Micro controllers

introduction

Areas of use & Numbers of machines

- You might have 1 or 2 Pentium class chips at home. You will have perhaps 50 to 100 other embedded computers in other devices.
- If you think of children's toys, the numbers grow even higher.

Characteristics

- Embedded computers have to be very low cost, simple and reliable.
- They can not use any moving parts (disk drives) because:

1. These are power hungry
2. They are bulky
3. They are expensive

Features

- Program in Flash Memory
- Limited RAM storage – variables only not code
- Built in I/O devices
- Use very little power

Families

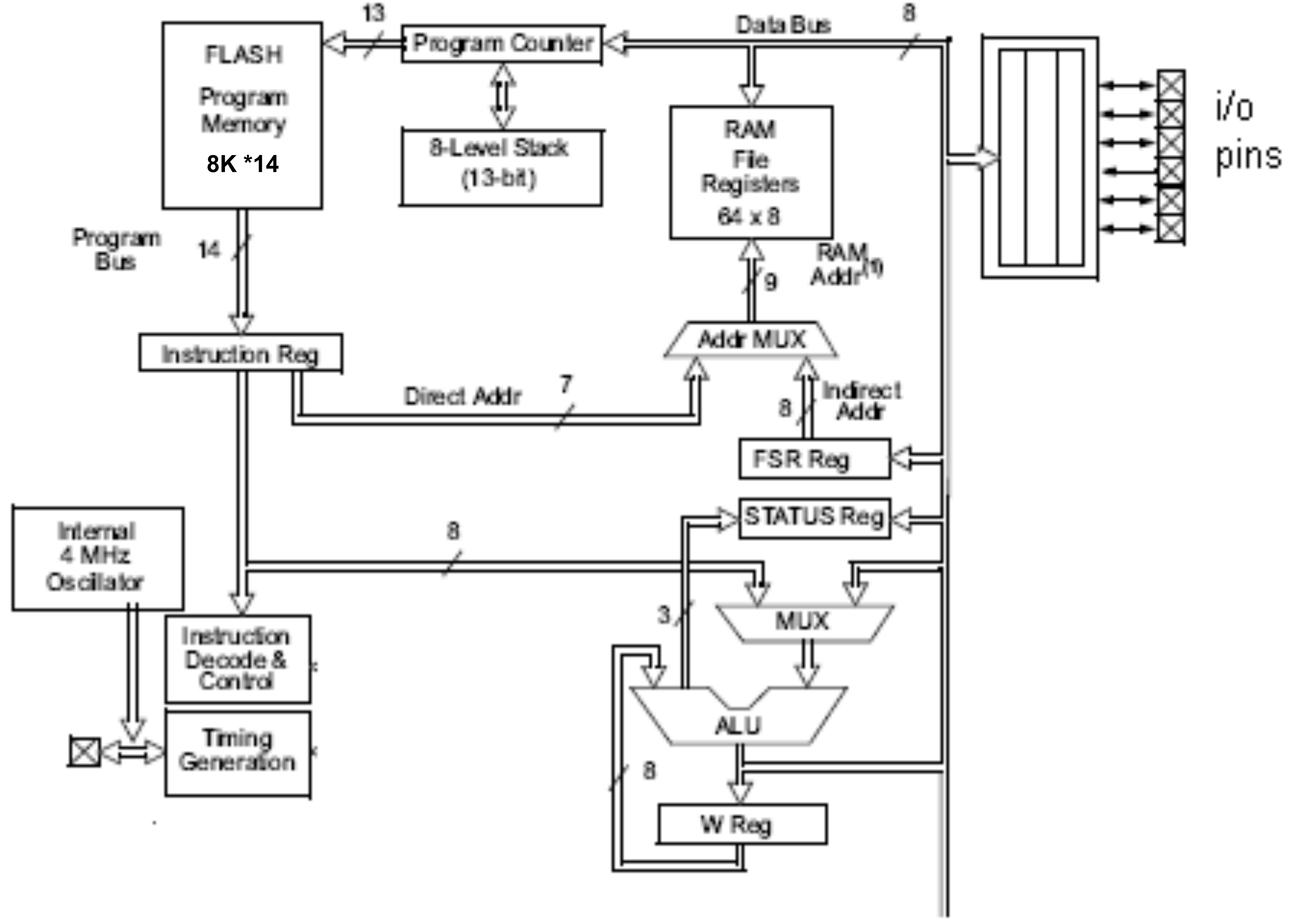
- Zilog Z8 series
- Intel 8051 series
- Arm 32 bit microcontrollers
- MicroChip – PIC microcontrollers

PIC (Microchip)

- Range of low end 8 bit microcontrollers.
- smallest have only 8 pins, largest 40 pins.
- Very cheap, you can pick them up at less than €1 each.
- Targeted at consumer products, alarms etc.

Harvard architecture

- Like many micros the PIC is a Harvard machine
- Different word lengths for instructions (14 bit) and data (8 bit).



Basic PIC architecture

to i/o
registers

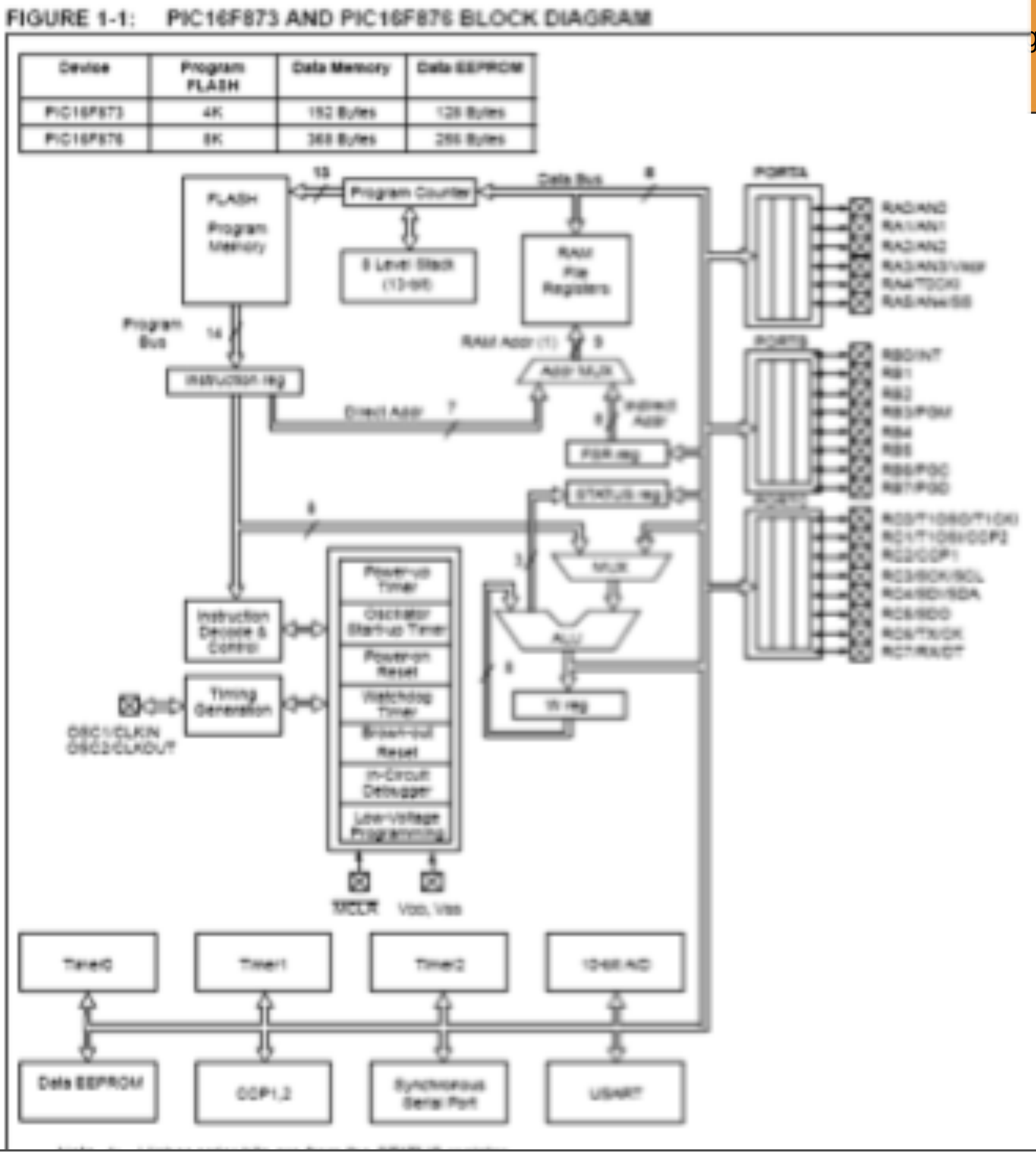


FIGURE 1-1: PIC16F873 AND PIC16F876 BLOCK DIAGRAM

Device	Program FLASH	Data Memory	Data EEPROM
PIC16F873	4K	192 Bytes	128 Bytes
PIC16F876	8K	368 Bytes	256 Bytes

I/O Ports

Pins are multifunctional

- input
- output
- interrupt
- analog
- Timer

C-compiler

- not case-sensitive
- types: short 1 bit (bladz 29?66)

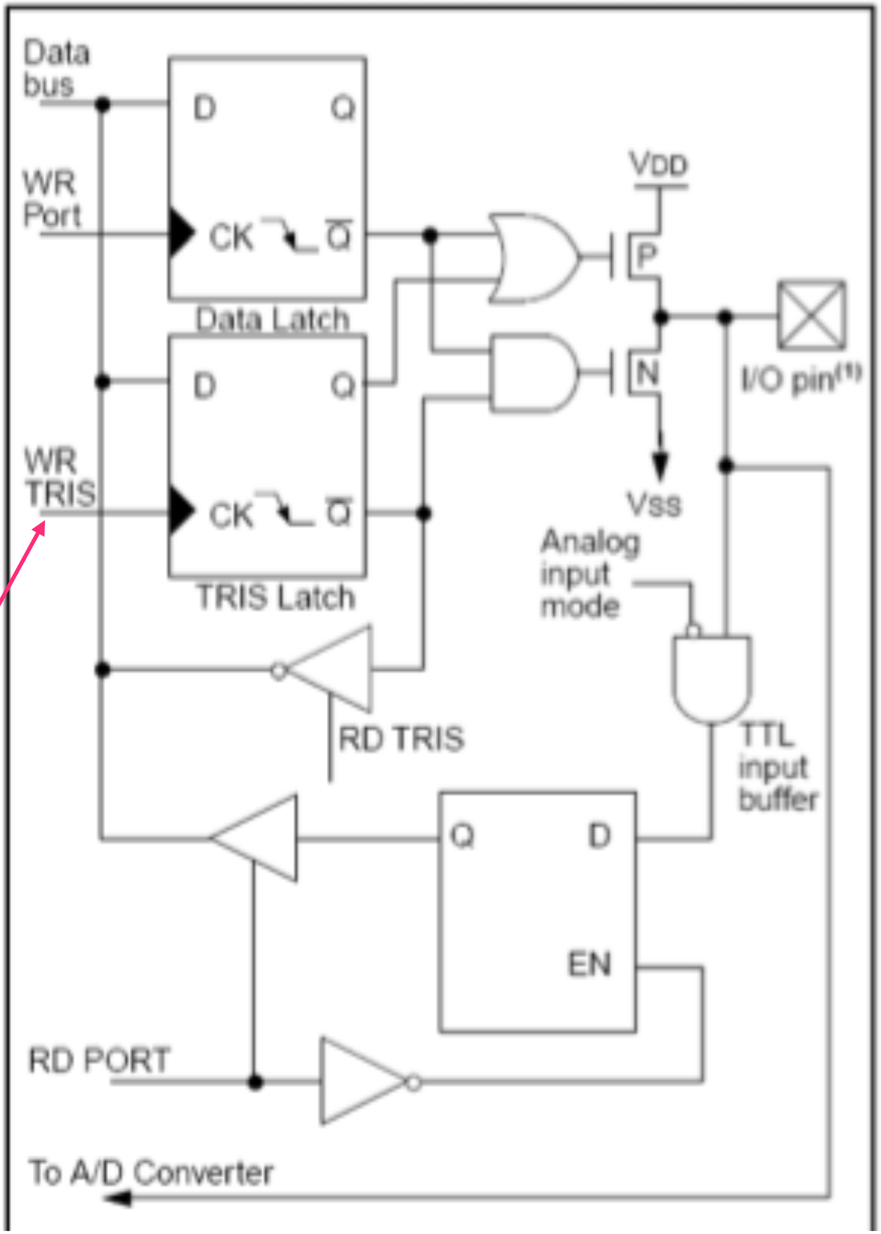
I/O-statements:

- output_bit(PIN_B0, 0);
- output_low(PIN_B0);
- output_high(PIN_B0);
- while(!input(PIN_B1));

//waits for B1 to go high

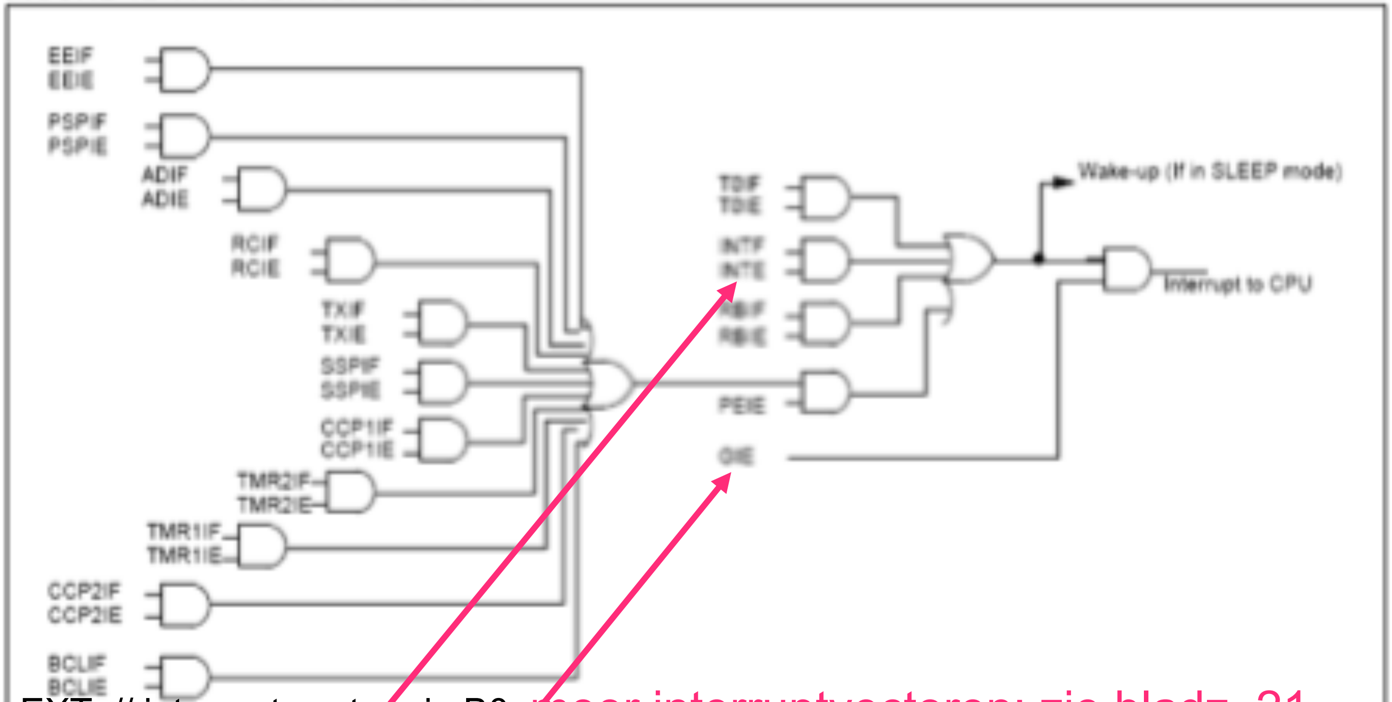
- set_tris_B(0x0F);

FIGURE 3-1: BLOCK DIAGRAM OF RA3:RA0 AND RA5 PINS



Interrupt Logic

FIGURE 12-11: INTERRUPT LOGIC

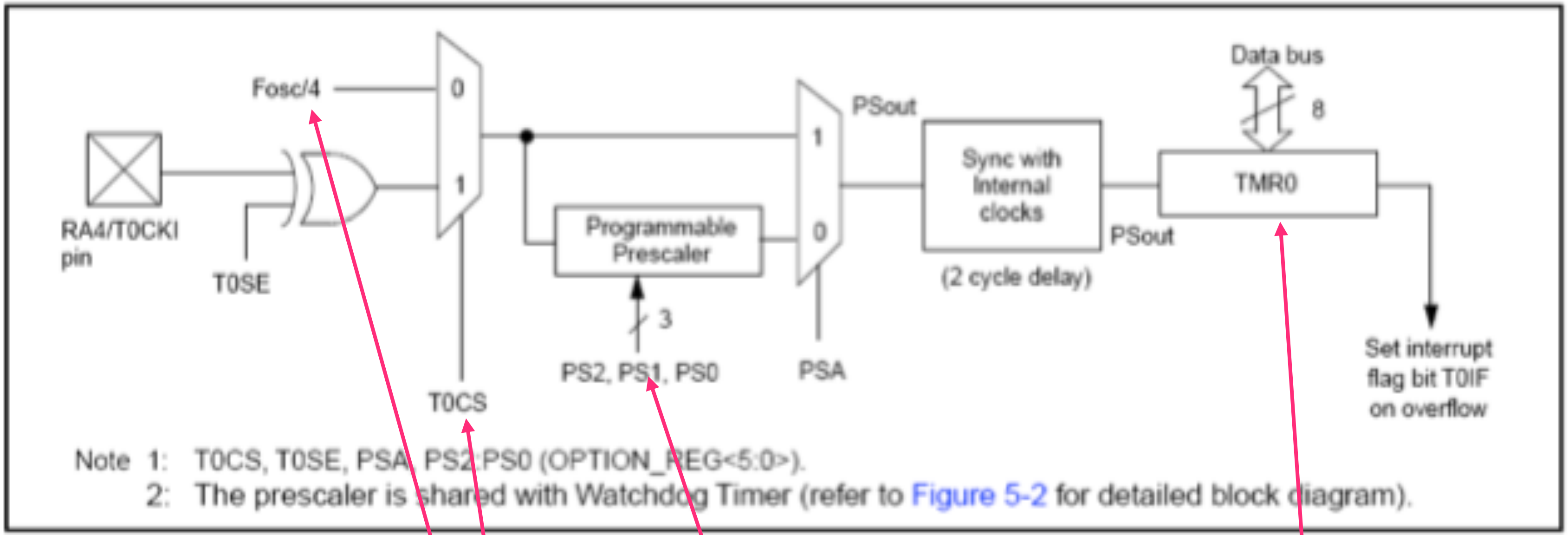


```
#INT_EXT  // interrupt vector pin B0.
pinB0handler() {
… // your interrupt handler
}
Main() {
….
enable_interrupts( GLOBAL );       // zie bladz.41 en 42
enable_interrupts( INT_EX );
…..
```

meer interruptvectoren: zie bladz. 21

Timer 0 (8 bits) ook “RTCC” genaamd

FIGURE 5-1: TIMER0 BLOCK DIAGRAM



Note 1: T0CS, T0SE, PSA, PS2:PS0 (OPTION_REG<5:0>).
2: The prescaler is shared with Watchdog Timer (refer to Figure 5-2 for detailed block diagram).

if(get_rtcc() == 25) set _rtcc(0);

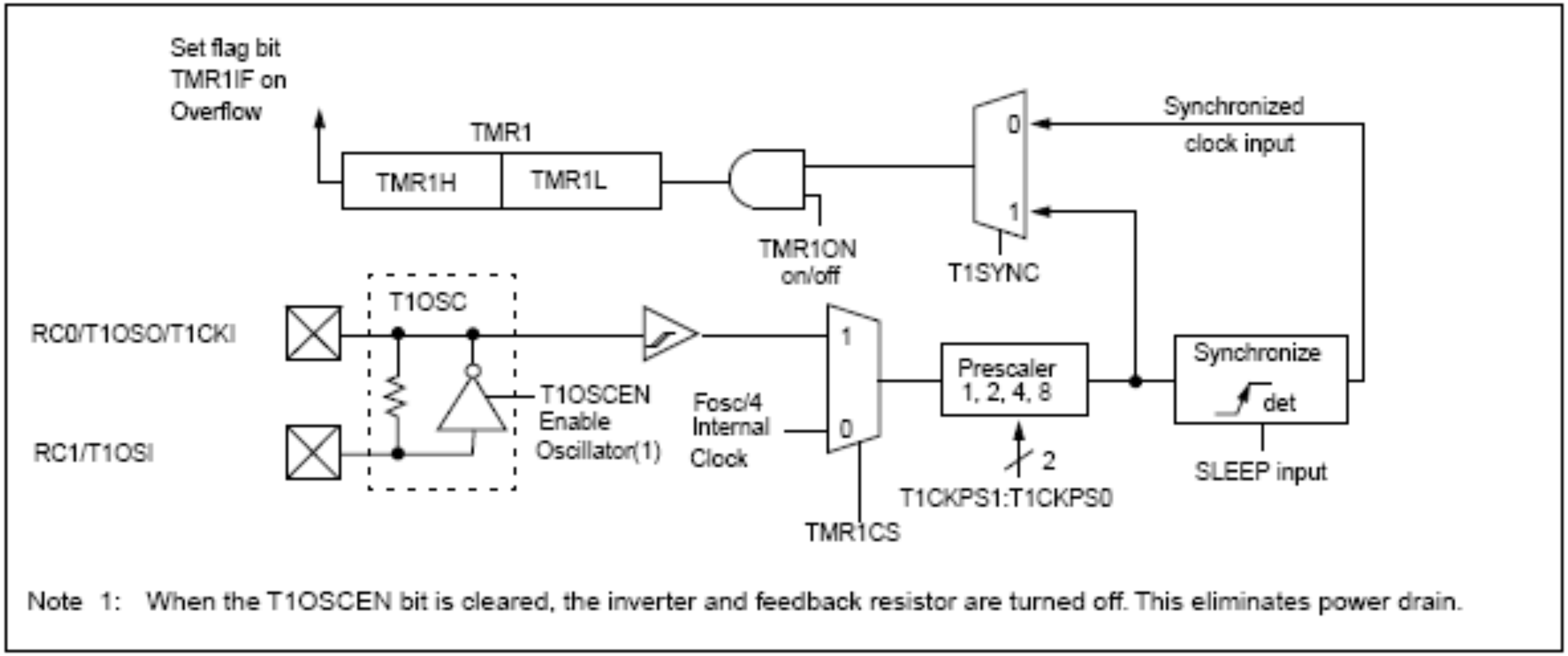
setup_counters(rtcc_internal, rtcc_div16);

zie voorbeeldprogramma bladz. 241

Voor practicumopgave gebruik Timer 1 (16 bits)

Timer 1 (16 bits)

FIGURE 6-3: TIMER1 BLOCK DIAGRAM



Note 1: When the T1OSCEN bit is cleared, the inverter and feedback resistor are turned off. This eliminates power drain.

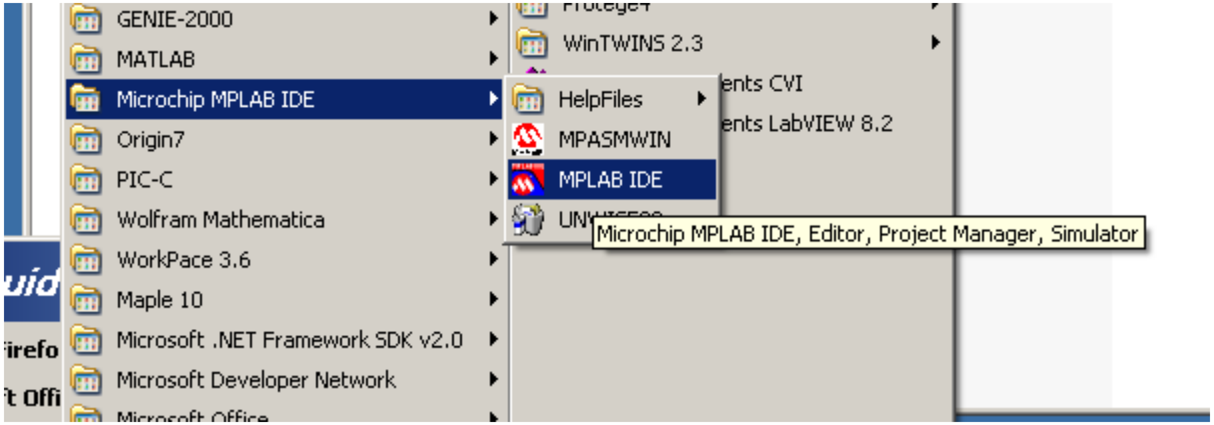
setup_timer_1(*mode*); Zie bladzijde 58 en 59

set_timer_1(value);

i = get_timer_1(); bladz. 43

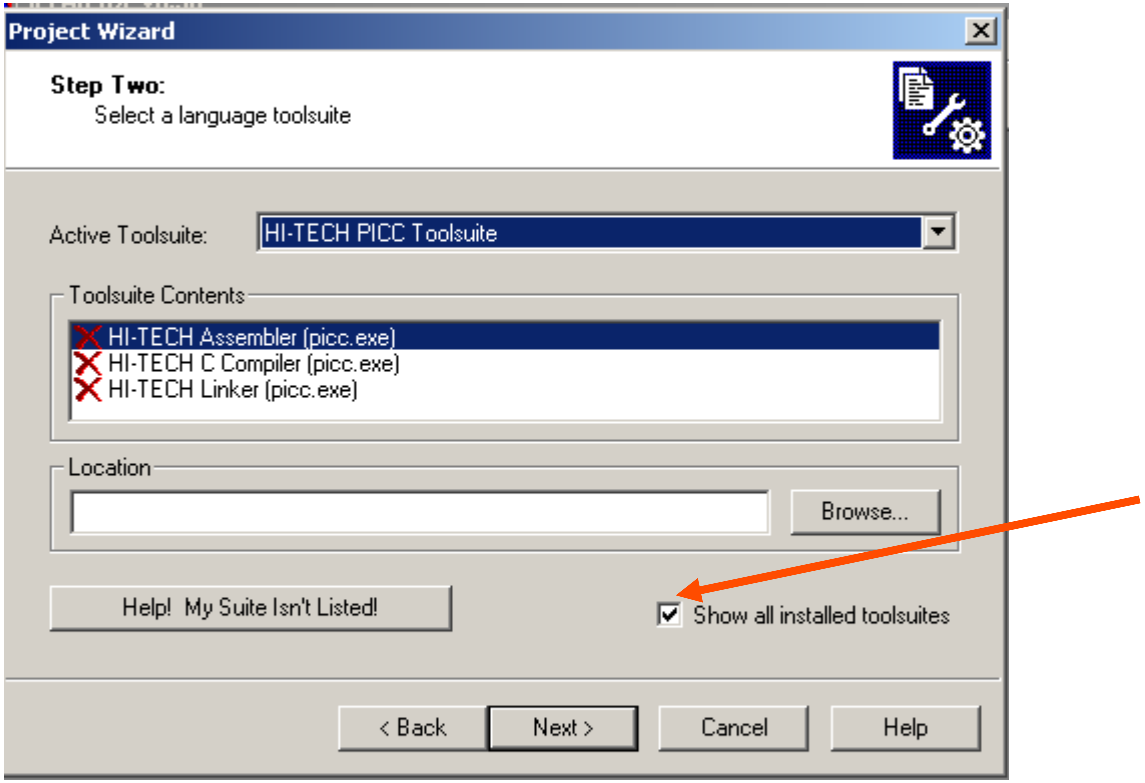
Installatie C-compiler in MPLab omgeving

1. Maak een nieuwe folder “Microcontroller” aan op je N-drive
2. Kopieer de file Gene\\benb\voorStudenten\test876.c naar deze folder
3. Start de applicatie: Microchip MPLAB IDE →MPLAB IDE



Blijf klikken totdat het lukt!

- Kies: Project → Project Wizard →Next →PIC 16F876
- Next



Project Wizard

Step Two:
Select a language toolsuite

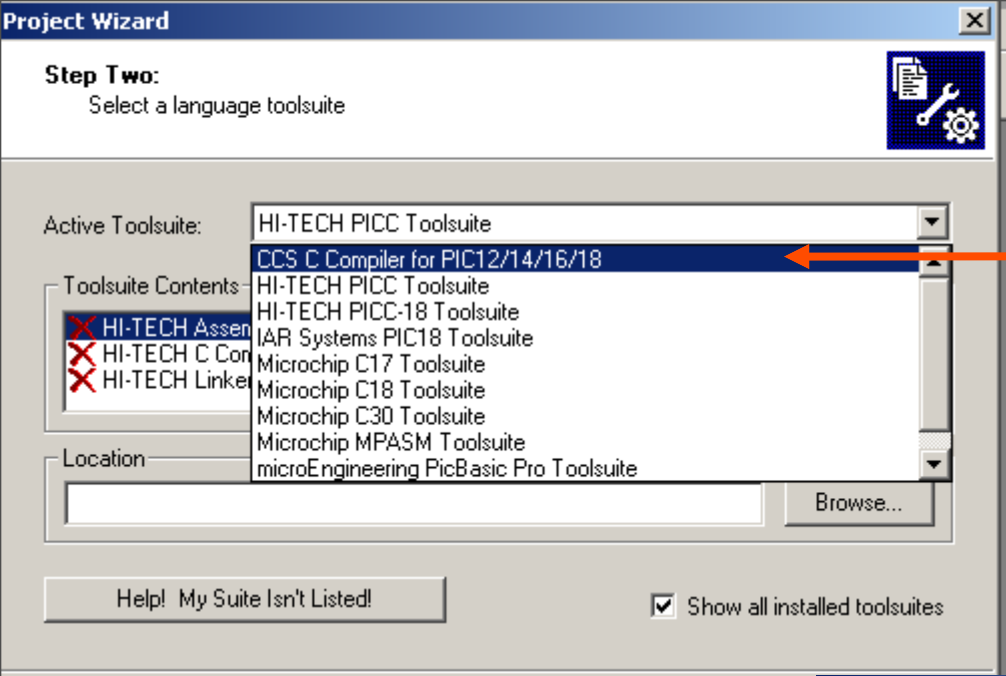
Active Toolsuite: HI-TECH PICC Toolsuite

Toolsuite Contents

- ✗ HI-TECH Assembler (picc.exe)
- ✗ HI-TECH C Compiler (picc.exe)
- ✗ HI-TECH Linker (picc.exe)

Location

☑ Show all installed toolsuites



Project Wizard

Step Two:
Select a language toolsuite

Active Toolsuite: HI-TECH PICC Toolsuite

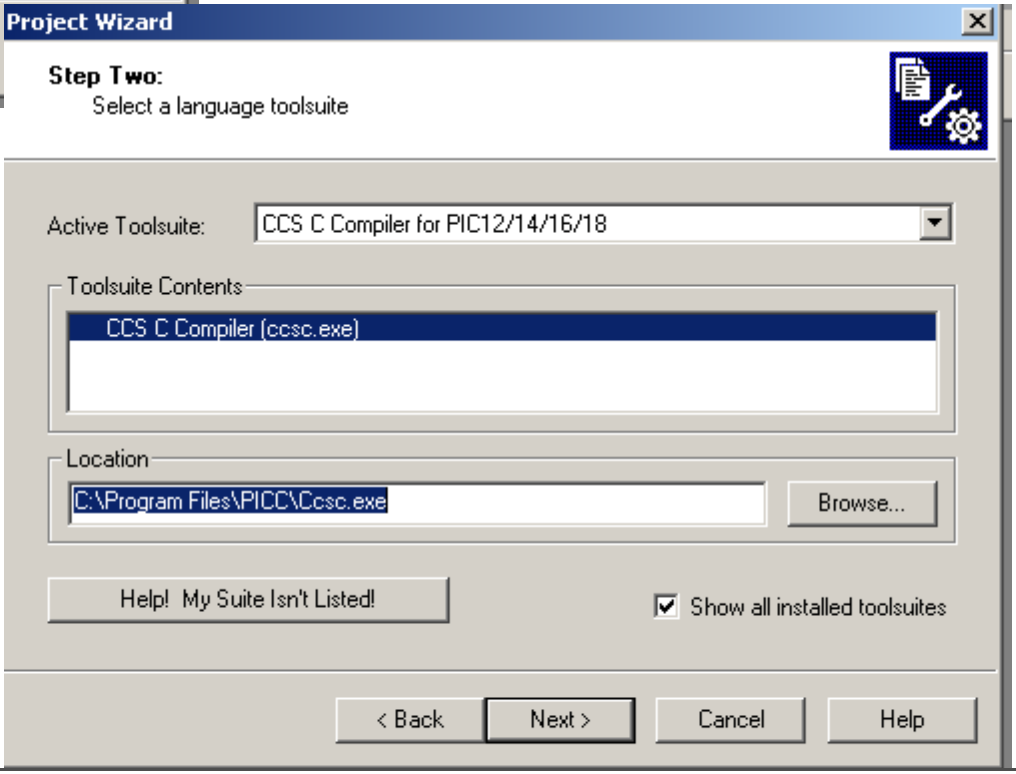
- CCS C Compiler for PIC12/14/16/18
- HI-TECH PICC Toolsuite
- HI-TECH PICC-18 Toolsuite
- IAR Systems PIC18 Toolsuite
- Microchip C17 Toolsuite
- Microchip C18 Toolsuite
- Microchip C30 Toolsuite
- Microchip MPASM Toolsuite
- microEngineering PicBasic Pro Toolsuite

Toolsuite Contents

- HI-TECH Assem
- HI-TECH C Com
- HI-TECH Linker

Location

☑ Show all installed toolsuites



Project Wizard

Step Two:
Select a language toolsuite

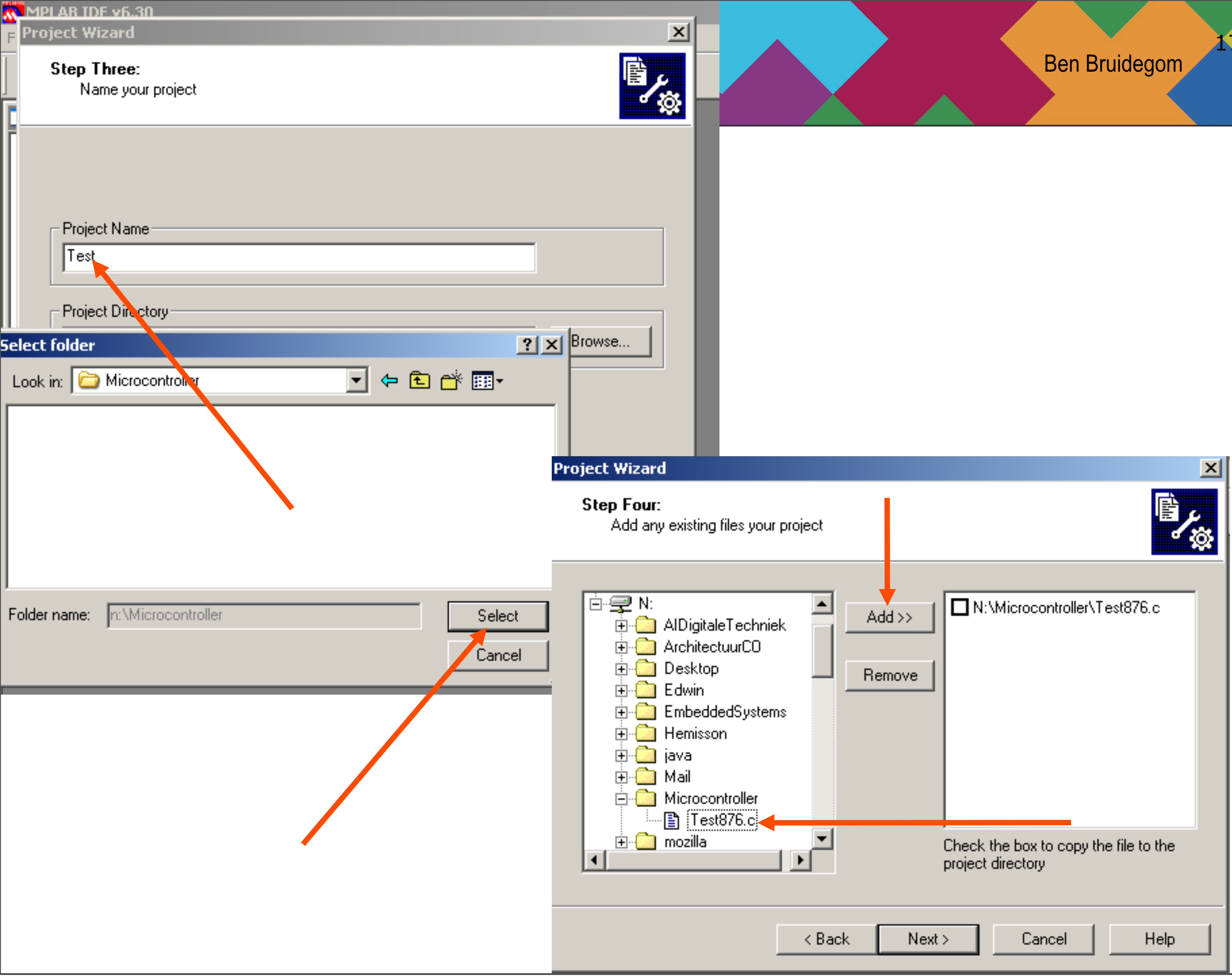
Active Toolsuite: CCS C Compiler for PIC12/14/16/18

Toolsuite Contents

- CCS C Compiler (ccsc.exe)

Location: C:\Program Files\PICC\Ccsc.exe

☑ Show all installed toolsuites



Project Wizard

Step Three:
Name your project

Project Name
Test

Project Directory

Select folder

Look in: Microcontroller

Folder name: n:\Microcontroller

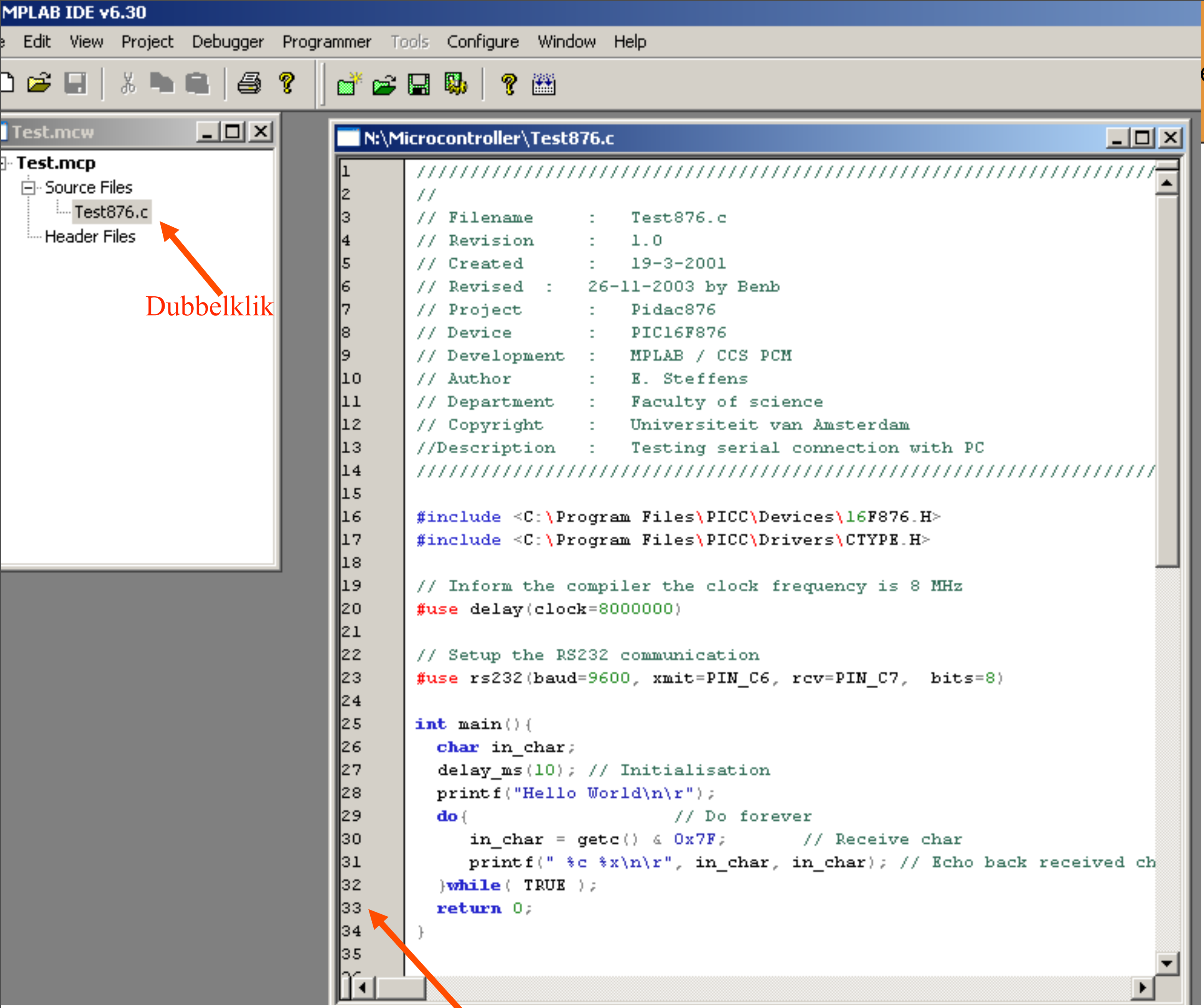
Project Wizard

Step Four:
Add any existing files your project

- N:
  - AIDigitaleTechniek
  - ArchitectuurCO
  - Desktop
  - Edwin
  - EmbeddedSystems
  - Hemisson
  - java
  - Mail
  - Microcontroller
    - Test876.c
  - mozilla

□ N:\Microcontroller\Test876.c

Check the box to copy the file to the project directory

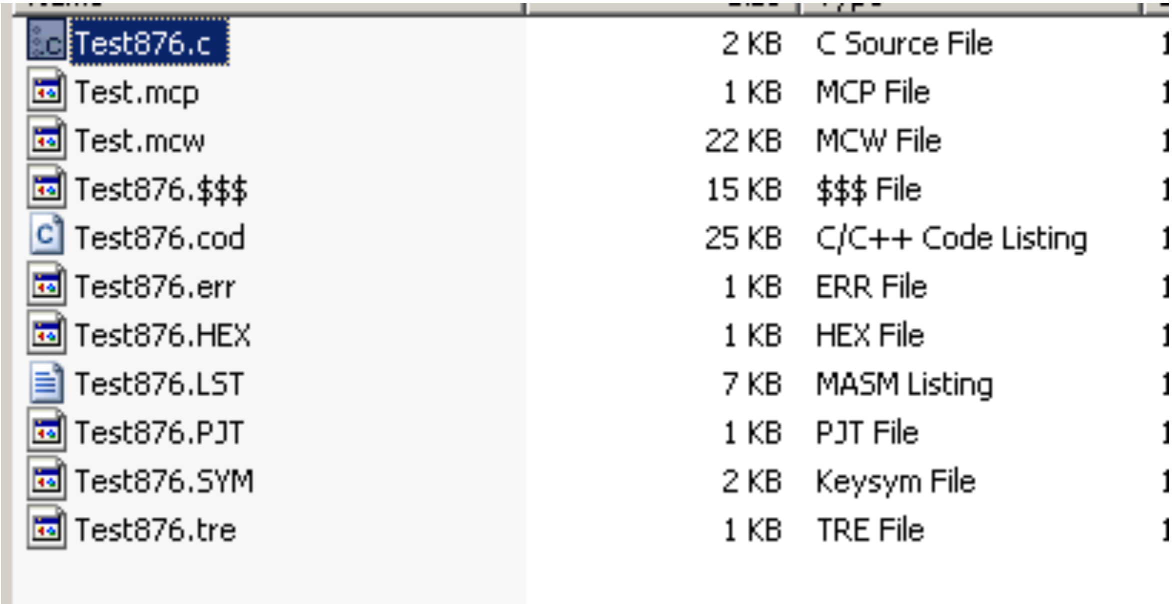

MPLAB IDE v6.30
e Edit View Project Debugger Programmer Tools Configure Window Help
Test.mcw
Test.mcp
Source Files
Test876.c
Header Files
Dubbelklik
N:\Microcontroller\Test876.c

```
/////////////////////////////////////////////////////////////////////
//
// Filename       :   Test876.c
// Revision       :   1.0
// Created        :   19-3-2001
// Revised  :   26-11-2003 by Benb
// Project        :   Pidac876
// Device         :   PIC16F876
// Development    :   MPLAB / CCS PCM
// Author         :   E. Steffens
// Department     :   Faculty of science
// Copyright      :   Universiteit van Amsterdam
//Description     :   Testing serial connection with PC
/////////////////////////////////////////////////////////////////////

#include <C:\Program Files\PICC\Devices\16F876.H>
#include <C:\Program Files\PICC\Drivers\CTYPE.H>

// Inform the compiler the clock frequency is 8 MHz
#use delay(clock=8000000)

// Setup the RS232 communication
#use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7,  bits=8)

int main(){
  char in_char;
  delay_ms(10); // Initialisation
  printf("Hello World\n\r");
  do{                     // Do forever
    in_char = getc() & 0x7F;        // Receive char
    printf(" %c %x\n\r", in_char, in_char); // Echo back received ch
  }while( TRUE );
  return 0;
}
```

Voor Line Numbers: Edit → Properties → Line Numbers

Build All maakt een reeks files aan:

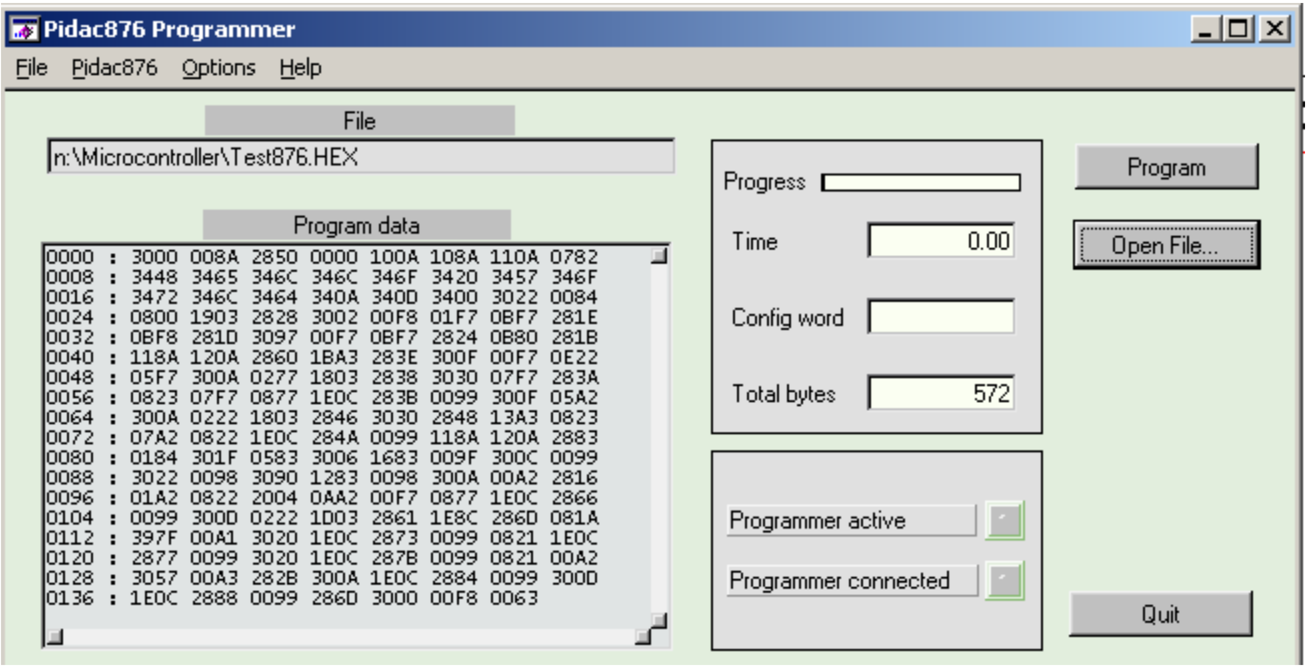
- Compileren etc: Project → Build All
- Ga naar de Folder: Microcontroller
- Bekijk met Notepad de files Test876.LST en Test876.HEX
- De file Test876.HEX moet in de microcontroller worden geladen



Test876.c	2 KB	C Source File	1
Test.mcp	1 KB	MCP File	1
Test.mcw	22 KB	MCW File	1
Test876.$$$	15 KB	$$$ File	1
Test876.cod	25 KB	C/C++ Code Listing	1
Test876.err	1 KB	ERR File	1
Test876.HEX	1 KB	HEX File	1
Test876.LST	7 KB	MASM Listing	1
Test876.PJT	1 KB	PJT File	1
Test876.SYM	2 KB	Keysym File	1
Test876.tre	1 KB	TRE File	1

Uploaden file: Test876.HEX naar Microcontroller

1. Kopieer de folder: Gene\\benb\voorStudenten\PicProg naar je N-drive
2. Start de applicatie PicProg.exe



File: n:\Microcontroller\Test876.HEX

Program data

```
0000 : 3000 008A 2850 0000 100A 108A 110A 0782
0008 : 3448 3465 346C 346C 346F 3420 3457 346F
0016 : 3472 346C 3464 340A 340D 3400 3022 0084
0024 : 0800 1903 2828 3002 00F8 01F7 0BF7 281E
0032 : 0BF8 281D 3097 00F7 0BF7 2824 0B80 281B
0040 : 118A 120A 2860 1BA3 283E 300F 00F7 0E22
0048 : 05F7 300A 0277 1803 2838 3030 07F7 283A
0056 : 0823 07F7 0877 1E0C 283B 0099 300F 05A2
0064 : 300A 0222 1803 2846 3030 2848 13A3 0823
0072 : 07A2 0822 1E0C 284A 0099 118A 120A 2883
0080 : 0184 301F 0583 3006 1683 009F 300C 0099
0088 : 3022 0098 3090 1283 0098 300A 00A2 2816
0096 : 01A2 0822 2004 0AA2 00F7 0877 1E0C 2866
0104 : 0099 300D 0222 1D03 2861 1E8C 286D 081A
0112 : 397F 00A1 3020 1E0C 2873 0099 0821 1E0C
0120 : 2877 0099 3020 1E0C 287B 0099 0821 00A2
0128 : 3057 00A3 282B 300A 1E0C 2884 0099 300D
0136 : 1E0C 2888 0099 286D 3000 00F8 0063
```

Progress:

Time: 0.00

Config word:

Total bytes: 572

Programmer active

Programmer connected

1. Sluit de Microcontroller aan op de PC via de seriële poort
2. Zet de stand van de Microcontroller op PGM en druk op RESET
3. Klik op Program

De applicatie testen

Open de applicatie: Tera Term Pro

1. Vink “Serial” aan
2. Zet de stand van de Microcontroller op uC en druk op RESET.
3. Type enkele karakters