

# Bootloader y primer programa ED 2

Cómo cargar el bootloader y el primer programa en el microcontrolador y  
verificar que todo está funcionando correctamente

## Temas

- Programación de microcontrolador PIC
  - IDE (Integrated Development Environment)
- Programador
- Bootloader
- Programación del bootloader en el PIC16f887
  - Indicadores led en estado inicial
  - Todo salió correcto
  - Algo falló
- Carga de programa de prueba
  - Circuito mínimo necesario
  - Descarga y Configuración software AN1310
  - Modo Bootloader del microcontrolador
  - Carga del archivo .hex
  - Verificación de funcionamiento del primer programa
  - Reentrar en modo bootloader para cargar nuevos programas

# Programación de microcontrolador PIC

A lo largo de la asignatura vamos a escribir varios programas destinados a controlar el comportamiento de un microcontrolador, en nuestro caso el pic16f887. El código que escribiremos, inicialmente en lenguaje ensamblador y posteriormente en lenguaje C, debe ser compilado y/o ensamblado para traducirlo a lenguaje máquina. El IDE (Entorno integrado de desarrollo) que utilizaremos será el MPLAB X, en particular, y solo temporalmente será la versión 5.35.

<https://www.microchip.com/en-us/tools-resources/archives/mplab-ecosystem>



# Programación de microcontrolador PIC

Se le llama entorno integrado de desarrollo porque nos permite escribir el código, compilar y ensamblar, linkear, depurar (debug), simular, y programar (a través de un programador) el microcontrolador.



ensamblador  
compilador      archivo .hex  
simulador



# Entorno de desarrollo

## Vista del IDE MPLAB X

Cada vez que escribamos nuestro código y lo compilemos nos dará como resultado un archivo .hex

The screenshot shows the MPLAB X IDE interface. The top menu bar includes File, Edit, View, Navigate, Source, Refactor, Production, Debug, Team, Tools, Window, Help. The title bar indicates "MPLAB X IDE v5.30 - segunda\_prueba : default". The left sidebar displays the "Projects" view with two projects: "digIBoard\_asm" and "segunda\_prueba". The "segunda\_prueba" project is selected. The main workspace contains three tabs: "Start Page" (selected), "PIC16 Bootloader.asm", and "bootconfig.inc". The "PIC16 Bootloader.asm" tab shows assembly code:

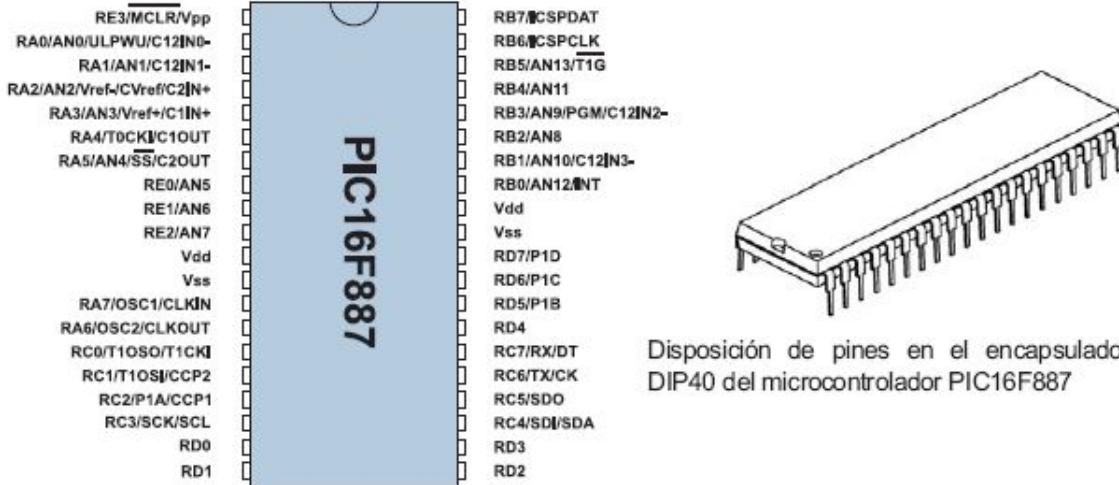
```
82    COMMAND      equ STARTBUFFER
83    ADDRESS_L    equ STARTBUFFER+1
84    ADDRESS_H    equ STARTBUFFER+2
85    ADDRESS_U    equ STARTBUFFER+3
86    ADDRESS_X    equ STARTBUFFER+4
87    DATA_COUNTL   equ STARTBUFFER+5
88    PACKET_DATA  equ STARTBUFFER+6
89    DATA_COUNTH   equ PACKET_DATA
90
91    #if BOOTLOADER_ADDRESS == 0
92    #ifndef BSR
93        PCLATH_TEMP    equ 0x7E
94        W_TEMP         equ 0x7F
95    #endif
96    #endif
97    ; ****
98
```

The bottom right pane shows an "Output - Project Loading Warning" message: "Some of the files in this Project 'segunda\_prueba' contain spaces or odd characters in thei".

The bottom left pane shows the "Project Properties" for the "segunda\_prueba" project, including sections for Device, Compiler Toolchain, Memory, and Debug Resources.

# Programador

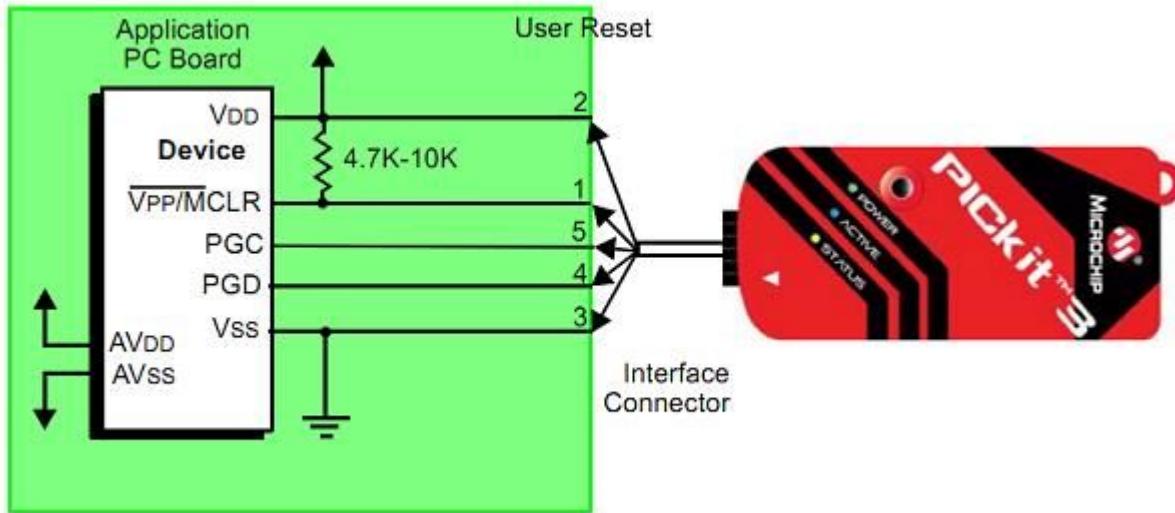
Una vez generado el archivo .hex lo cargaremos en nuestro microcontrolador y para esto necesitaremos un programador pickit 3 (obsoleto), pickit 4 (a punto de ser obsoleto), pickit 5, Real ICE... clones pickit 2/3, etc.



Disposición de pines en el encapsulado  
DIP40 del microcontrolador PIC16F887

## Programador

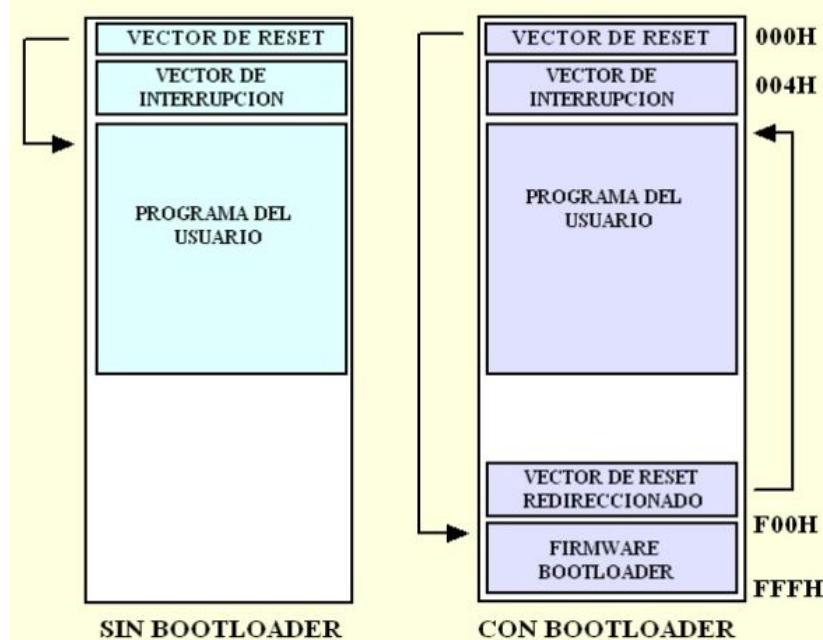
El programador, por ejemplo pickit3, necesita conectarse a 5 pines del microcontrolador para poder cargar el archivo .hex



# Bootloader

Por una cuestión de disponibilidad de programadores con soporte a la fecha y fundamentalmente por una cuestión de los costo asociados a estos, existe una alternativa para cargar nuestro programa .hex en el microcontrolador que consiste en programar el microcontrolador con un Bootloader (cargador de arranque).

El bootloader es un pequeño código (FIRMWARE BOOTLOADER) que se encargará de recibir el archivo .hex mediante comunicación serie y lo cargará en la memoria de programa (PROGRAMA DEL USUARIO)



## Programación del bootloader en el PIC16f887

La programación del bootloader se debe hacer de la forma clásica, utilizando un programador, y algún software que permita la carga del archivo .hex en el microcontrolador.

En particular, el programador Pickit 3, entre otros, cuenta con el modo “Programmer To Go” que permite dejar un programa .hex en el mismo programador, listo para cargarse en un microcontrolador, sin la necesidad de conectarlo a una computadora.



# Programación del bootloader en el PIC16f887

## - Indicadores Led en estado inicial

Al conectar la fuente de alimentación, el pickit 3 enciende sus 3 leds (POWER, ACTIVE y STATUS)



## Programación del bootloader en el PIC16f887

- Indicadores Led en estado inicial

Luego, solo quedarán encendidos el Led POWER permanente y el led ACTIVE intermitente



## Programación del bootloader en el PIC16f887

### - Indicadores Led en estado inicial

Ahora es momento de colocar el microcontrolador en el zócalo ZIF.

**PRESTAR ESPECIAL ATENCIÓN A LA POSICIÓN DEL MICROCONTROLADOR**

Muesca del encapsulado



# Programación del bootloader en el PIC16f887

## - Indicadores Led en estado inicial

Se baja la palanca para conseguir la conexión de los pines.



## Programación del bootloader en el PIC16f887

### - Indicadores Led en estado inicial

Presionamos el pulsador que forma parte del programador Pickit 3

El led STATUS se iluminará en color rojo por un periodo breve de tiempo



## Programación del bootloader en el PIC16f887

- Todo salió correcto

Si todo ha salido correctamente, el led STATUS quedará en verde permanente, indicando que el código se cargó correctamente en el microcontrolador

El led ACTIVE parpadeará en azul indicando que está en espera de programar el próximo microcontrolador



Programación del bootloader en el PIC16f887

- Todo salió correcto

Felicitaciones, tu  
microcontrolador ya tiene  
cargado el bootloader, ya falta  
menos!!!



## Programación del bootloader en el PIC16f887

- Algo Falló (este paso es solo ilustrativo, no es necesario llevarlo a cabo)

Simularemos una mala conexión y presionaremos el pulsador del microcontrolador.



## Programación del bootloader en el PIC16f887

- Algo Falló (este paso es solo ilustrativo, no es necesario llevarlo a cabo)

El led STATUS quedará parpadeando en color rojo (un parpadeo corto, y uno largo) indicando que no pudo realizarse la correcta programación del microcontrolador.

Verifique:

- que el microcontrolador esté presente
- palanca del zócalo
- conexión entre pickit3 y zócalo

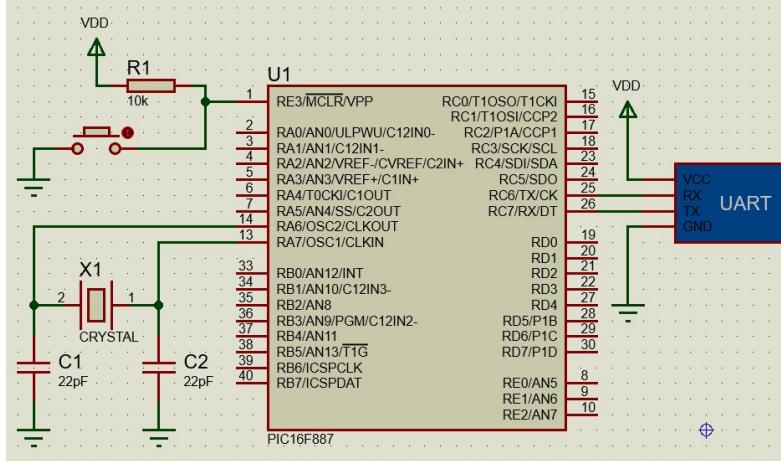
si todo sigue fallando informe a los docentes o ayudantes.



## Carga de programa de prueba - Circuito mínimo necesario

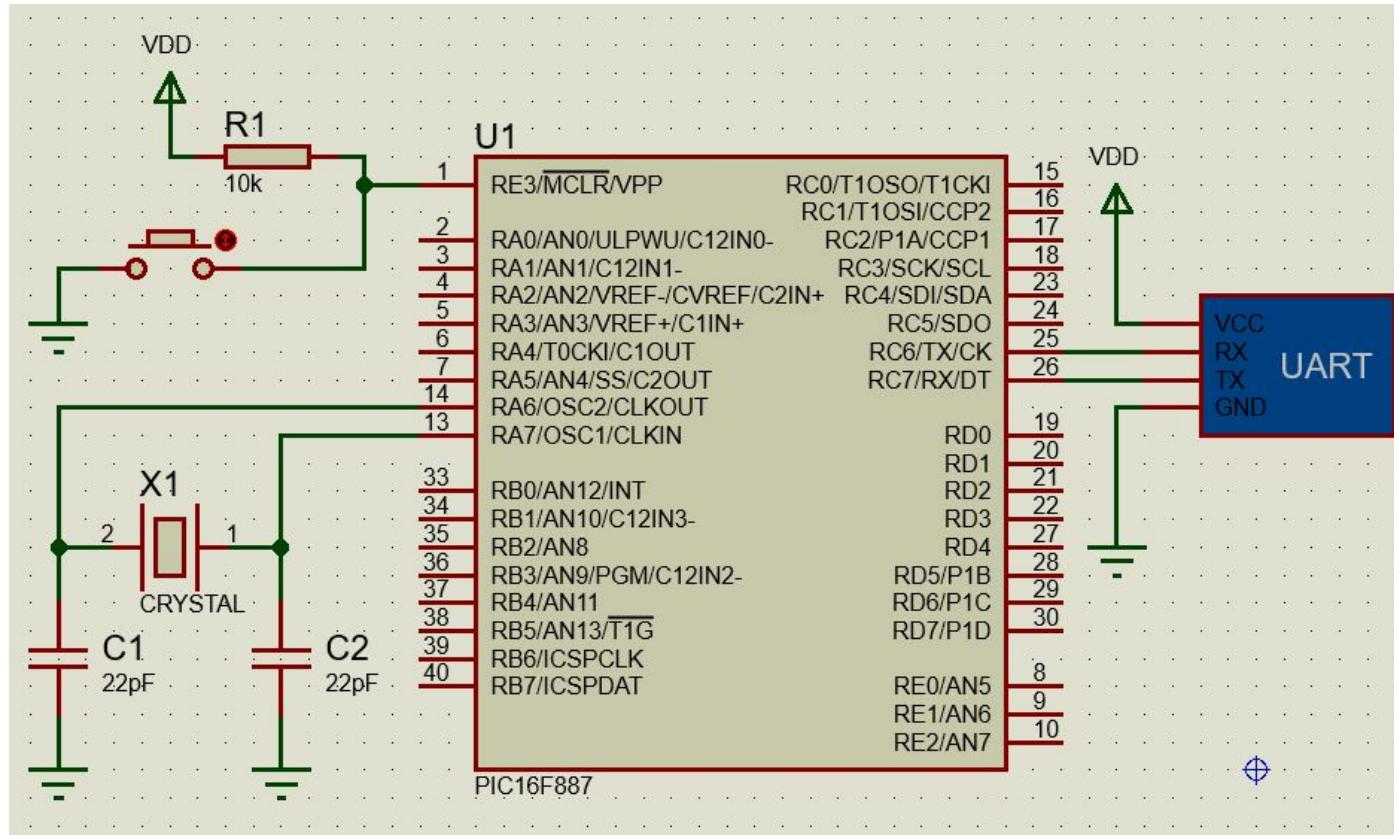
Recordando que el bootloader debe conectarse a la PC para recibir el archivo .hex a través del puerto serie y cargarlo en la memoria de programa, necesitaremos un circuito mínimo para verificar que todo está bien.

Para funcionar, el bootloader necesita una conexión serie con la PC (adaptador USB/UART), necesita además un cristal de cuarzo que le brinde la base de tiempo y por último, un resistor y un pulsador conectados a MCLR/VPP. (MCLR es el reset del microcontrolador, con el pulsador lo podremos resetear y en particular configurarlo en modo BOOTLOADER, que será necesario para poder cargar programas)



Si bien el circuito no lo indica, debe conectarse VDD y VSS del microcontrolador a los pines de 5 V y GND del adaptador USB/UART

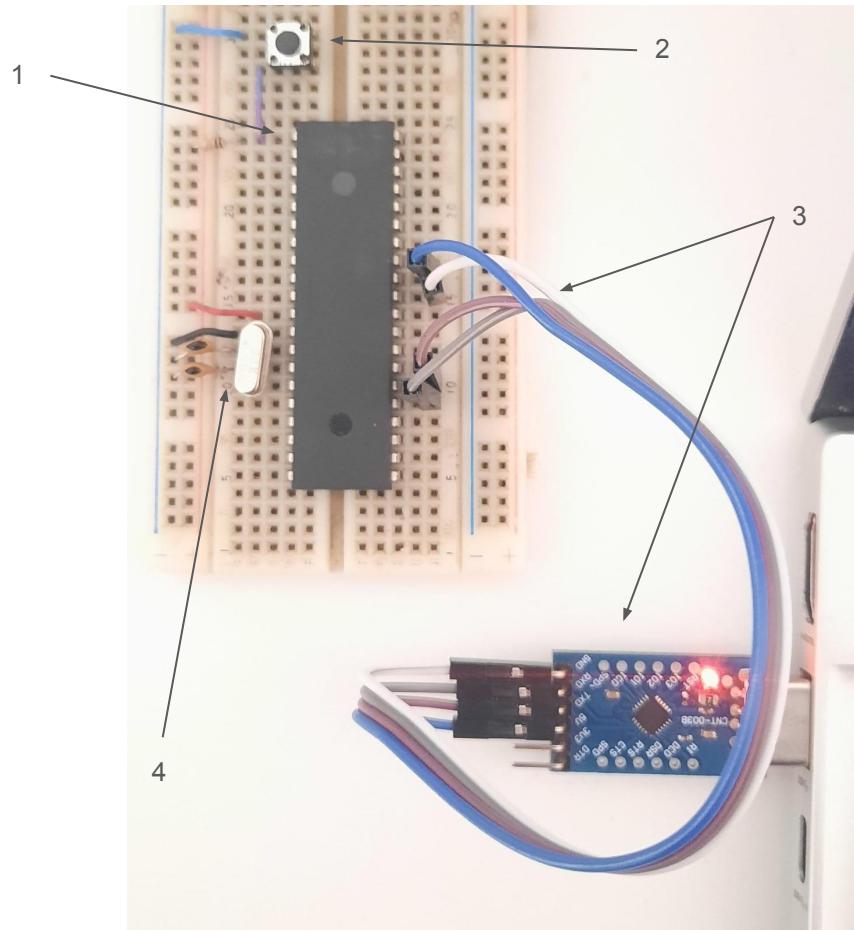
# Carga de programa de prueba - Circuito mínimo necesario



## Carga de programa de prueba - Circuito mínimo necesario

Puede verse en la imagen, cada uno de los componentes:

1. El pin 1 (MCLR) conectado a VDD a través de un resistor (10 kOhm)
2. El pulsador que permite cortocircuitar MCLR a VSS/GND
3. El adaptador USB/UART conectado al microcontrolador (TX, RX, VDD/5V, VSS/GND)
4. El cristal conectado a OSC1 y OSC2 y luego un capacitor conectando cada una de esas patas a GND.



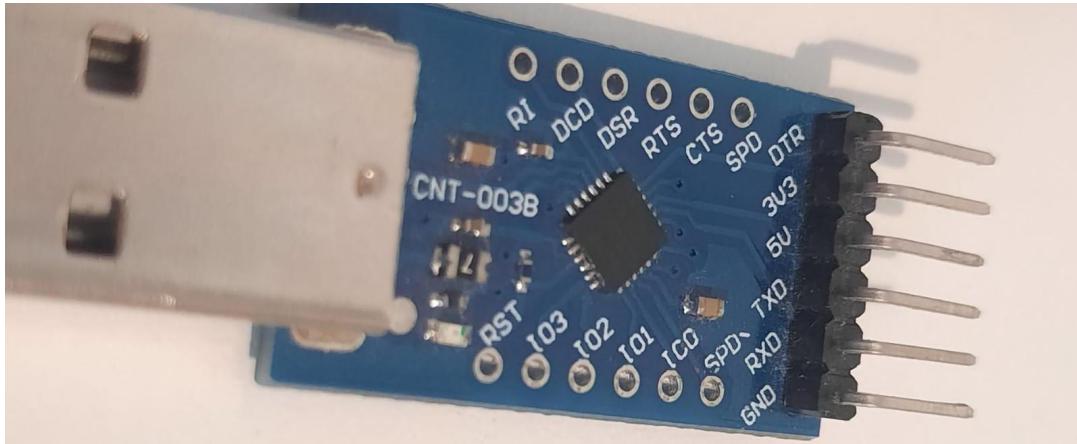
## Carga de programa de prueba - Circuito mínimo necesario - Cristal 4 MHz y cap 22 pF

Como referencia, en una casa de electrónica de nuestra ciudad el cristal de 4 Mhz cuesta al rededor de 500 pesos, equivalente a 0.4 dólares en la actualidad.



Capacitor tipo lenteja 22 pF, 33 pF, 47 pF,  
valores en ese orden funcionan bien

## Carga de programa de prueba - Circuito mínimo necesario - adaptador USB/UART



Casi sin excepción, los adaptadores USB/UART cuentan con TX, RX, GND y VDD que puede ser 5 V o 3v3 (3,3 V), **es recomendable que tenga salida de 5 V**. Esto es porque alimentaremos el microcontrolador directamente de los 5 V del puerto USB

**Tener especial cuidado con los cortocircuitos ya que se puede dañar el puerto USB de la computadora**

## Carga de programa de prueba - Circuito mínimo necesario - adaptador USB/UART

Para que el bootloader pueda cargar el archivo .hex en la memoria de programa del microcontrolador debemos emplear una conexión UART pero debido a que las PC en general no cuentan con una conexión de este tipo, debemos utilizar un adaptador. Existen muchos adaptadores basados en diferentes chips que realizan la tarea de “traducir” de USB a UART. Entre estos chips tenemos FT232, CP2102, CH340, MCP2200.

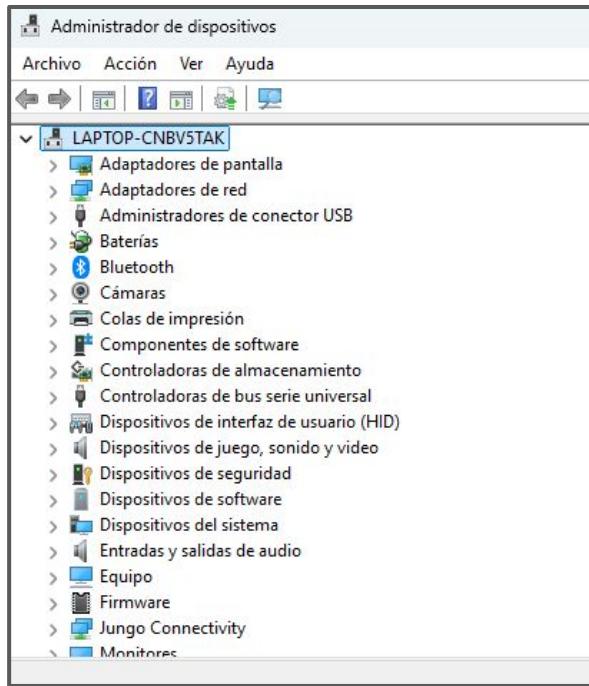
**Evitar comprar adaptadores basados en el chip PL2303 ya que muchas versiones baratas no cuentan con chips originales y generan fuertes inconvenientes con los drivers**

# Carga de programa de prueba - Circuito mínimo necesario - adaptador USB/UART



# Carga de programa de prueba - Circuito mínimo necesario - adaptador USB/UART

Una vez que tenemos un adaptador USB/UART proseguiremos a conectarlo a la PC e iremos a Inicio -> Administrador de dispositivos, esto nos abrirá una ventana como la siguiente:



Iremos a la opción  
Puertos COM y LPT

- > Mouse y otros dispositivos señaladores
- > Objetos de procesamiento de audio (APO)
- > Procesadores
- > Puertos (COM y LPT)
  - シリコンラボ CP210x USB to UART Bridge (COM12)
- > Sensores

Debemos recordar el puerto  
COM que la PC nos asignó,  
COM12 en este caso

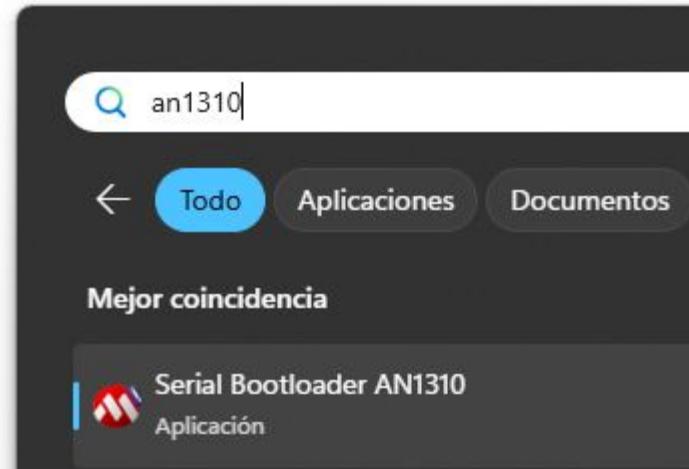
si no figura o figura con un cartel amarillo puede que  
estén fallando los drivers, instalarlos o actualizarlos

## Carga de programa de prueba - Descarga y Configuración software AN1310

Descargaremos la AN1310 que es la “application note”. Se descargará una aplicación .exe que instalaremos, además de documentación relacionada con el bootloader.

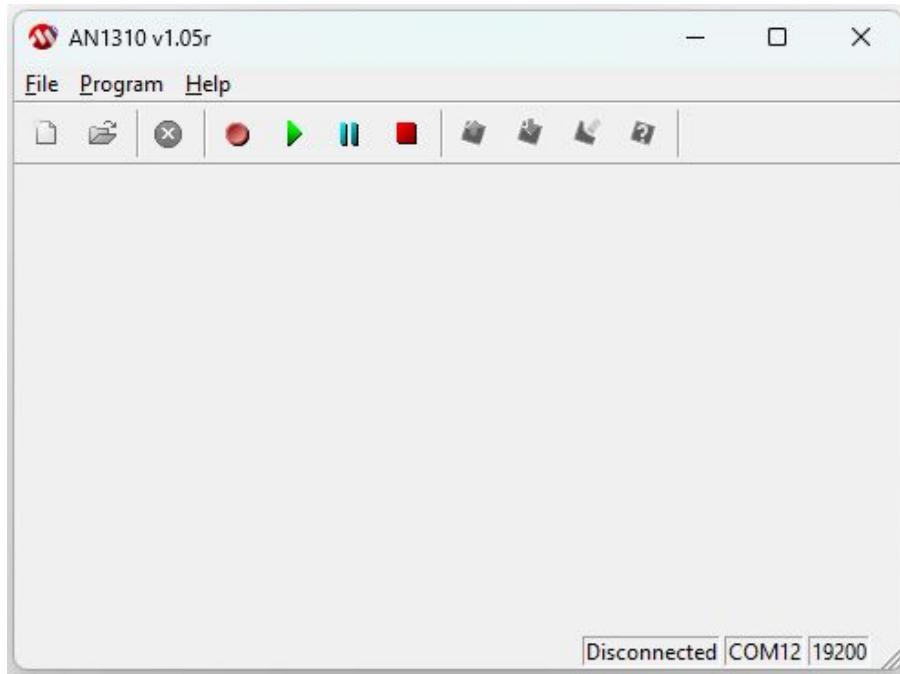
<https://ww1.microchip.com/downloads/en/AppNotes/Serial%20Bootloader%20AN1310%20v1.05r.zip>

Una vez instalada la aplicación la buscaremos como AN1310 en inicio

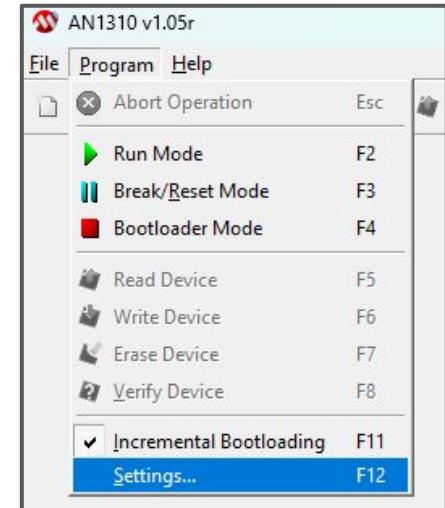


# Carga de programa de prueba - Descarga y Configuración software AN1310

Se abrirá la siguiente ventana

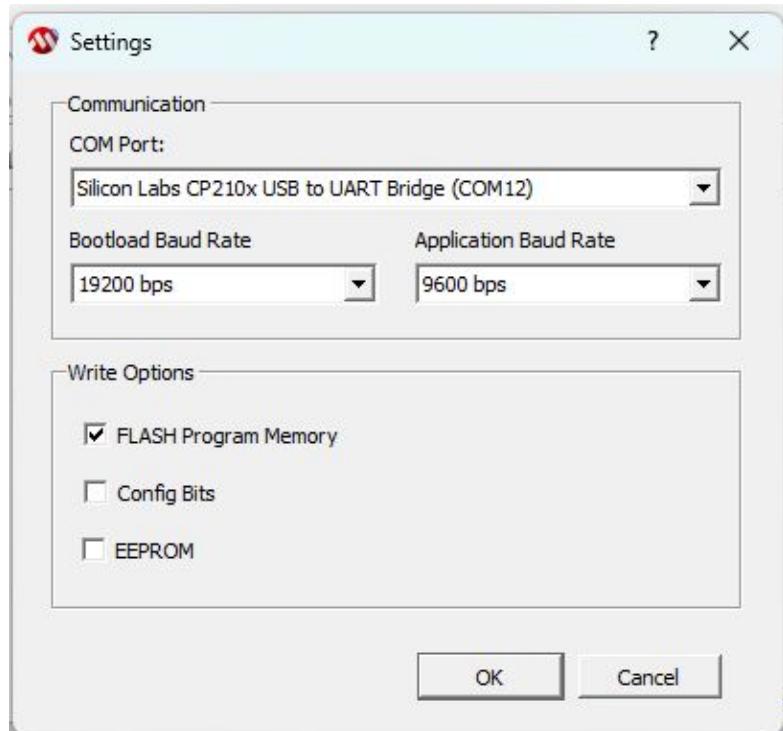


Iremos a la pestaña Program, a la opción settings



# Carga de programa de prueba - Descarga y Configuración software AN1310

Se abrirá la siguiente ventana



En esta ventana seleccionaremos el puerto COM correspondiente al adaptador USB/UART que figuraba en el “Administrador de dispositivos”.

Luego cargaremos 2 velocidades (BAUD RATE)

- Bootload Baud Rate, correspondiente a la velocidad a la que se cargará el programa (19200 bps) a través del puerto serie.
- Application Baud Rate, correspondiente a la velocidad de comunicación configurada en nuestro código de prueba (9600 bps).

hacemos click en OK y se cerrará la ventana.

## Carga de programa de prueba - Modo Bootloader del microcontrolador

Volviendo a la ventana Inicial. Si solo tenemos cargado el bootloader y no le hemos cargado todavía ningún programa, debemos presionar el botón “Bootloader Mode” (cuadrado rojo)



# Carga de programa de prueba - Modo Bootloader del microcontrolador

El paso anterior debe mostrarnos la ventana de inicio de la siguiente forma

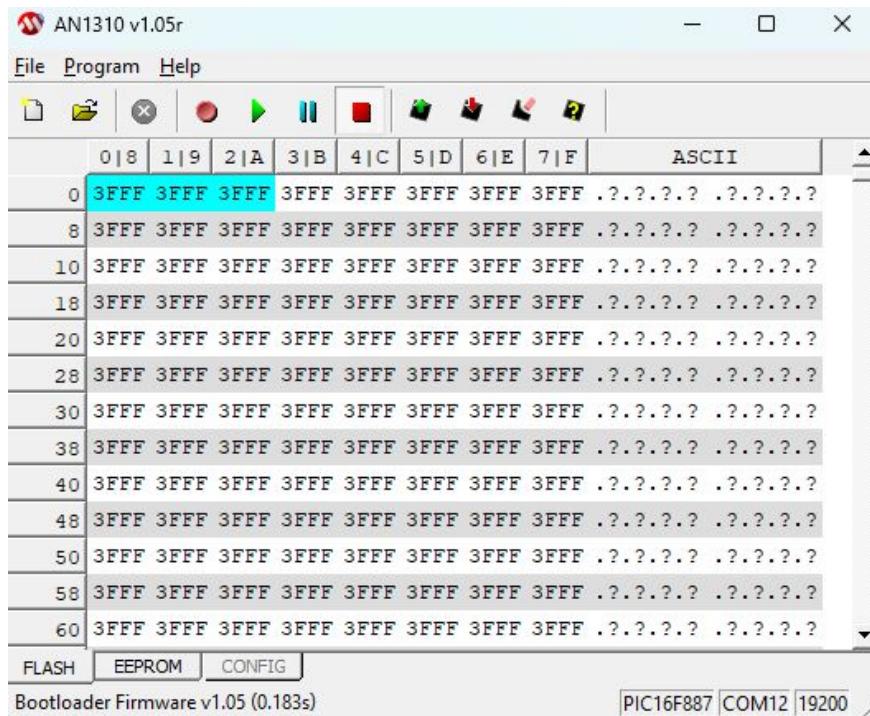
The screenshot shows the AN1310 v1.05 software interface. The main window displays a memory dump for the PIC16F887 microcontroller. The dump consists of two columns: hex values and ASCII characters. The hex values are mostly '3FFF' with some '?' characters, indicating memory that has not been programmed. The ASCII column shows a series of question marks ('?'). The software interface includes a toolbar with various icons, a menu bar with File, Program, and Help, and a status bar at the bottom with the text 'Bootloader Firmware v1.05 (0.183s)', 'PIC16F887', 'COM12', and '19200'. The tabs at the bottom are labeled FLASH, EEPROM, and CONFIG, with FLASH being the active tab.

	0 8	1 9	2 A	3 B	4 C	5 D	6 E	7 F	ASCII
0	3FFF	.?.?.?.?.?.?.?.?							
8	3FFF	.?.?.?.?.?.?.?.?							
10	3FFF	.?.?.?.?.?.?.?.?							
18	3FFF	.?.?.?.?.?.?.?.?							
20	3FFF	.?.?.?.?.?.?.?.?							
28	3FFF	.?.?.?.?.?.?.?.?							
30	3FFF	.?.?.?.?.?.?.?.?							
38	3FFF	.?.?.?.?.?.?.?.?							
40	3FFF	.?.?.?.?.?.?.?.?							
48	3FFF	.?.?.?.?.?.?.?.?							
50	3FFF	.?.?.?.?.?.?.?.?							
58	3FFF	.?.?.?.?.?.?.?.?							
60	3FFF	.?.?.?.?.?.?.?.?							

En esta ventana tenemos mucha información importante

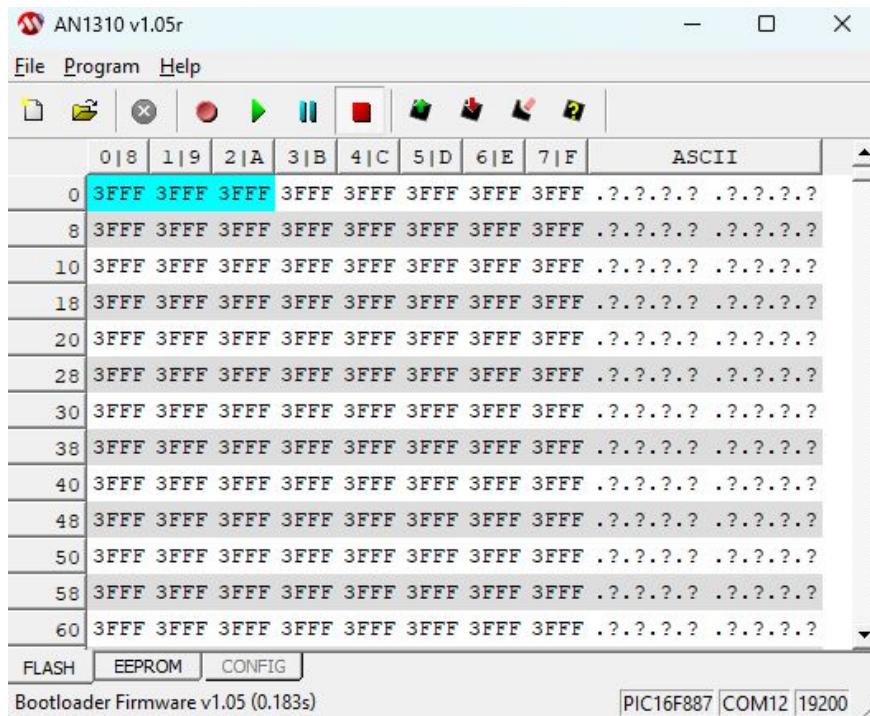
- Bootloader Firmware v1.05 (inferior izquierda) lo que indica que el bootloader fue detectado correctamente, lo que también significa que fue correctamente programado con el Pickit3.

# Carga de programa de prueba - Modo Bootloader del microcontrolador



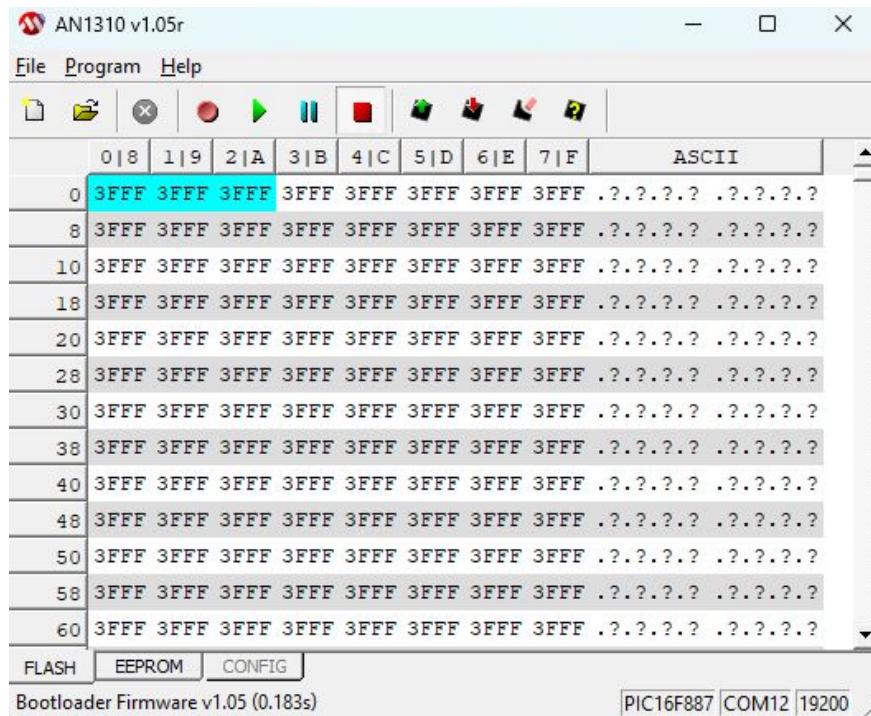
- “PIC16F887-COM12-19200” (inferior derecho) que indica el puerto COM correctamente asignado y la velocidad de comunicación del bootloader

# Carga de programa de prueba - Modo Bootloader del microcontrolador



- La parte central de la ventana muestra el código que se cargará en el microcontrolador, en particular, en celeste están marcadas las direcciones ocupadas por el bootloader.

## Carga de programa de prueba - Carga del archivo .hex



En este punto estamos en condiciones de cargar nuestro primer programa en el microcontrolador.

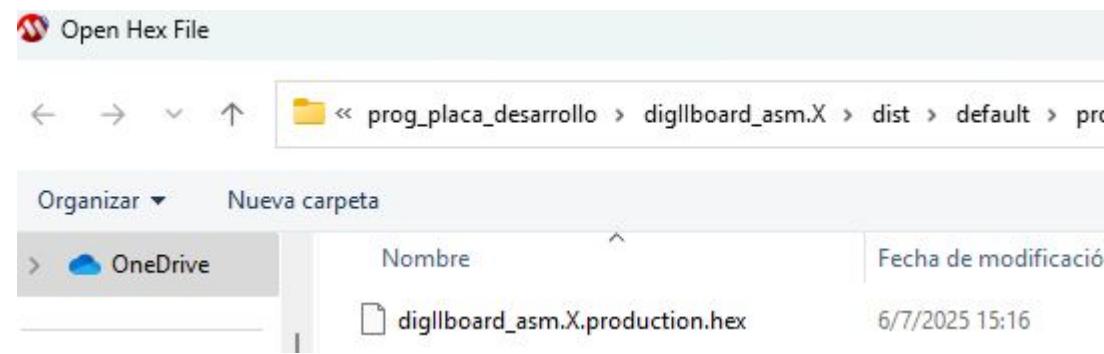
## Carga de programa de prueba - Carga del archivo .hex

Para este punto vamos a cargar un programa .hex de prueba que estará disponible para descargar por el alumno.

Iremos a File -> Open

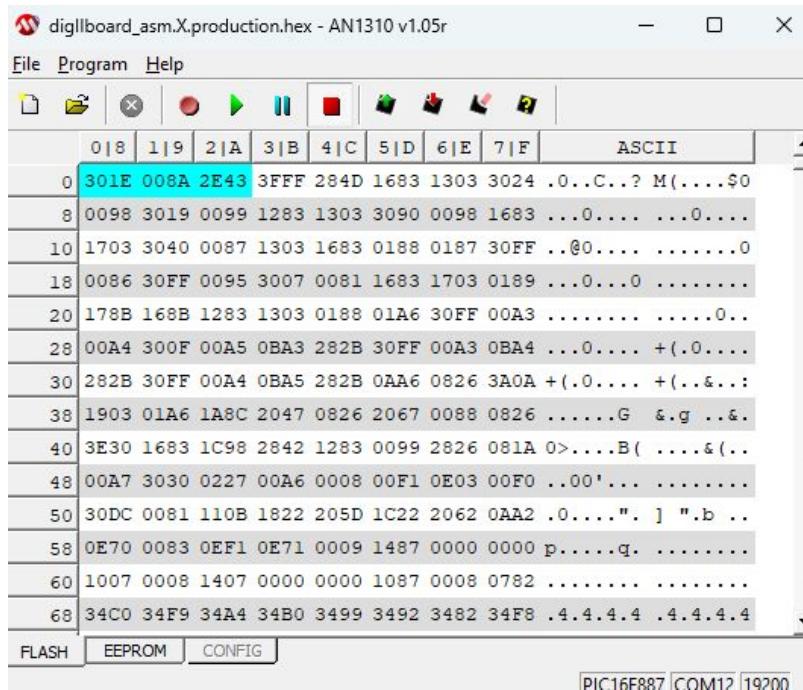


Seleccionaremos el archivo .hex



## Carga de programa de prueba - Carga del archivo .hex

Ahora la ventana principal nos muestra el código que se cargará en el microcontrolador. Recordemos que en celeste está marcado el código del bootloader, el resto es el código correspondiente a nuestra aplicación.

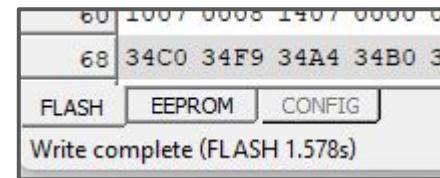


The screenshot shows the digIBoard software interface with the file 'digIboard\_asm.X.production.hex' open. The code is displayed in a table with columns for address, data, and ASCII representation. A red box highlights the first few bytes (0x301E, 0x008A, 0xE43) which represent the bootloader. The software has a menu bar with File, Program, and Help. Below the menu is a toolbar with various icons. At the bottom, there are tabs for FLASH, EEPROM, and CONFIG, and a status bar indicating 'PIC16F887 COM12 19200'.

Haremos click en la opción “write Device” (flecha roja hacia abajo)

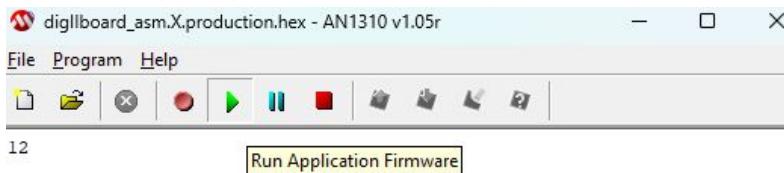


En la parte inferior izquierda de la ventana, se nos informará que se completó la escritura, y el tiempo que tomó



# Carga de programa de prueba - Verificación de funcionamiento del primer programa

El siguiente paso es hacer click en el botón “Run Application Firmware” botón de play color verde



Esta acción hace salir del modo bootloader al microcontrolador y comienza a correr el código de prueba que se le suministró. Adicionalmente abre un terminal serie que se conecta al microcontrolador (COM12, 9600). El código de prueba que se le suministró envía por puerto serie una cuenta que va incrementando de uno en uno y se va desplegando en el terminal de la aplicación AN1310. Si le es posible ver esta cuenta, significa que todo está funcionando como debe.

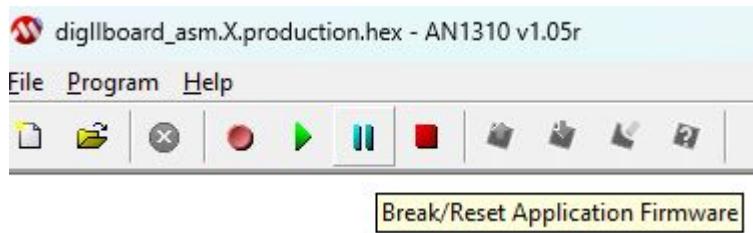
Carga de programa de prueba - Verificación de funcionamiento del primer programa

**Felicitaciones, ha cargado su primer programa en el microcontrolador y verificó que se comunique correctamente con la PC**

# Carga de programa de prueba - Reentrar en modo bootloader para cargar nuevos programas

Lo único que nos falta ver es cómo hacemos para reentrar en modo bootloader para cargar los archivos .hex de futuros proyectos

Para esto,



- Hacemos click en “Break/Reset Application Firmware” (botón de pausa celeste)
- Luego presionamos el pulsador que colocamos en la protoboard que cortocircuita MCLR con VSS.
- Presionamos el botón “bootloader mode” (cuadrado rojo) y deberíamos entrar en modo bootloader nuevamente, en condiciones de cargar un nuevo código.