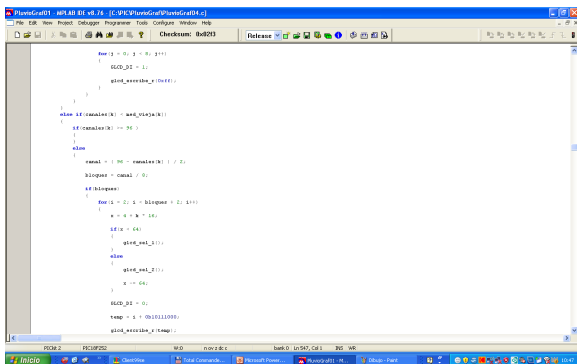
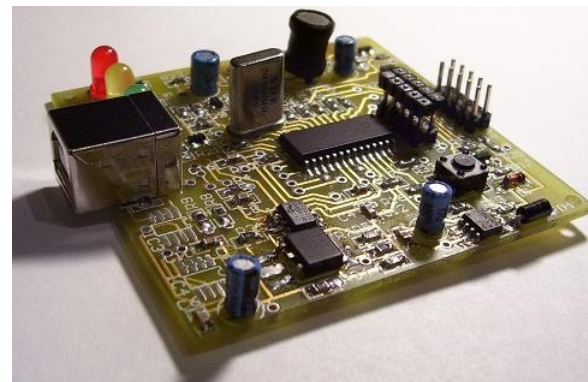


EMULACIÓN EN TIEMPO **REAL CON PIC**

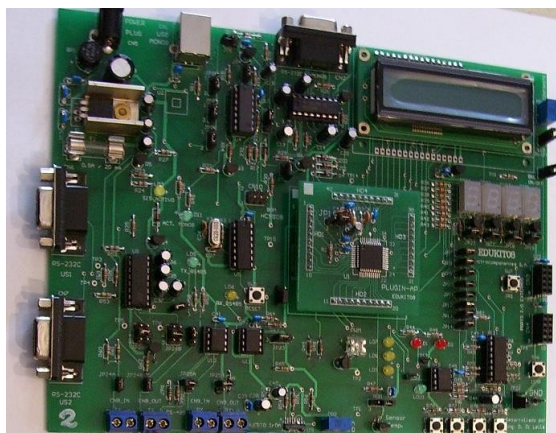
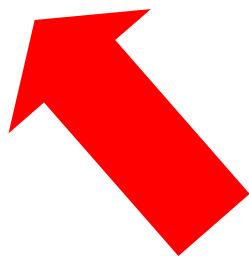
El proceso de desarrollo



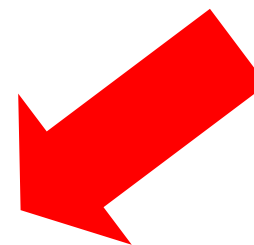
```
int main() {
    // ...
    while (1) {
        // ...
    }
}
```



Escribo el código fuente



Programo el μ C



Hago la experiencia

Simulación por software

- Un programa de PC imita el comportamiento del μC
- Las variables y registros se actualizan virtualmente
- Pueden detectarse errores de lógica en los algoritmos

Limitación de la simulación

- No pueden simularse eventos propios del hardware
- Ruidos eléctricos, interferencia de 50Hz y glitches no pueden simularse
- Malas configuraciones de los periféricos pueden impedir la actualización de un puerto
- En simulación la RAM amanece en 0x00

Conclusiones

- En simulación todo está correcto
- En la práctica el programa no funciona bien
- El μC me lo hace a propósito/me odia

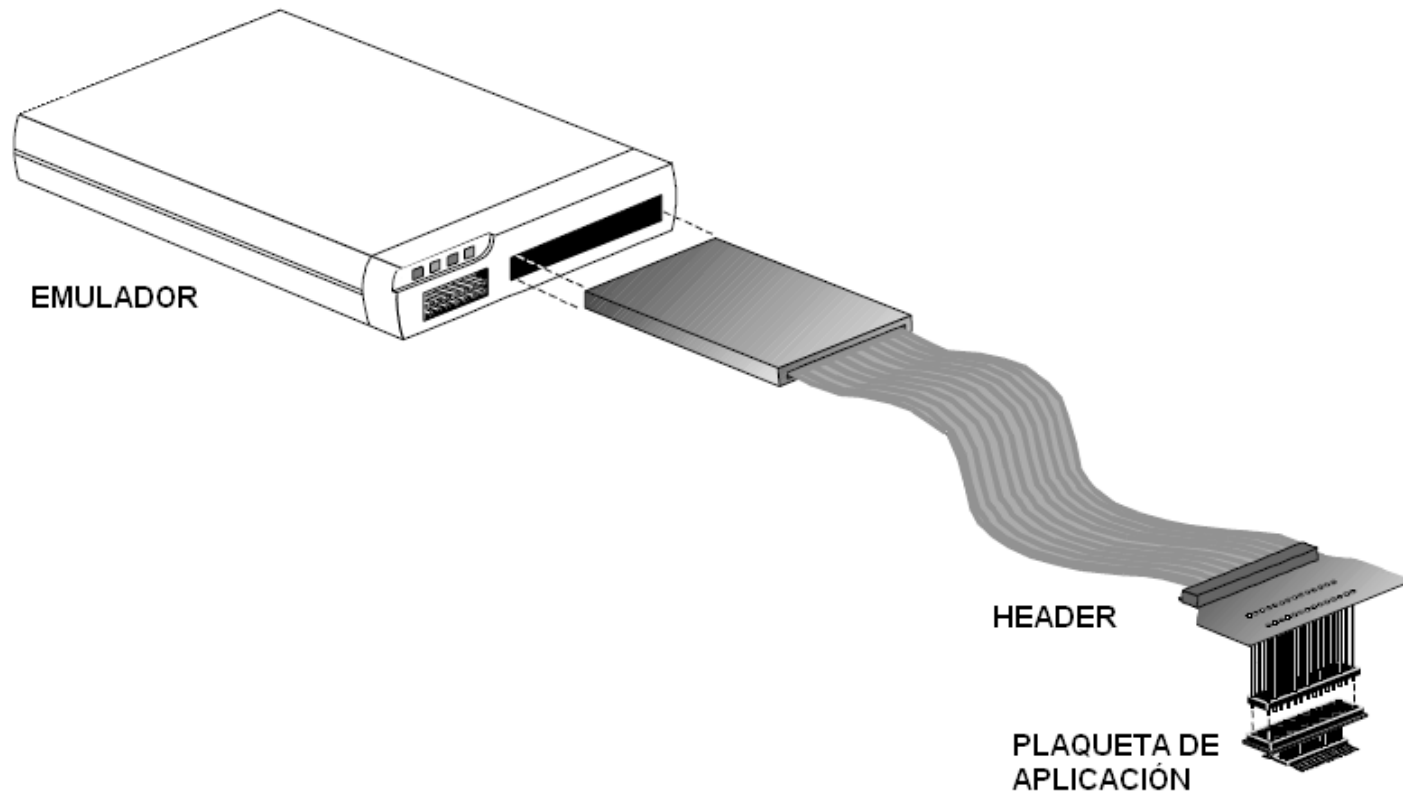
Debug en tiempo real

- Se busca interactuar con el hardware real donde correrá la aplicación
- Puede emplearse un equipo que reemplace al μC en el zócalo de aplicación
- Puede emplearse un μC que se comuniquen con la PC mientras corre en la aplicación
- El firmware de aplicación debe correr durante el Debug a la misma velocidad que fuera de Debug

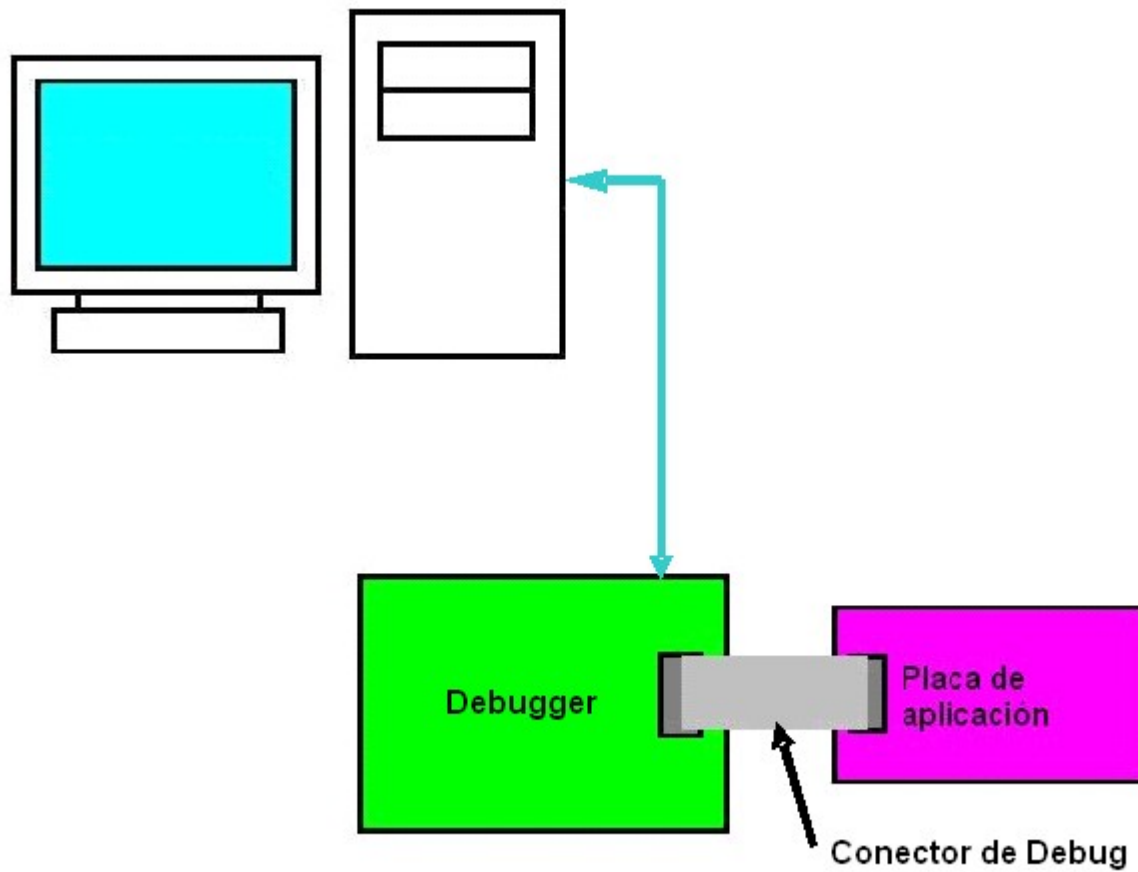
Emulación y simulación

- La emulación corre a la misma velocidad que un programa real; la simulación no
- La emulación interactúa con el hardware real; la simulación no siempre
- La emulación permite ver cambios reales en el estado interno del μC ; la simulación solo imagina que pasaría en el μC

ICE

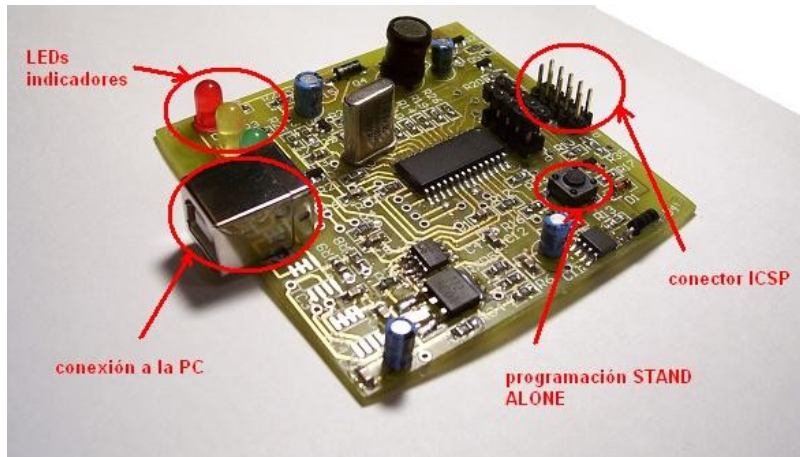


Debugger

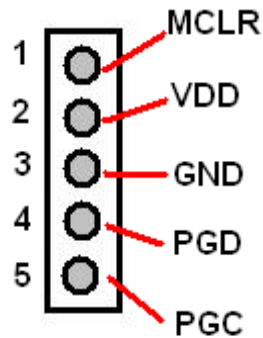


El PICkit 2

- Integrado al MPLAB
- Alimentación por USB
- Puede alimentar la aplicación
- $VDD = 1,8V - 5V$
- Detecta cortocircuitos en VDD y VPP
- Familias PIC10F, PIC12F, PIC16F, PIC18F, dsPIC30F, dsPIC33F
- Programación STAND-ALONE
- 2 breakpoints
- Emula desde 32KHz



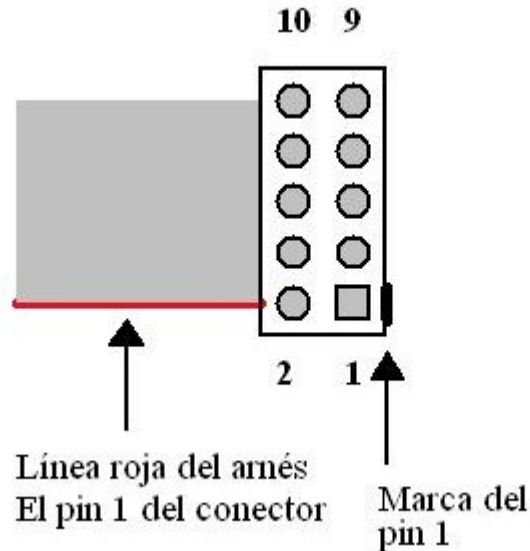
Conexión a 5 hilos



Conector tradicional de Microchip

Conector PICKit 2 de
Edudevices

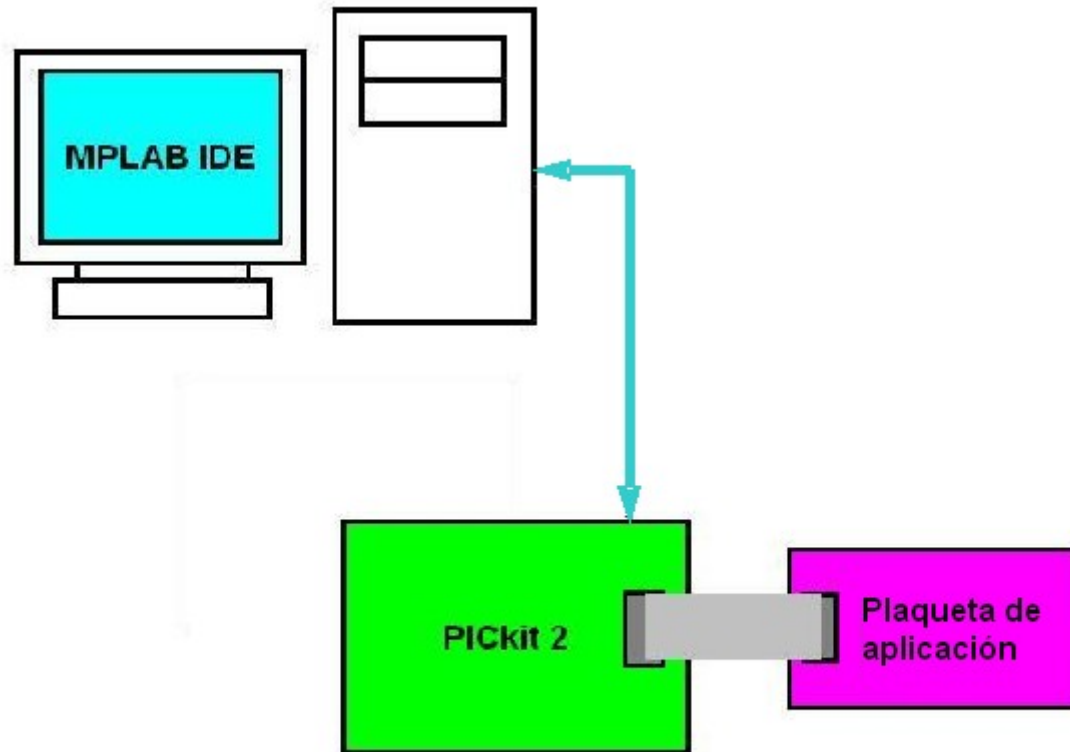
Hacia el PICKit2



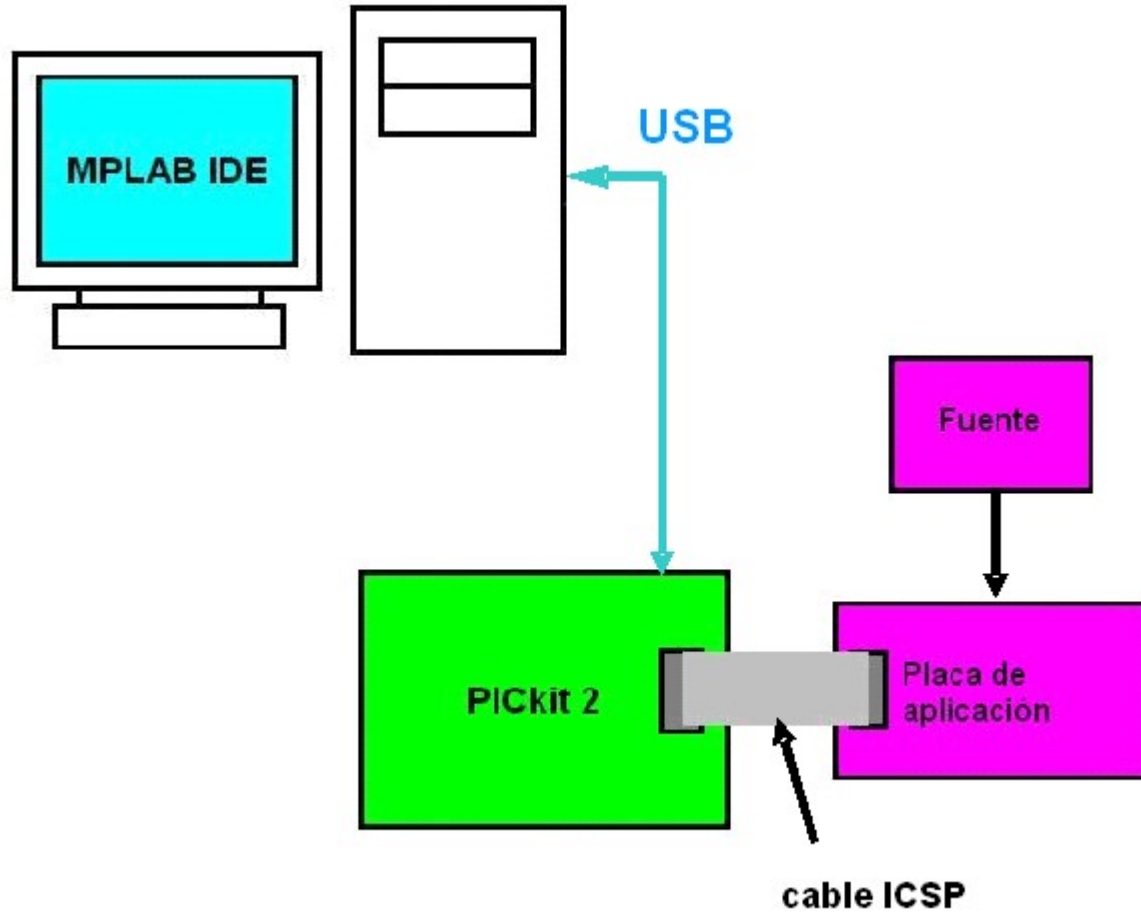
Disposición de pines:

- 1) MCLR
- 2) GND
- 3) VDD
- 4) GND
- 5) GND
- 6) AUX
- 7) PGD
- 8) GND
- 9) PGC
- 10) GND

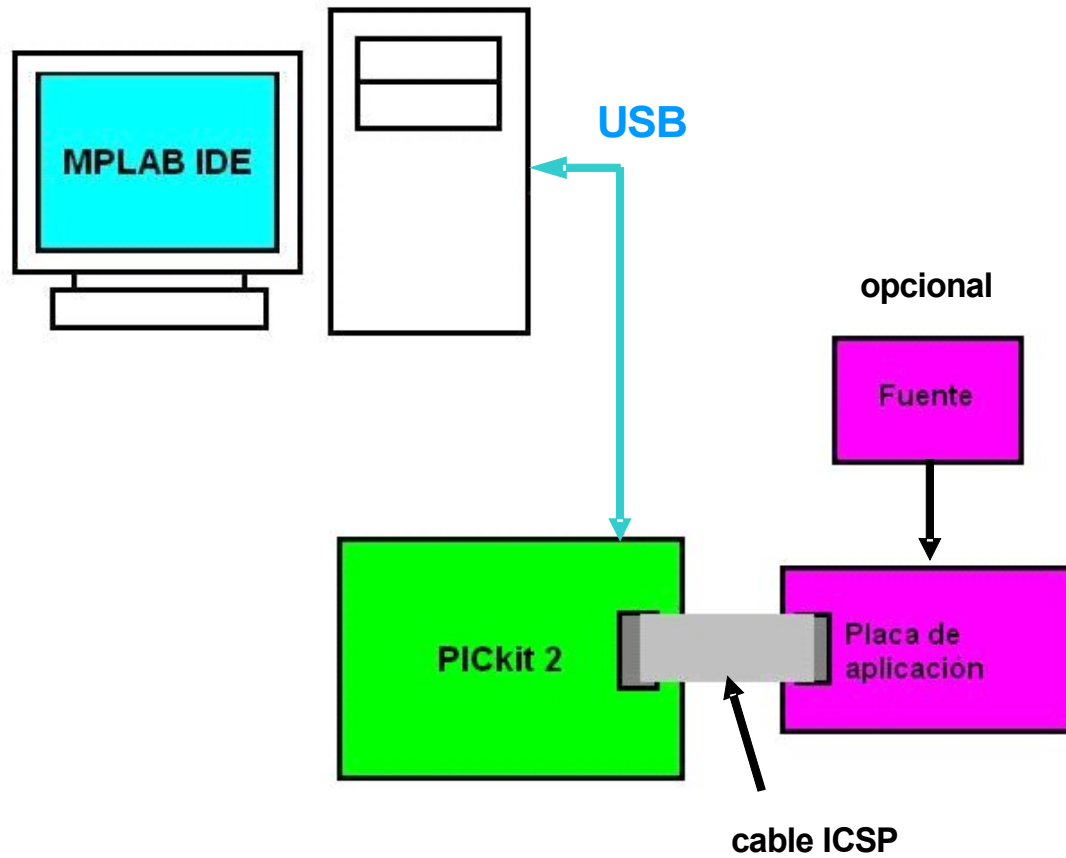
Esquema de programación



Con alimentación externa

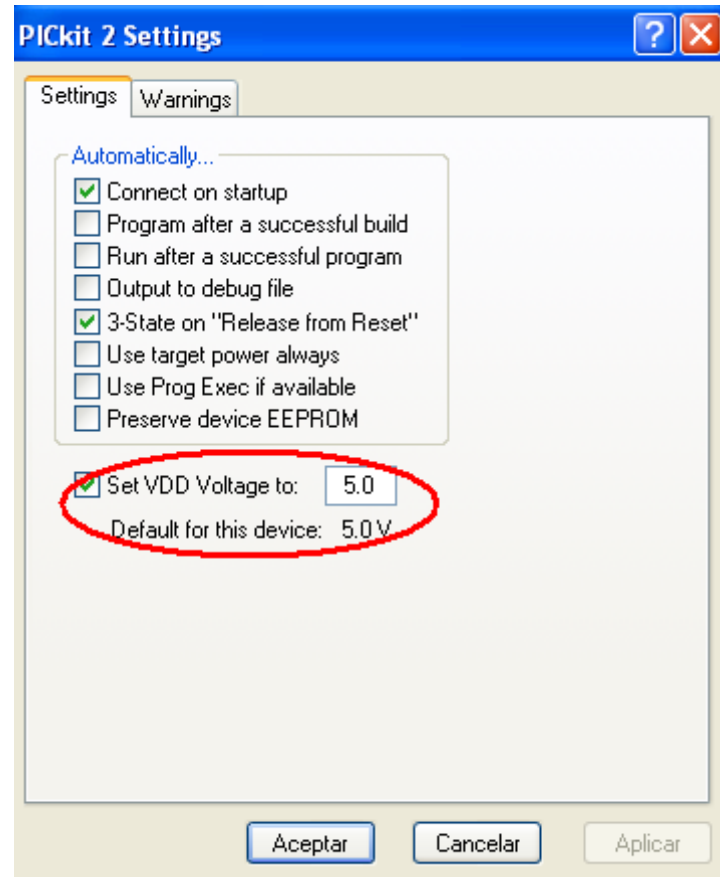


Esquema para el Debug



Alimentación desde el PICkit 2

- No debe haber reguladores de tensión
- Puede existir un jumper para relevar la fuente externa
- Corriente limitada por USB



Breakpoints

- Permite correr en tiempo real hasta esa dirección
- Permite saber si el CPU pasa por esa instrucción alguna vez
- Permite ver la evolución de los registros hasta ese punto.



```
bcf      STATUS,C
rrf      nota,W
movwf   CCPR1L

movf    nota,W
banksel PR2
movwf   PR2

banksel PORTA

goto    isr_fin
```


Breakpoint skidding

```
*****
;*
;*      INICIALIZACIÓN DE PERIFÉRICOS Y VARIABLES      *
;*
;*
*****

inicio
  call  configura ← Desde aquí se ejecutó
                    el programa

*****
;*
;*      PROGRAMA PRINCIPAL      *
;*
;*
*****

ciclo
  bsf   LED_ROJO      ; Enciende LED rojo,
                    ; aplicando un '1' en el puerto

  |     call  demora_1s ; Llama a la subrutina de demora
                    ; de 1 segundo, basada en ciclo
                    ; anidados.

  bcf   LED_ROJO      ; Apaga LED rojo
                    ; aplicando un '0' en el puerto.
```

Limitaciones del Debug

- El PICkit 2 utiliza recursos del μ C en modo Debug
- Al modificarse el programa el μ C debe reprogramarse
- Breakpoints limitados
- No puede usarse la instrucción SLEEP
- No puede usarse el Watchdog timer durante el Debug
- La pila no está visible
- La frecuencia mínima de ejecución es 32KHz

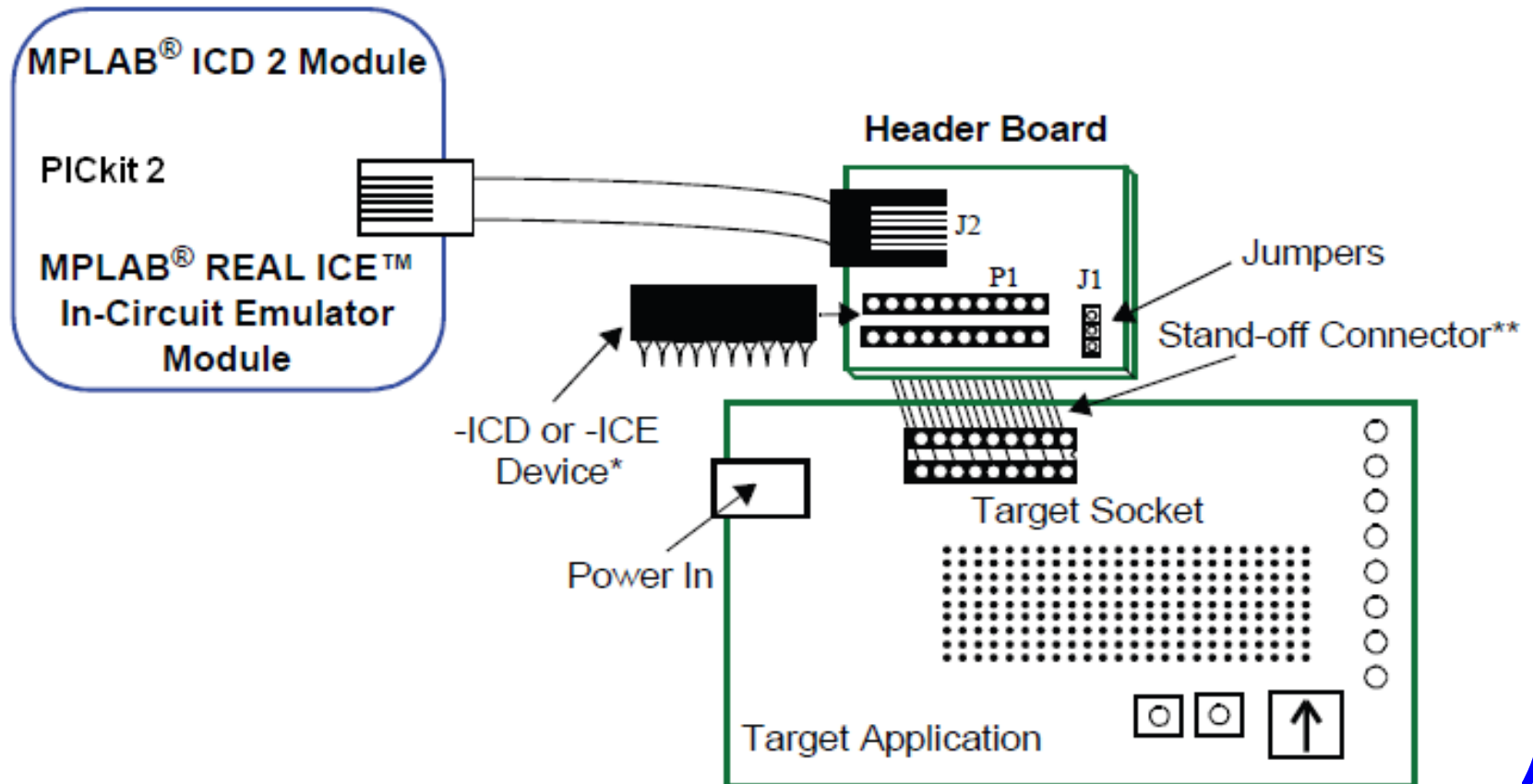
El modo release

- Elimina recursos utilizados en modo Debug
 - Debug Executive
 - Registros temporales en RAM
- La aplicación definitiva debe estar en modo Release

MCU de pocos pines

- μ C de las familias PIC12F y PIC16F tienen pocos pines
- Utilizar MCLR, PGD, PGC para el Debug deja al μ C sin E/S
- En la familia PIC16F se presenta esta situación en μ C de 18 ó menos pines

Header boards



Problemas y errores frecuentes

- ¿Porqué el “*PICkit 2*” no detecta el MCU cuando quiero programarlo?
- ¿Cuál es la razón por la que el MCU es detectado, programado o entra en modo Debug solo alguna veces?
- ¿A qué se debe que el MPLAB IDE me informe un cortocircuito en VDD si todas las conexiones entre el “*PICkit 2*” y el MCU es correcta?
- ¿Cómo sé si el “*PICkit 2*” puede programar al PIC que estoy utilizando?