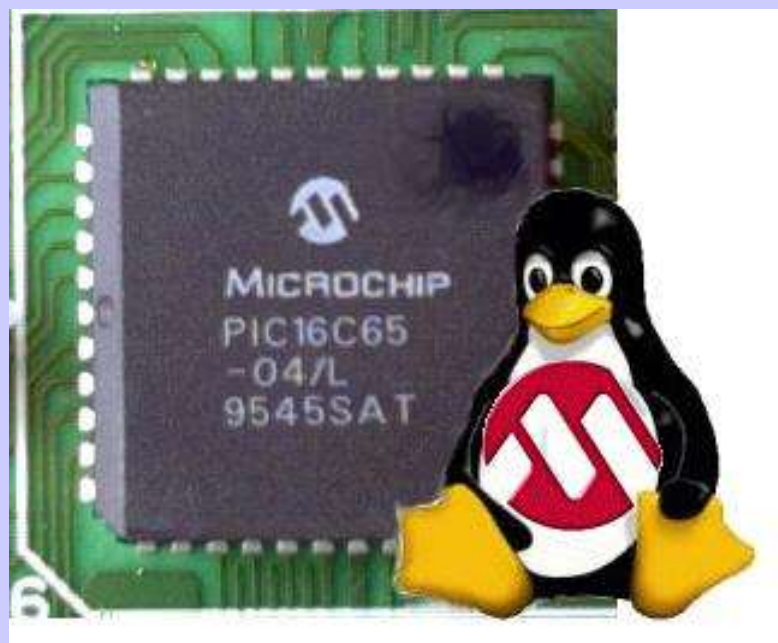
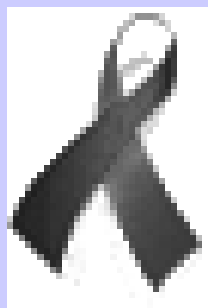


Herramientas hardware y software para el desarrollo de aplicaciones con Microcontroladores PIC bajo plataformas GNU/Linux



Juan González Gómez
Escuela Politécnica Superior
Universidad Autónoma de Madrid



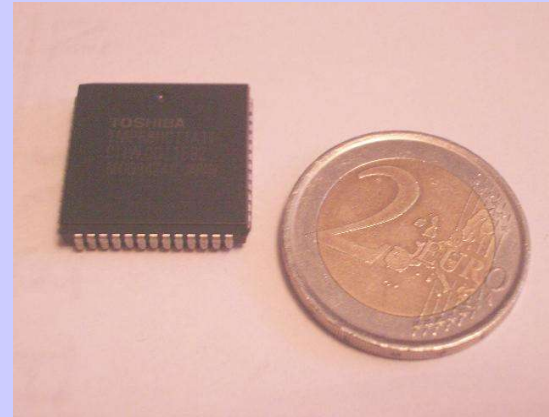
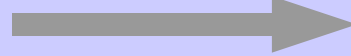
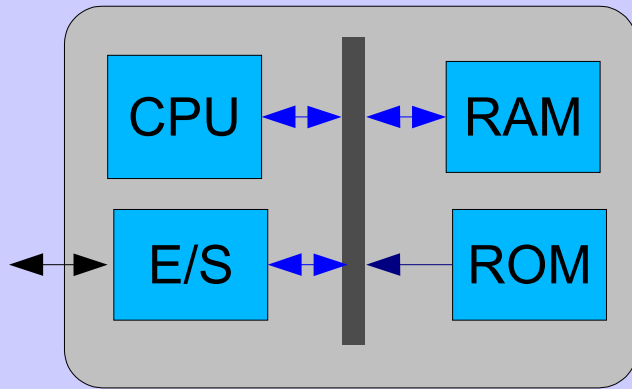
Andrés Prieto-Moreno Torres
Ifara Tecnologías
Profesor asociado UPSAM

ÍNDICE

- **INTRODUCCIÓN**
- **¿Por qué GNU/Linux?**
- **Herramientas GNU/Linux para PICs**
- **Grabación**
- **Conclusiones**
- **Demostración**

Introducción (I)

- Microcontroladores: "Un mini-ordenador en un circuito integrado"



- **Ejecutan un programa**
- **Son reprogramables**
- **Aplicaciones:** Controladores, Robótica, Sistemas empotrados...
- **Muchos fabricantes diferentes:** Motorola, Intel, Atmel, Microchip...

Introducción (II): Microcontroladores PIC

- **Fabricante:** Microchip
- **Muchas familias** de PICs
- **Características familia 16F8X, 16F87X:**
 - Micros de 8 bits
 - Arquitectura RISC
 - Frecuencia máxima: 4MHz – 20MHz
 - Temporizadores
 - Puertos E/S
 - Otros periféricos (dependiendo del modelo concreto):
 - Conversores A/D
 - Comunicaciones serie síncronas/asíncronas
 - PWM

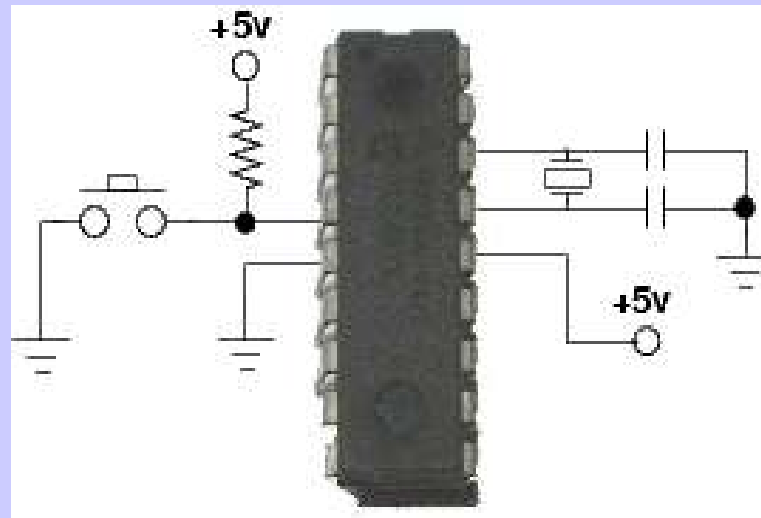


Nos hemos centrado en las familias 16F8X y 16F87X

Introducción (III): Microcontroladores PIC

■ VENTAJAS:

Es muy sencillo construirte tu propio sistema microcontrolador, usando PICs

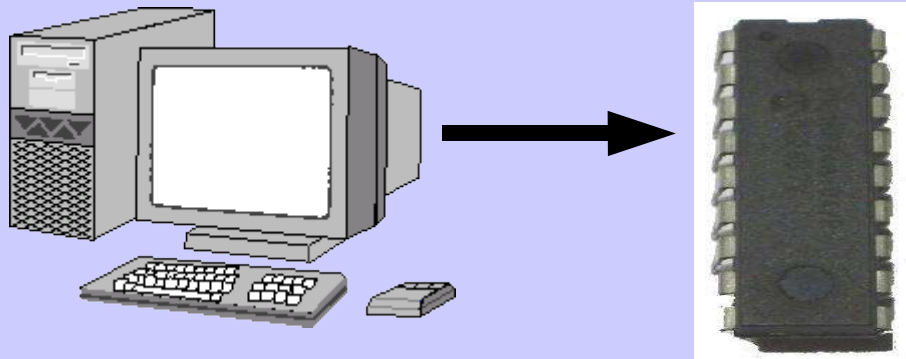


...y además es barato

Introducción (IV): Microcontroladores PIC

■ ¿Cómo programamos los PIC?

- Necesario un ordenador anfitrión donde escribir los programas, compilarlos, simularlos y grabarlos en el pic



■ ¿Microchip ofrece "gratis" la herramienta MPLAB

- Es "gratis", pero no es Libre. No disponemos de las fuentes
- Sólo está disponible para plataformas Windows

Si quiero hacer una aplicación con PICs, tendré que disponer de un ordenador PC, con Sistema Operativo Windows (con su licencia). Microchip, a priori, decide por nosotros

ÍNDICE

- **INTRODUCCIÓN**
- **¿Por qué GNU/Linux?**
- **Herramientas GNU/Linux para PICs**
- **Grabación**
- **Conclusiones**
- **Demostración**

¿Por qué GNU/Linux?

- Porque es el sistema operativo que uso a diario
- Independencia del fabricante
 - Seremos "inmunes" a las decisiones que tome el fabricante por nosotros
- Multiplataforma
 - El diseñador no está atado a una única arquitectura (PC Intel)
 - Ej. La distribución Debian está portada a 11 arquitecturas diferentes.
- Creación Live CD
 - Impartición de talleres, evaluación de nuevo software, etc.
- Prácticas avanzadas

En general, por la **libertad de elección**. Si yo soy el diseñador, yo decido qué herramientas usar y qué plataformas

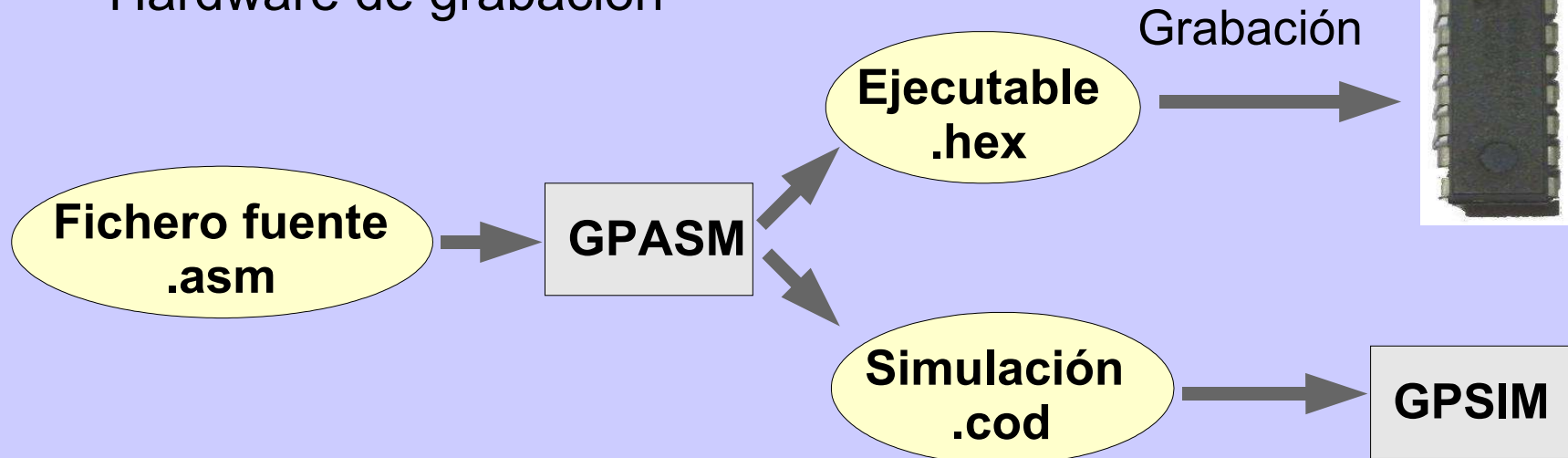
ÍNDICE

- **INTRODUCCIÓN**
- **¿Por qué GNU/Linux?**
- **Herramientas GNU/Linux para PICs**
- **Grabación**
- **Conclusiones**
- **Demostración**

Herramientas (I)

■ Para trabajar con los pics necesitamos:

- Editor de texto ✓
- Ensamblador cruzado → GPASM ✓
- Simulador → GPSIM ✓
- Grabador ✗
 - Software grabación
 - Hardware de grabación



Herramientas (II): gpasm

- Ensamblador 100 % compatible con el MPASM de Microchip
- Forma parte del paquete GPUTILS, que incluye otras herramientas: Enlazador, desensamblador...

<http://gputils.sourceforge.net/>

■ Utilización: Línea de comandos

```
$ gpasm prueba.asm
```

prueba.hex

prueba.cod

Posibilidad de utilizar herramientas como **make** o entornos de desarrollo integrados (IDE) como **anjuta**

Herramientas (III): gpsim

- Simulador
- No están soportados todos los PICs

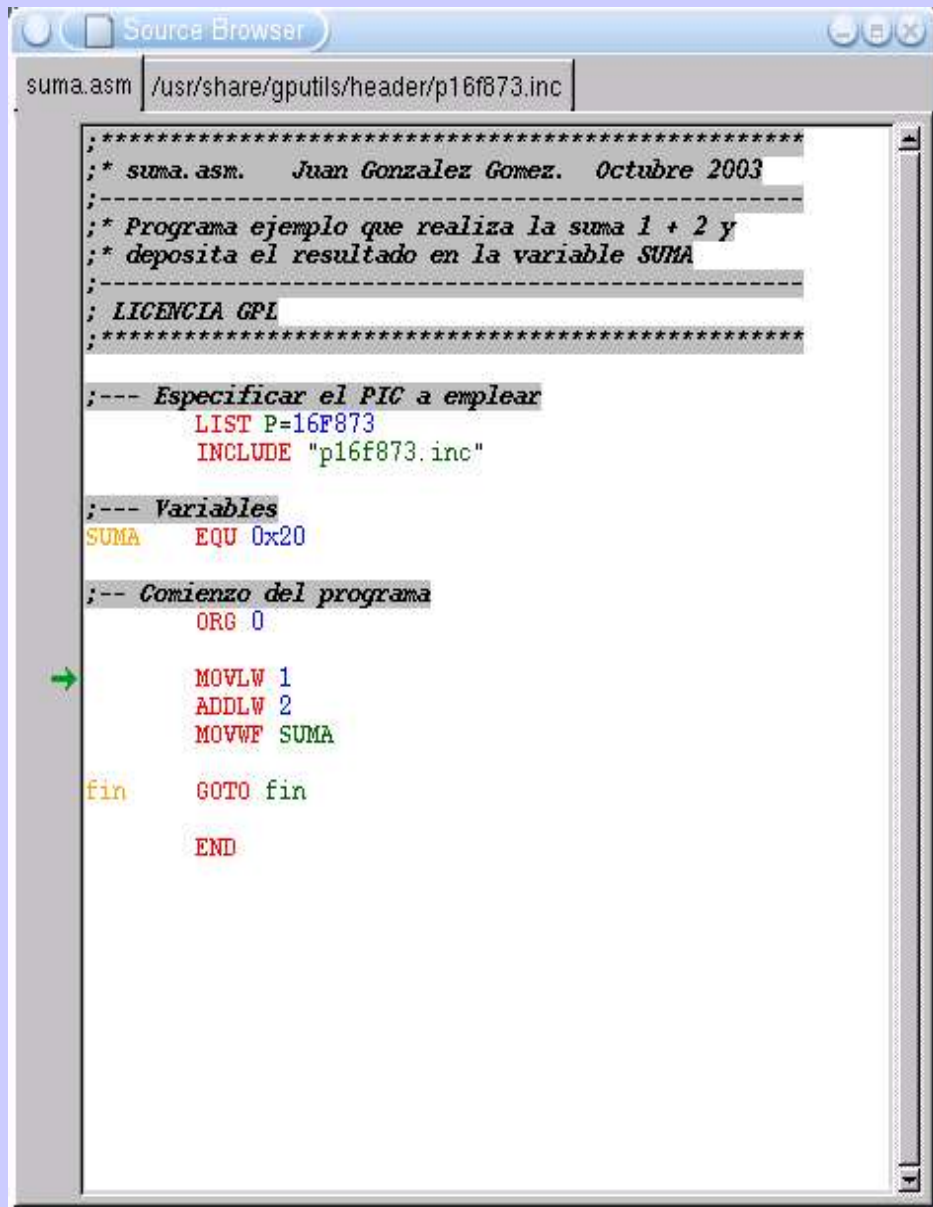
<http://www.dattalo.com/gnupic/gpsim.html>

- Utilización:

```
$ gpsim -s prueba.cod
```



Herramientas (VI): gpsim



```
suma.asm /usr/share/gputils/header/p16f873.inc
;*****
;* suma.asm. Juan Gonzalez Gomez. Octubre 2003
;-----
;* Programa ejemplo que realiza la suma 1 + 2 y
;* deposita el resultado en la variable SUMA
;-----
; LICENCIA GPL
;*****

;--- Especificar el PIC a emplear
LIST P=16F873
INCLUDE "p16f873.inc"

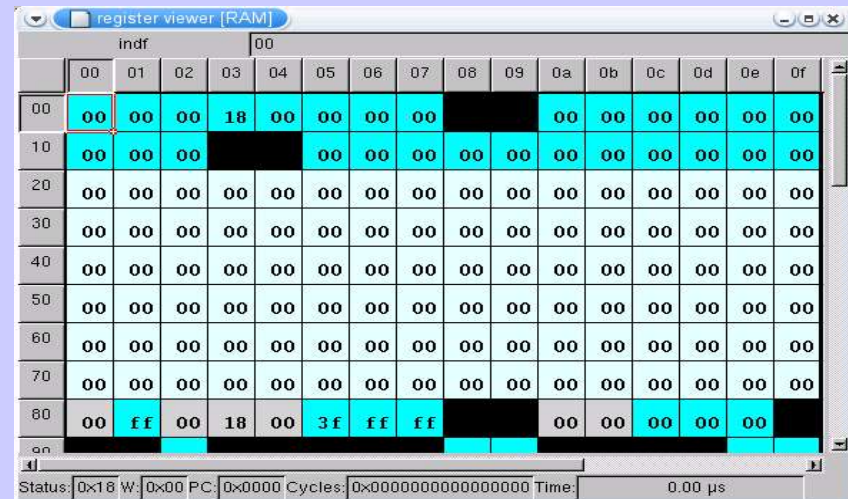
;--- Variables
SUMA EQU 0x20

;-- Comienzo del programa
ORG 0

MOV LW 1
ADD LW 2
MOV WF SUMA

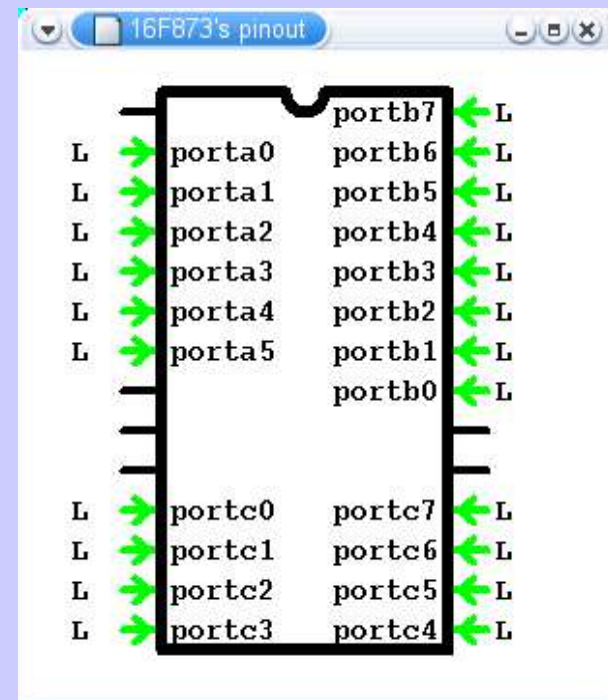
fin GOTO fin

END
```



addr	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	00	00	00	18	00	00	00	00		00	00	00	00	00	00	00
10	00	00	00		00	00	00	00	00	00	00	00	00	00	00	00
20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80	00	ff	00	18	00	3f	ff	ff		00	00	00	00	00	00	00
90																

Status: 0x16 W: 0x00 PC: 0x0000 Cycles: 0x0000000000000000 Time: 0.00 µs

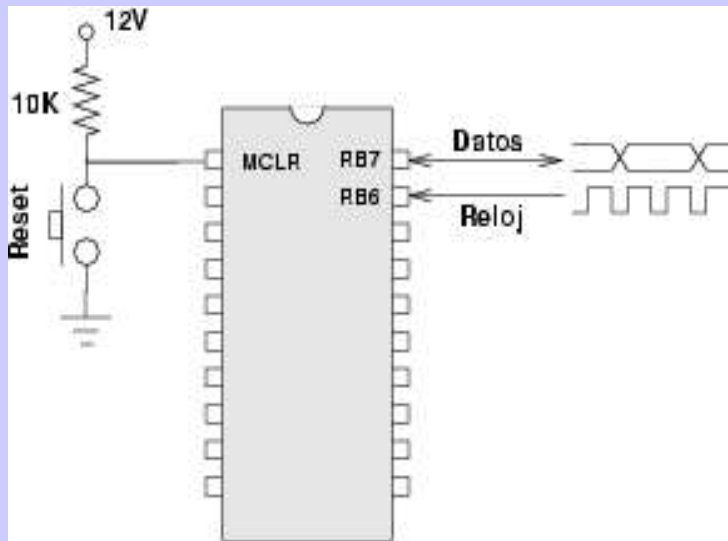


ÍNDICE

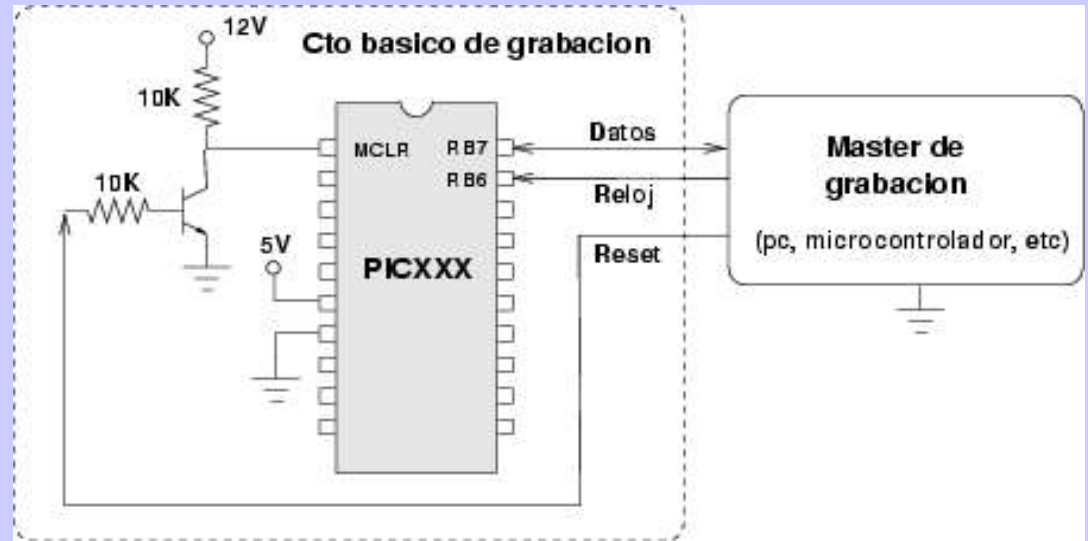
- **INTRODUCCIÓN**
- **¿Por qué GNU/Linux?**
- **Herramientas GNU/Linux para PICs**
- **Grabación**
- **Conclusiones**
- **Demostración**

Grabación (I)

- Para grabador los PICs se necesita:
 - Tensión de 12V para entrar en modo "monitor"
 - Una señal para los datos
 - Una señal para el reloj (protocolo serie síncrono)



Circuito 1 para grabación

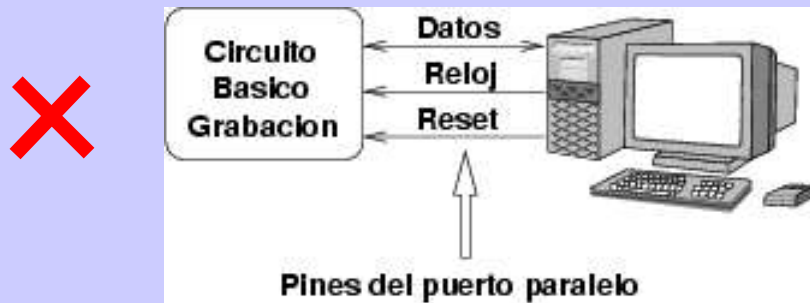


Circuito 2: Incluye señal de reset

Grabación (II)

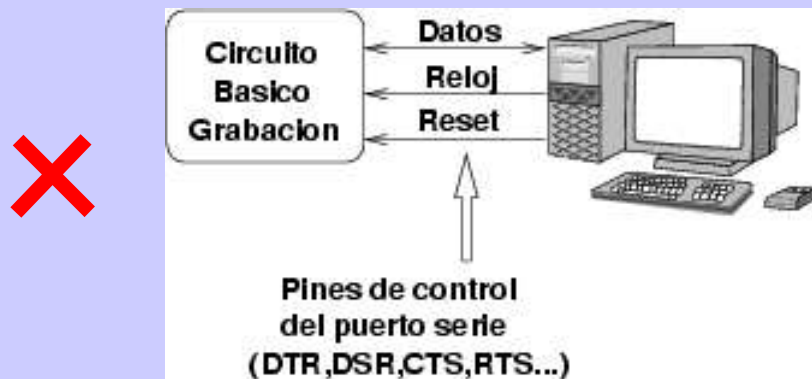
■ Existen distintos tipos de grabadores:

1.- Grabadores por puerto paralelo del PC:



- **Software muy dependiente del PC:** si uso otro PC distinto puede que no me funcione
- **Los sistemas operativos no ofrecen una API** para manejo del puerto paralelo a bajo nivel

2.- Grabadores por puerto serie del PC:



- **No usan el estándar RS-232:** utilizan los pines de control como señales digitales de E/S. Software dependiente de la máquina y poco portable
- **Problemas de alimentación**

Grabación (III)

3.- Grabador con microcontrolador

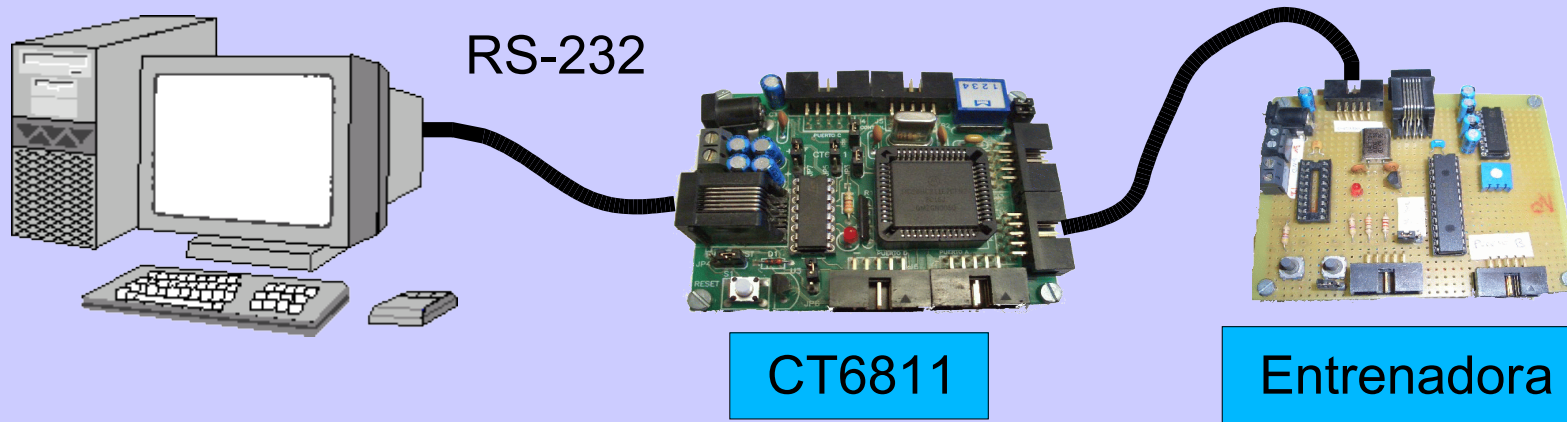


- El protocolo de grabación se implementa en un microcontrolador
- El PC envía los datos a grabar, usando una comunicación estándar (USB, RS-232)
- Solución empleada por Microchip

El grabador que hemos desarrollado se basa en esta arquitectura. La comunicación con el PC se hace por RS-232 estándar, a 9600 Baudios

Grabación (IV)

- Grabador implementado en una tarjeta CT6811

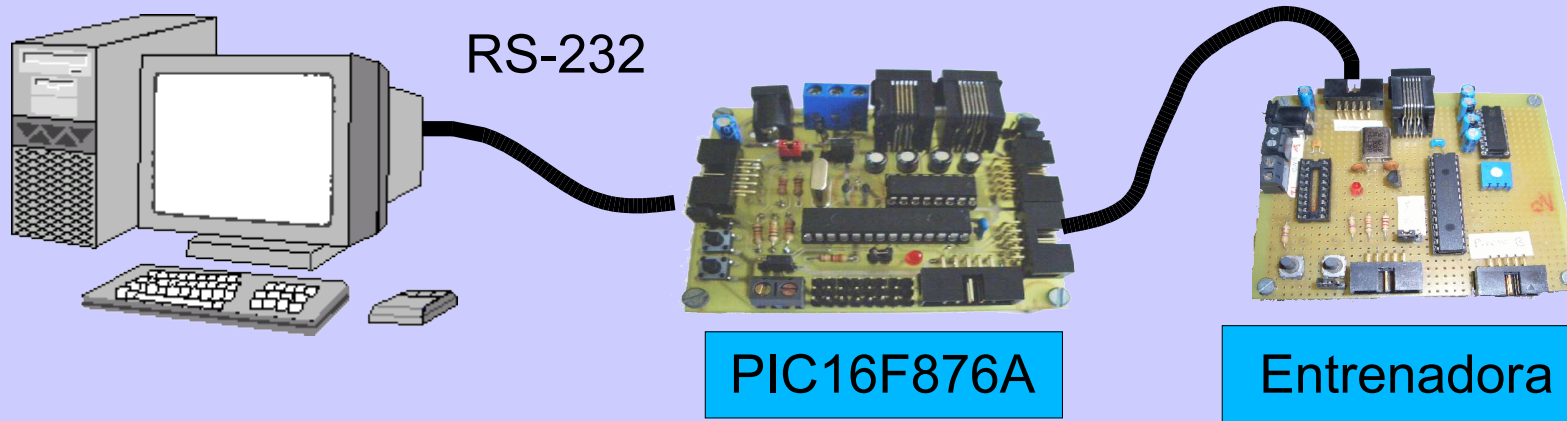


- Arquitectura software: Cliente-servidor



Grabación (V)

- Grabador implementado con un PIC16F876A



- El software del PC es el mismo, porque los servicios que ofrece el servidor de grabación son los mismos
- Si sabes construirte una tarjeta entrenadora para el PIC, también te sirve para tener tu propio grabador
- Software del PC: programa skypic-down (linux)
- Fácilmente portable a cualquier otra plataforma (Windows)

Grabación (VI)

■ Ejemplo de grabación

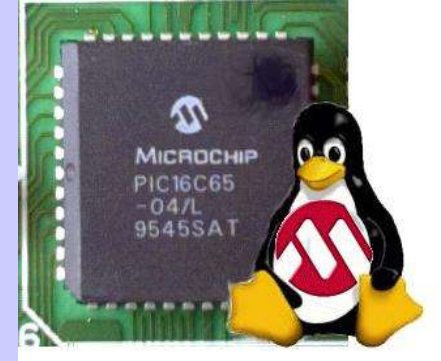
```
$ gpasm prueba.asm
```

ENSAMBLADO



```
$ skypic-down prueba.asm
```

GRABACIÓN



- Con sólo dos comandos tenemos nuestro PIC grabado
- Posibilidad de realizar el proceso desde un IDE

¡Hemos grabado un PIC desde GNU/Linux!

ÍNDICE

- **INTRODUCCIÓN**
- **¿Por qué GNU/Linux?**
- **Herramientas GNU/Linux para PICs**
- **Grabación**
- **Conclusiones**
- **Demostración**

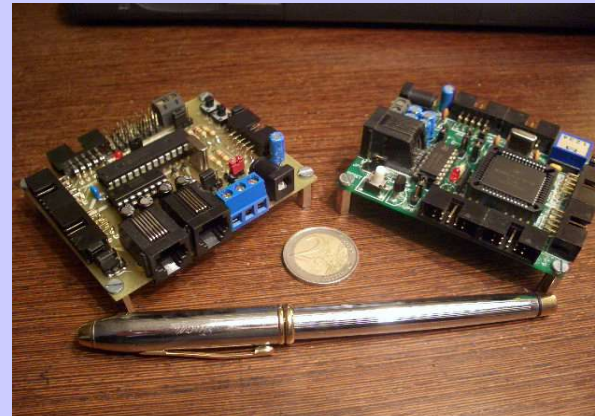
CONCLUSIONES

- Los microcontroladores PIC se están empleando mucho
 - Baratos
 - Fácil construcción de tarjetas
- Se han evaluado las herramientas disponibles para GNU/Linux
 - GPASM, GPSIM
- Diseñado un grabador, implementado con la tarjeta CT6811 o con un PIC16F876A
- Software de grabación para GNU/Linux
- Sistema probado en el laboratorio de Arquitectura de computadores

Es totalmente viable plantearse la docencia práctica sobre los microcontroladores PIC utilizando entornos GNU/Linux.

TRABAJO FUTURO

- **Estamos trabajando en una entrenadora sencilla, barata y libre**
 - **Tarjeta SKYPIC.**
 - **Sirve como entrenadora, o como grabadora**
 - **Plataformas Linux/Windows. El diseñador decide**



- **Software skypic-down:**
 - **Interfaz gráfica para GTK 2.0**
 - **Autodetección del PIC**

ÍNDICE

- **INTRODUCCIÓN**
- **¿Por qué GNU/Linux?**
- **Herramientas GNU/Linux para PICs**
- **Grabación**
- **Conclusiones**
- **Demostración**

PASEMOS A LA ACCION...

The image shows a screenshot of a PIC simulator interface. It consists of three main windows: a Source Browser, a register viewer, and a terminal window.

Source Browser: Displays the assembly code for a program named 'suma.asm'. The code includes comments in Spanish, a license notice, and assembly instructions. A green arrow points to the instruction `MOVLW 1`.

```
*****
;* suma.asm. Juan Gonzalez Gomez. Octubre 2
*****
;-----
;* Programa ejemplo que realiza la suma 1 + 2
;* deposita el resultado en la variable SUMA
;-----
; LICENCIA GPL
*****
;--- Especificar el PIC a emplear
LIST P=16F873
INCLUDE "p16f873.inc"
;--- Variables
SUMA EQU 0x20
;--- Comienzo del programa
ORG 0
MOVLW 1
ADDLW 2
MOVWF SUMA
fin GOTO fin
END
```

Register Viewer: Shows the state of the PIC registers. The 'indf' register is highlighted in red. The status bar at the bottom indicates: Status: 0x18 W: 0x00 PC: 0x0000 Cycles: 0x0000000000000000 Time: 0.00 µs.

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	00	00	00	18	00	00	00	00			00	00	00	00	00	00
10	00	00	00			00	00	00	00	00	00	00	00	00	00	00
20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80	00	ff	00	18	00	3f	ff	ff			00	00	00	00	00	

Terminal: Shows the command line execution of the simulator.

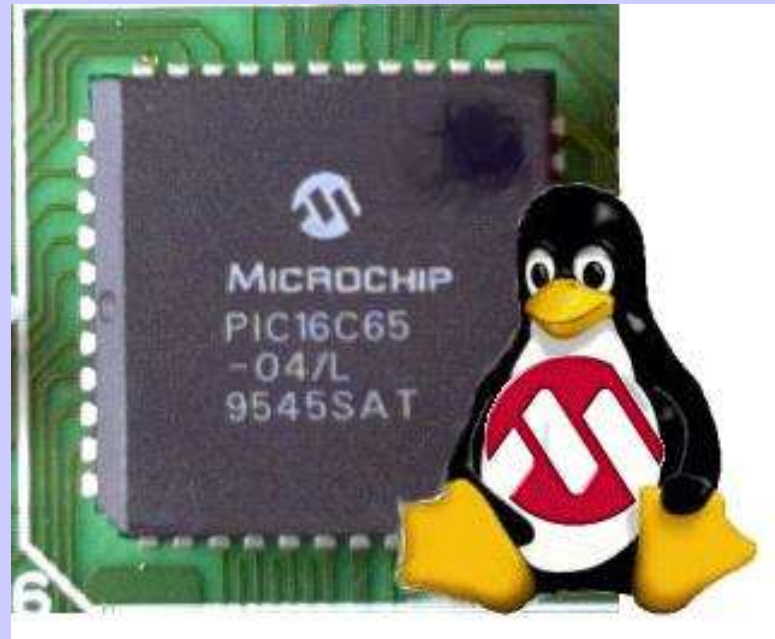
```
juan@milenium:~/pic$ gpsim -s suma.cod
gpsim - the GNU PIC simulator
version: 0.20.14
type help for help
gpsim> Loading suma.cod
processing cod file suma.cod
f873 create
symbol at address 3 name fin
--- Reset
Enabling WDT timeout = 0.018 seconds
gpsim>
```

GUI Window: A control window titled '0.20.14' with buttons for 'step', 'over', 'finish', 'run', 'stop', and 'reset'. It also features a 'gui_update' button and a numeric input field containing '1048567'. A 'Quit gpsim' button is at the bottom.

Enlaces:

- **GPUTILS: Ensamblador y otras herramientas para PICs**
 - <http://gputils.sourceforge.net/>
- **GPSIM: Simulador**
 - <http://www.dattalo.com/gnupic/gpsim.html>
- **SKYPIC-DOWN: Grabador**
 - <http://www.iearobotics.com/personal/juan/proyectos/skypic-down/skypic.html>
- **Charla/taller en el Chat, sobre Microcontroladores PIC y Linux**
 - <http://www.iearobotics.com/proyectos/charlas-irc/pic-linux/log-26-oct-2003.html>
- **Tarjetas PICMIN y PICUPSAM:**
 - <http://www.iearobotics.com/personal/andres/proyectos/picmin/pic1.html>
- **TARJETA CT6811:**
 - <http://www.iearobotics.com/proyectos/ct6811/ct6811.html>

Herramientas hardware y software para el desarrollo de aplicaciones con Microcontroladores PIC bajo plataformas GNU/Linux



Juan González Gómez
Escuela Politécnica Superior
Universidad Autónoma de Madrid

Andrés Prieto-Moreno Torres
Ifara Tecnologías
Profesor asociado UPSAM