIE = 1;

    PORTC.B0 = 0;
    while(1);
}

```

v.



4.

i. -

ii.

- a. Un μ C se escoge en función de:
 1. Ancho de palabra \rightarrow 4,8,16,32 bits
 2. Memoria secundaria \rightarrow EPROM, EEPROM, OTP, Flash, ROM; la memoria RAM suele ser SRAM, y no hay muchas variaciones entre μ Cs
 3. Convertidores A/D \rightarrow 4,8,16 canales; 8,12,16 bits
 4. Convertidores D/A
 5. Temporizador (uno o varios) \rightarrow 8,16,32 bits ; contador
 6. Temporizador *watchdog*
 7. Sistema de interrupciones (muy común)
 8. Comunicación con periféricos \rightarrow computadores, displays, impresoras... mediante dispositivos de comunicación serie síncrona y asíncrona
 9. Distintos puertos E/S
 10. Unidades de generación de señales especiales \rightarrow trenes de pulsos, PWM, etc.
 11. Distintas herramientas de desarrollo \rightarrow ensambladores, enlazadores, compiladores, simuladores...
 12. Consumo de energía
- b. Los DSP destacan por su aplicabilidad frente a aplicaciones de procesamiento numérico intensivo a alta velocidad, y se relacionan sobre todo con algoritmos de procesamiento digital de señal como filtrado digital, análisis espectral y otros. Es práctico en sistemas que procesan señales analógicas en tiempo real.
Un μ C, en cambio, está destinado a tareas (sencillas y complejas) que se adaptan a un hardware que se quiere controlar (sistemas empujados).
- c. Procesamiento de señales de audio, reconocimiento de voz, comunicaciones, multimedia, control industrial...
- d. Los criterios de selección de un DSP mostrados en el pdf parecen ser válidos.
- e. La multiplicación, mediante el multiplicador MAC.
- f. Permite mayor precisión con números flotantes. Es más preciso tratar con un tamaño de palabra mayor, pues permite considerar más decimales.
- g. De un mínimo de 24, preferiblemente 32