

PIC 16F87X



Juan González

Escuela Politécnica Superior
Universidad Autónoma de Madrid

Andrés Prieto-Moreno

Flir Networked Systems

Ricardo Gómez

Flir Networked Systems

PIC 16F87X

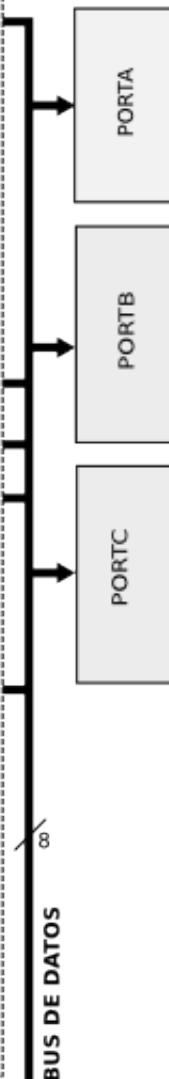
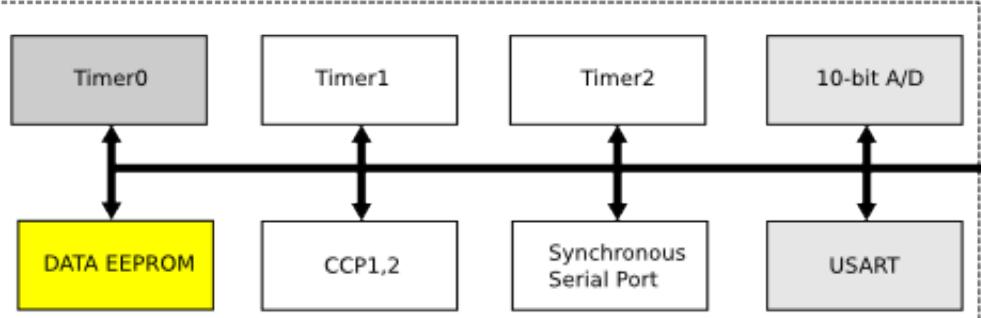


MÓDULO 7:

Memoria EEPROM

NUCLEO PIC16F876

PERIFERICOS

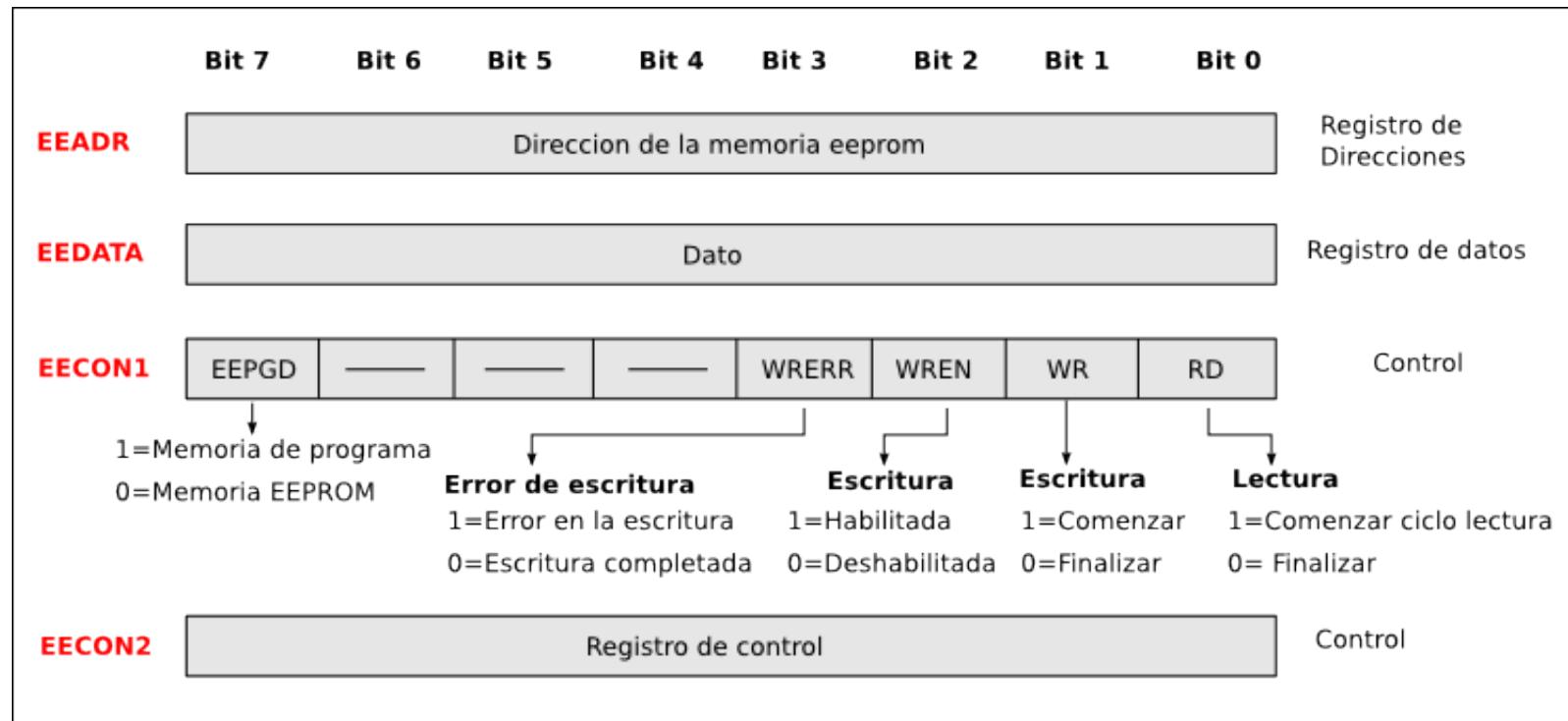


**Memoria
EEPROM**

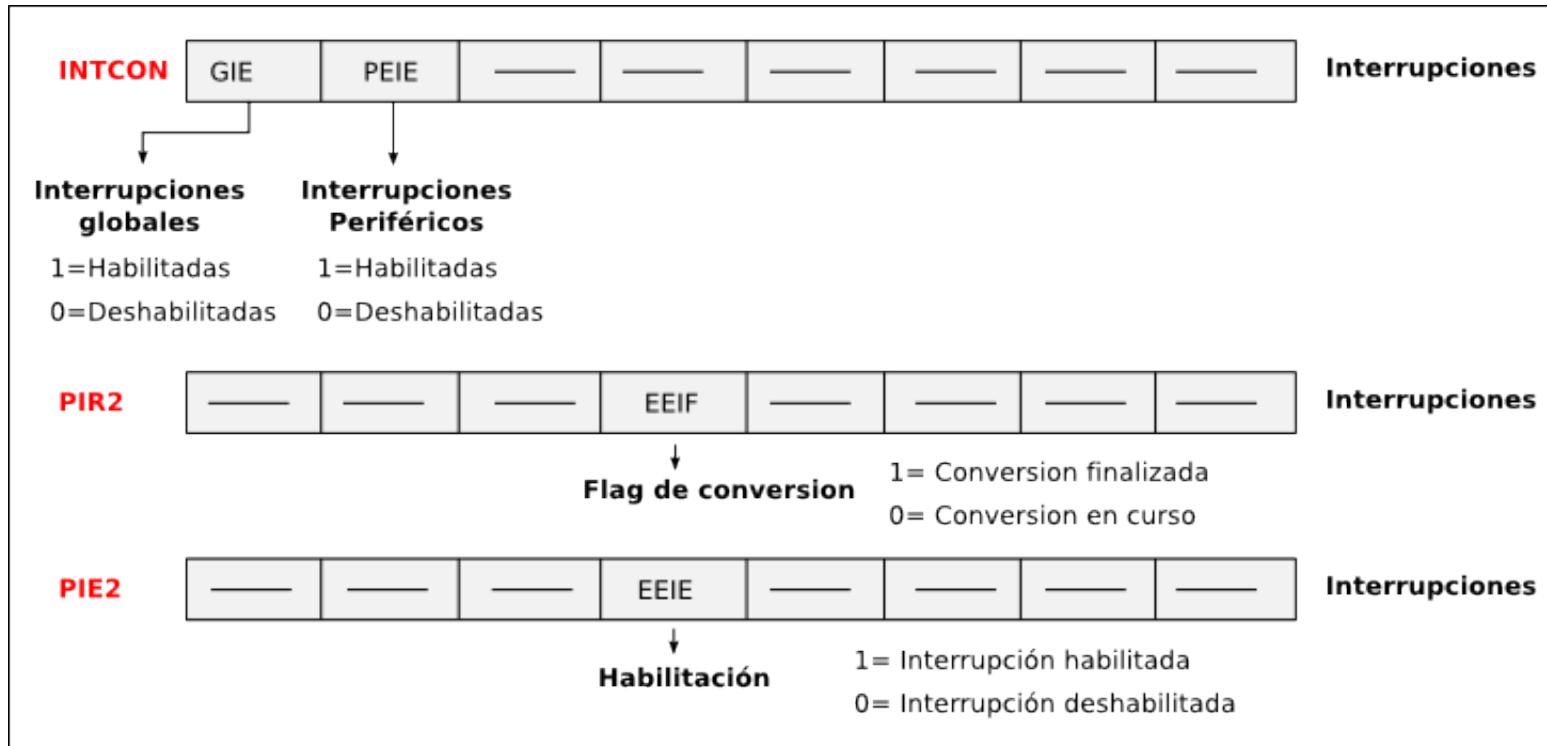
Introducción

- Memoria EEPROM de **256 bytes**
- **No** está mapeada directamente en el espacio de memoria
- El acceso es indirecto, a través de unos registros especiales
- Interrupción para indicar el final de una escritura

Registros (I)



Registros (II)



Escritura en la eeprom

Para escribir en la eeprom hay que realizar las siguientes operaciones

- Establecer la **dirección** donde escribir (registro **EEADR**);
- Establecer el **dato** (registro **EEDATA**)
- Configurar acceso a la EEPROM (Bit **EEPGD** a cero)
- Habilitar la escritura (activar **WREN**)
- Escribir el valor **0x55** en registro **EECON2**
- Escribir el valor **0xAA** en el registro **EECON2**
- **Comenzar la escritura** activando el bit **WR**
- Cuando la escritura finaliza se activa el flag **EIF**

Escritura en la eeprom: Ejemplo

eeprom-write-byte.c

```
#include <pic16f876a.h>
void eeprom_write(unsigned char dir, unsigned char dato)
{
    EEADR=dir;           → Establecer dirección
    EEDATA=dato;         → Establecer dato
    EEPROM=0;            → Seleccionar la memoria EEPROM
    WREN=1;              → Habilitar la escritura
    EECON2=0x55;          → Escribir los valores en EECON2
    EECON2=0xAA;
    WR=1;                → Comenzar la escritura
    while (EEIF==0);      → Esperar a que la escritura
                           termine
    EEIF=0;               → Limpiar flag de la eeprom
}
void main(void)
{
    TRISB1=0;             → Configurar led
    eeprom_write(0, 0x55);   → Escribir el valor 0x55 en la
                           dirección 0 de la EEPROM
    RB1=1;
    while(1);             → Encender led para indicar
                           fin de la escritura
}
```

Función para escribir un dato en la eeprom en la dirección indicada

Lectura de la eeprom

Para leer de la eeprom hay que realizar las siguientes operaciones:

- Establecer la **dirección** de donde leer (registro **EEADR**);
- Configurar acceso a la EEPROM (Bit **EEPGD** a cero)
- **Comenzar la lectura** activando el bit **RD**
- En el siguiente ciclo ya está disponible el dato en el registro **EEDATA**

Lectura de la eeprom: Ejemplo

eeprom-read-byte.c

```
#include <pic16f876a.h>
unsigned char eeprom_read(unsigned char dir)
{
    EEADR=dir;           → Establecer dirección
    EEPGD=0;             → Establecer dato
    RD=1;                → Seleccionar la memoria
    return EEDATA;        → EEPROM
}

void main(void)
{
    TRISB=0;             → Configurar puerto B para
    PORTB=eeprom_read(0x00); → salida
    while(1);
}
```

Función para leer un dato de la dirección de la EEPROM indicada

Leer el dato que está en la dirección 0 de la EEPROM y sacarlo por los leds

Ejemplo: Escribir una tira de bytes

eeprom-write.c

```
#include <pic16f876a.h>
unsigned char tabla[]={ 'H', 'O', 'L', 'A', '!' };
unsigned char size =
    sizeof(tabla)/sizeof(unsigned char);
void main(void)
{
    unsigned int i;

    TRISB1=0;
    for (i=0; i<size; i++) {
        eeprom_write(i, tabla[i]);
    }
    for (i=size; i<255; i++) {
        eeprom_write(i, '.');
    }
    RB1=1;
    while(1);
}
```

Tabla con los bytes a escribir en la eeprom

Tamaño de la tabla

Recorrer la tabla escribiendo los bytes en la eeprom, comenzando por la dirección 0

Usamos la función empleada en el ejemplo de escritura

Rellenar el resto de la memoria con el carácter .

Encender el led para indicar que la operación ha terminado

Ejemplo: Volcado de la eeprom

eeprom-dump-sci.c

```
#include <pic16f876a.h>
#include "sci.h"
void main(void)
{
    unsigned char i;
    unsigned char valor;
    sci_conf();
    TRISB=0;

    for (i=0; i<255; i++) {
        valor=eeprom_read(i);
        PORTB=valor;
        sci_write(valor);
    }
    while(1);
}
```

Recorrer la memoria eeprom
enviando todos los bytes por el
puerto serie

Configurar puerto serie

Configurar puerto B para salida

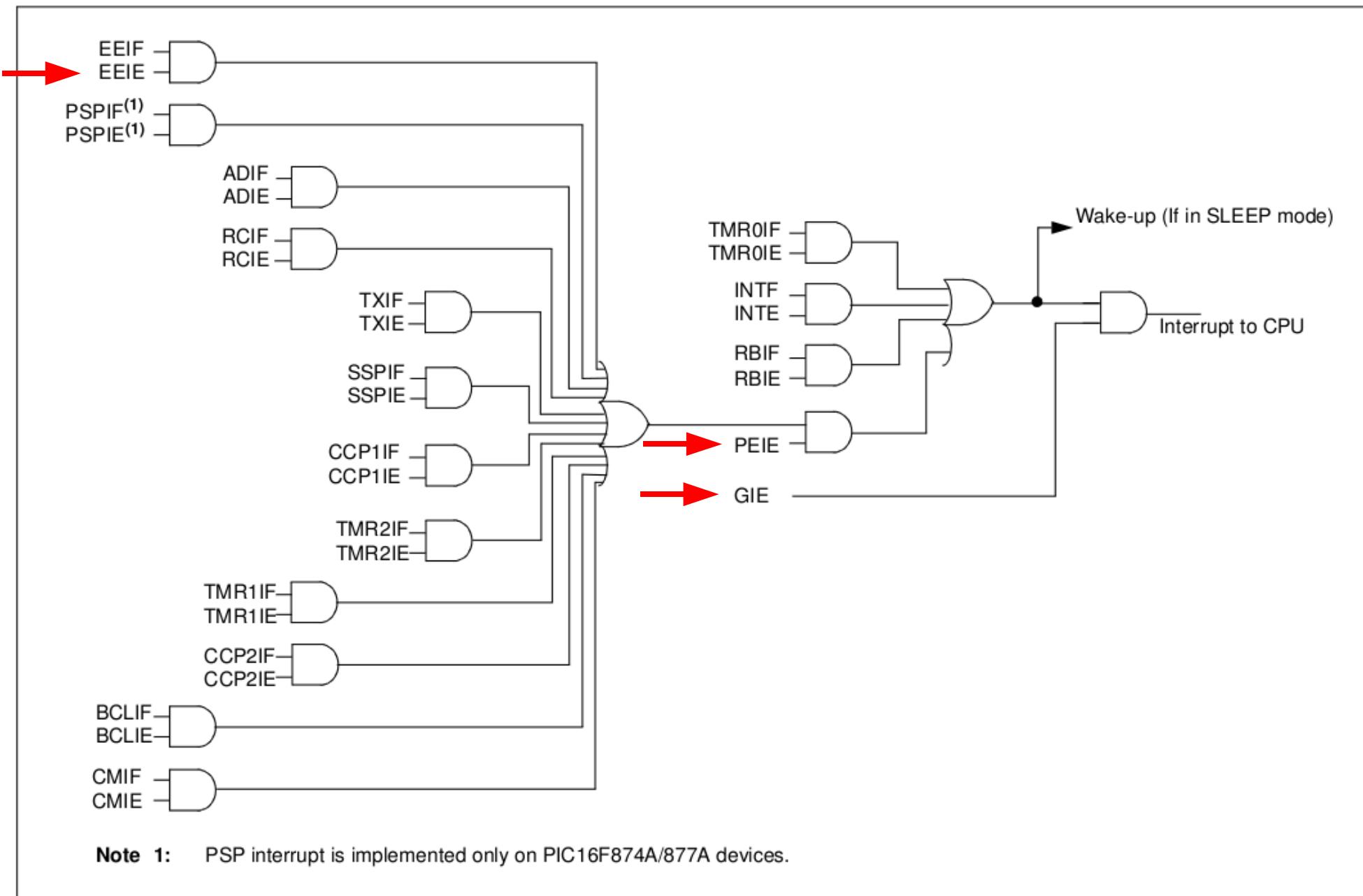
Recorrer toda la eeprom

Leer la posición i

Sacarla por los leds

Enviarla por el puerto serie

Interrupciones



eeprom-write-int.c (parte I)

Interrupciones: Ejemplo

```
#include <pic16f876a.h>
unsigned char tabla[]={'P','O','D','E','M','O',
'S','!','!'};
unsigned char size =
 sizeof(tabla)/sizeof(unsigned char);
unsigned char i=0;
```

```
void isr() interrupt 0
```

```
{
```

```
EEIF=0;
```

Limpiar flag de interrupcion

```
i++;
```

Incrementar indice del dato a escribir

```
if (i==size) {
```

```
RB1=1;
```

```
return;
```

```
}
```

```
EEADR=i;
```

Establecer direccion

```
EEDATA=tabla[i];
```

Establecer el dato a grabar en la eeprom

```
EEPGD=0;
```

```
WREN=1;
```

```
EECON2=0x55;
```

```
EECON2=0xAA;
```

```
WR=1;
```

Tabla con los bytes a escribir en la EEPROM

Tamaño de la tabla

Si es el último dato encender el led y terminar

Realizar la grabacion. Cuando finalice se volverá a llamar a la rutina de atención a la interrupción

```
void main(void)
{
    TRISB1=0;
    EEIF=0;
    EEIE=1;
    PEIE=1;
    GIE=1;

    EEADDR=0;
    EEDATA=tabla[0];
    EEPGD=0;
    WREN=1;
    EECON2=0x55;
    EECON2=0xAA;
    WR=1;
    while(1);
}
```

Configurar led para salida

Limpiar flag de la eeprom

Activar las interrupciones

Seguir el protocolo para grabar el primer dato de la table en la dirección 0. El resto de bytes se graban mediante interrupciones

Este ejemplo escribe una cadena en la eeprom, mediante interrupciones

Ejercicio:

- Escribir en el LCD la cadena que esté grabada a partir de la dirección 0 de la EEPROM