11018 VDI

Introduction to the Visual Device Initializer (MPLAB® IDE VDI)

Class Objectives

When you finish this class you will:

- Understand what VDI *is*
- Have used VDI to initialize a microcontroller
- Know the benefits of using VDI

Agenda

- **Talk/Demo Section**
  - What is MPLAB® IDE VDI?
  - Feature Configuration
  - Conflict Detection & Resolution
  - Initialization-Code Generation
  - Miscellany
- **Hands-On Section**
  - Lab 1 - Configure Oscillator, Port, Timer, and UART
  - Lab 2 - Build a Voltage Meter

What is MPLAB® IDE VDI?

VDI Is…

- **A graphical configuration tool for PIC® MCUs**
- **An initialization-code generator**
  - Selectable optimization/robustness
  - Callable from either asm or C
- **A project-report generator**
- **A way to access to Datasheets & Errata**
- **Integrated with MPLAB® IDE**
  - MPLAB IDE Plug-in

Feature Configuration

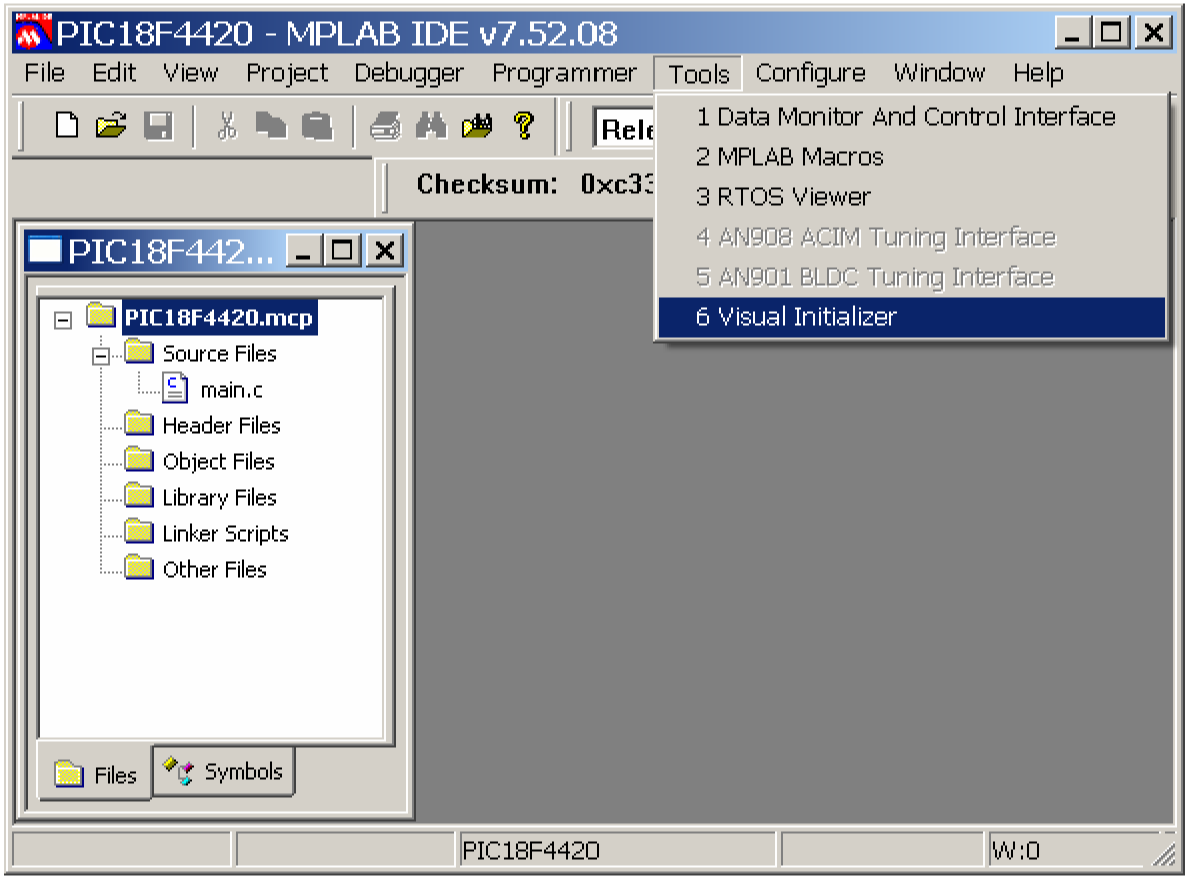
What is a VDI "Feature"?

- **We configure a device by adding, removing, and configuring its features**
- **A "feature" represents a device capability (functional, not physical)**
- **Different VDI features may share the same silicon**
  - Counter 1, Timer 1 and RTC
- **Represented as icons on the palette and on the chip outline**

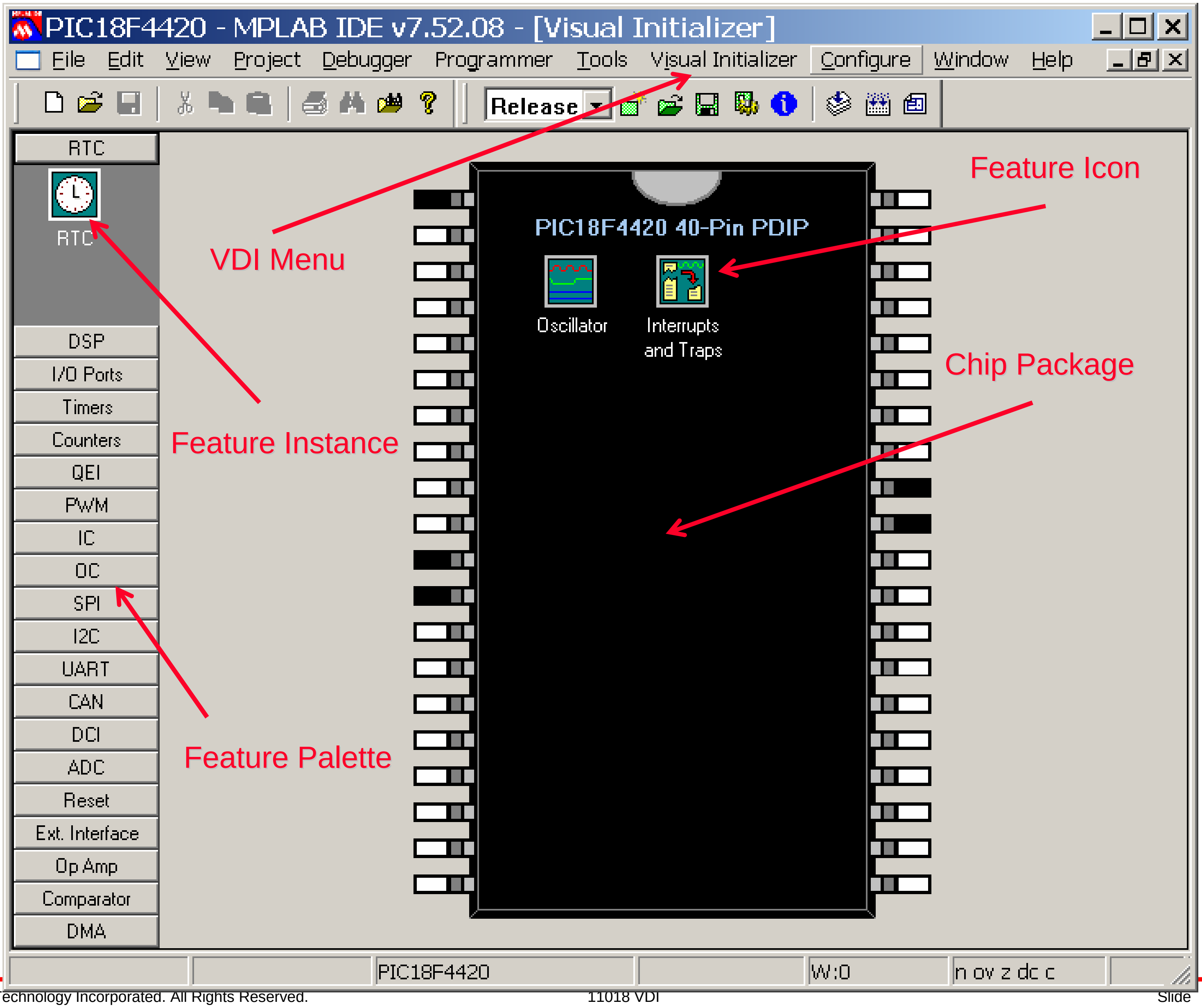
How to Start VDI

- Start MPLAB® IDE if it is not yet running
- Select the "Tools" menu
- Select "Visual Initializer"
- The "Visual Initializer" window pops up
  - Chip outline of selected part and package
  - Palette of available features
- The "Visual Initializer" menu appears with the following menu items:
  - Data Sheet and Errata
  - Report Generation
  - Code Generation
  - (Others too)

How to Start VDI



Hello VDI!



VDI Menu

Feature Instance

Feature Palette

Feature Icon

Chip Package

Why is Stuff Already on Chip?

- **Oscillators and Interrupts**
  - Can’t live without them
- **Oscillators**
  - No heartbeat - no life
- **Interrupts**
  - One-stop-shop for all interrupts
    - **Both used and unused**
  - Can provision interrupts from dialog

How to Add a Feature

- **Click palette “drawer” for the type of feature**
  - I/O Ports, ADC, Timer, etc.
- **Scroll palette to the feature you want**
  - Port A, Timer 3, CAN 2, etc.
- **Click it and drag it onto the chip outline**
- **Drop it!**
  - Feature icons self-organize
  - Scroll-bar appears when chip outline overfull

How to Remove a Feature

- **Click the feature icon on the chip that you want to remove**
  - Port A, Timer 3, CAN 2, etc.
- **Drag it out of the chip outline**
- **Drop it**
  - Feature icon will disappear

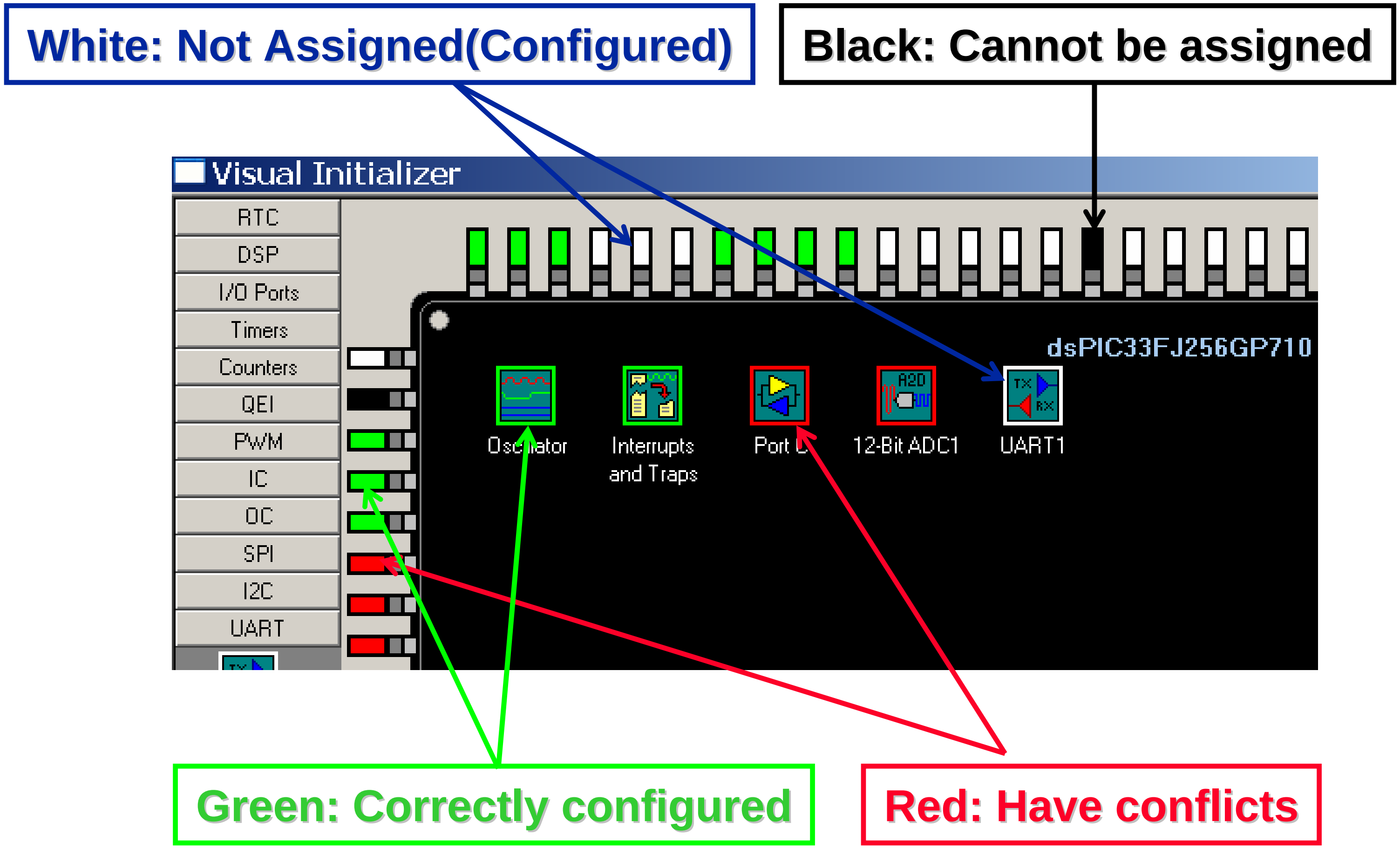
What Can I Do with a Feature?

- **Right-click on a feature icon on chip**
  - Brings up a context menu of possible actions
- **Configure**
  - Same as clicking a gray or green feature icon
- **Error**
  - Show conflict information
- **Zoom**
  - Shows connections between features and pins
- **Help**

VDI Color Code

- **Black**
  - Pin which cannot be assigned (V_{DD}, V_{SS})
- **White**
  - Pin/feature which has not been assigned/configured
- **Green**
  - Pin or feature correctly assigned/configured
- **Red**
  - Pin/feature conflicts* with another pin/feature

VDI Color Code



White: Not Assigned(Configured)

Black: Cannot be assigned

Green: Correctly configured

Red: Have conflicts

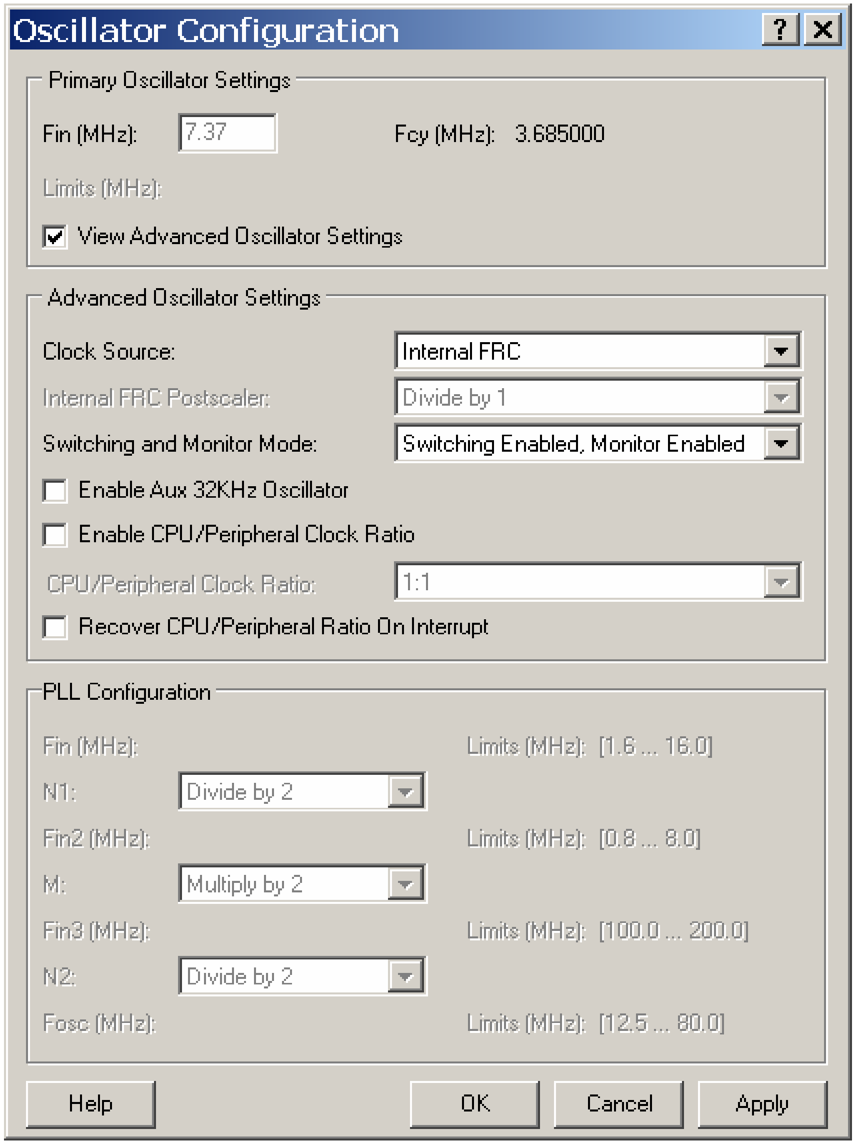
How to Configure a Feature

- Click on the icon of the feature that you want to configure
- The feature-configuration dialog will pop up
- Select/type-in the desired parameters in the configuration dialog
- Click Apply button to set those parameters

How to Configure the Oscillator

- **Select the clock source**
- **Specify Fin (Fosc, if internal is not used)**
- **Specify desired PLL factors if PLL mode is used**
- **Specify other desired parameters**
- **Oscillator frequency used to calculate**
  - Baud rate
  - Timer periods
  - Etc.

How to Configure the Oscillator



Oscillator Configuration

Primary Oscillator Settings

Fin (MHz): 7.37

Fcy (MHz): 3.685000

Limits (MHz):

☑ View Advanced Oscillator Settings

Advanced Oscillator Settings

Clock Source: Internal FRC

Internal FRC Postscaler: Divide by 1

Switching and Monitor Mode: Switching Enabled, Monitor Enabled

☐ Enable Aux 32KHz Oscillator

☐ Enable CPU/Peripheral Clock Ratio

CPU/Peripheral Clock Ratio: 1:1

☐ Recover CPU/Peripheral Ratio On Interrupt

PLL Configuration

Fin (MHz): Limits (MHz): [1.6 ... 16.0]

N1: Divide by 2

Fin2 (MHz): Limits (MHz): [0.8 ... 8.0]

M: Multiply by 2

Fin3 (MHz): Limits (MHz): [100.0 ... 200.0]

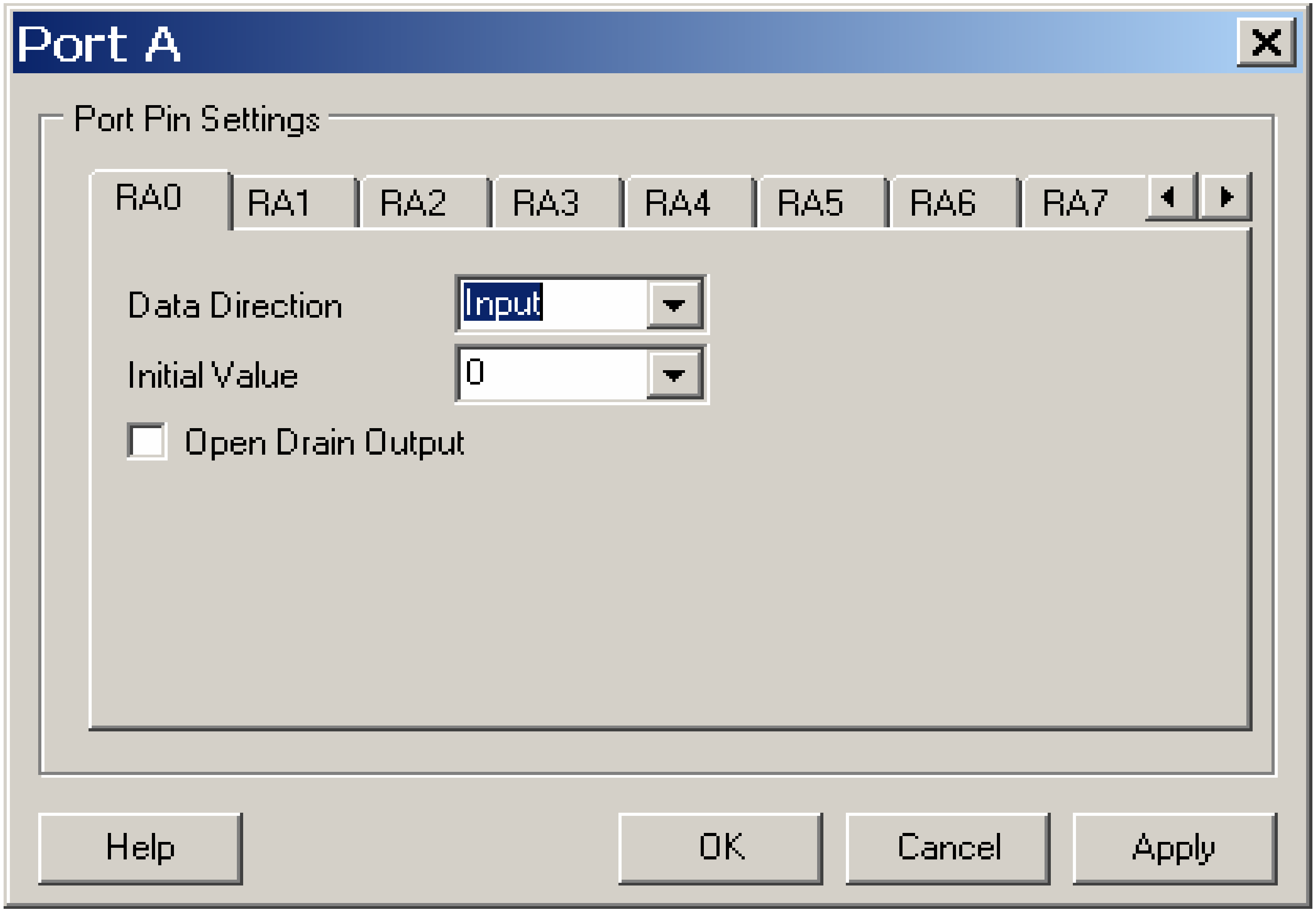
N2: Divide by 2

Fosc (MHz): Limits (MHz): [12.5 ... 80.0]

How to Configure a Port

- **Select I/O ports from the palette**
- **Pick a port, drag it onto the chip outline**
- **Click the feature icon to configure it**
- **For each bit of the port specify:**
  - Input or output
  - If output, initial value
- **Currently there is no way to group port bits**

How to Configure a Port



Port A

Port Pin Settings

Data Direction: Input

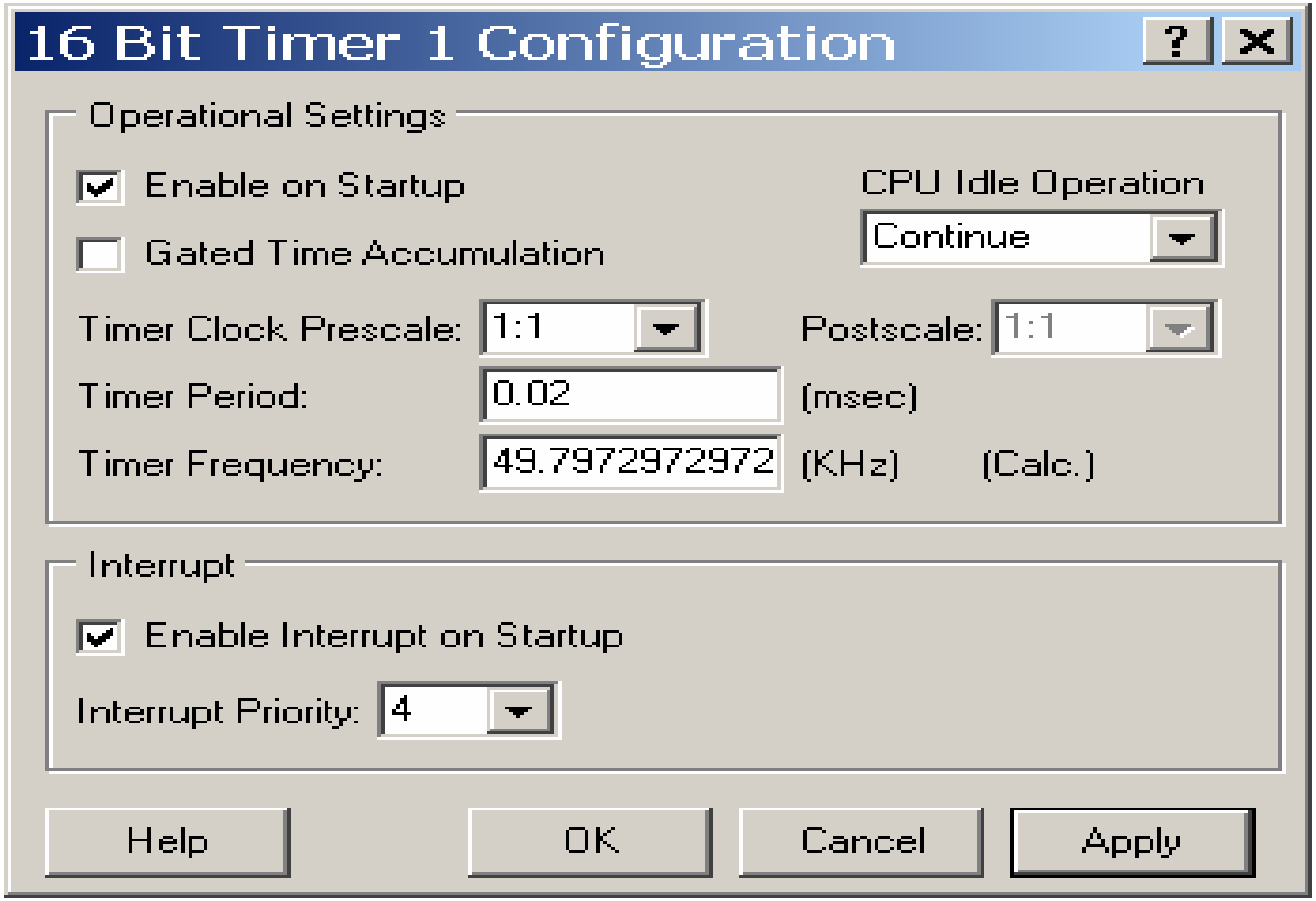
Initial Value: 0

□ Open Drain Output

How to Configure a Timer

- **Select timers from the palette**
- **Pick a timer, drag it onto the chip outline**
  - TIP: Some timers have special functions
- **Click the feature icon to configure it**
- **Timer period in milliseconds**
  - Not register values
  - Derived from oscillator settings
- **Operation enable on startup**
- **Interrupt enable on startup**

How to Configure a Timer



16 Bit Timer 1 Configuration

Operational Settings

☑ Enable on Startup

□ Gated Time Accumulation

CPU Idle Operation: Continue

Timer Clock Prescale: 1:1

Postscale: 1:1

Timer Period: 0.02 (msec)

Timer Frequency: 49.7972972972 (KHz) (Calc.)

Interrupt

☑ Enable Interrupt on Startup

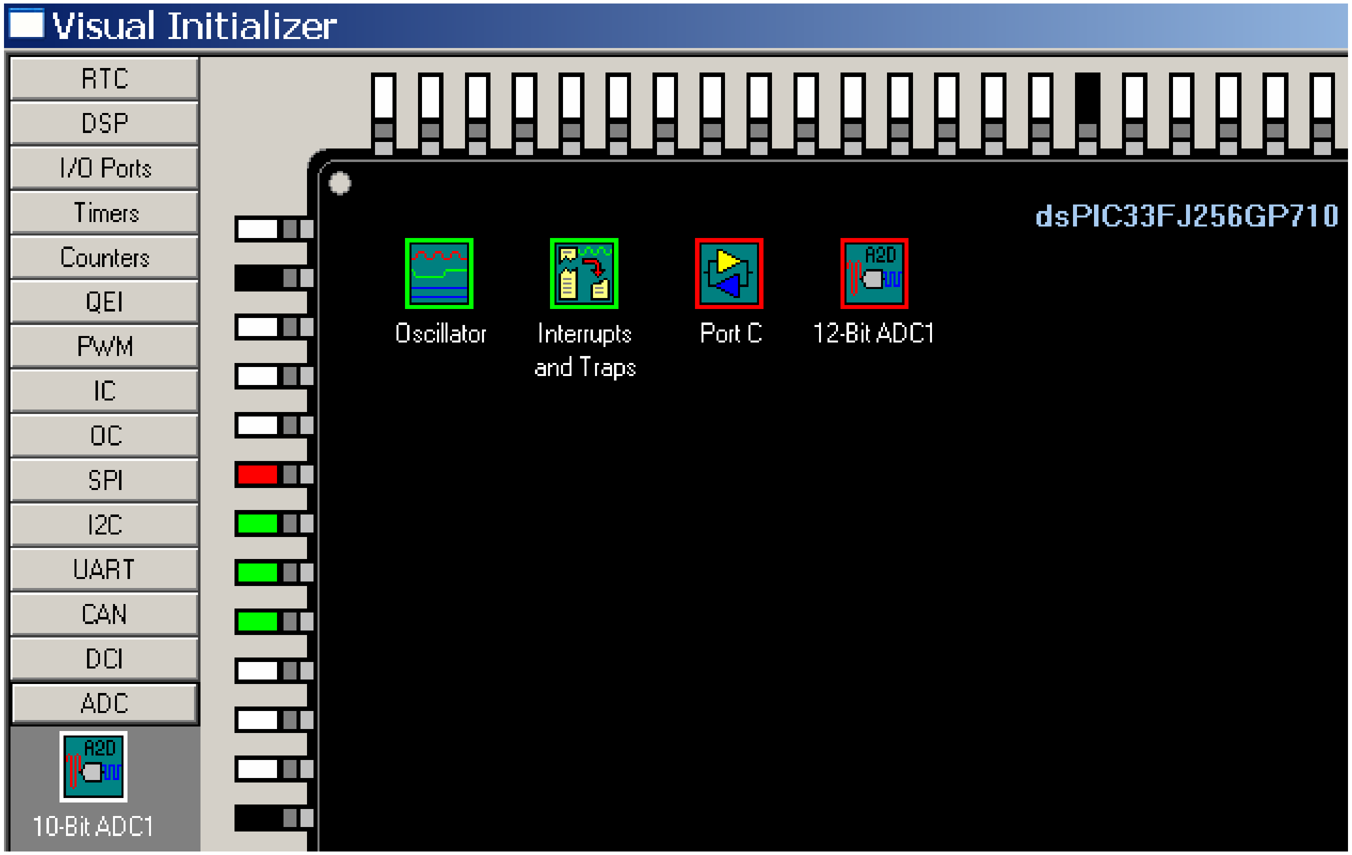
Interrupt Priority: 4

Conflict Detection & Resolution

A Conflict Example

- Claim A2D input pin 16
- Claim Port C pin RC1

A Conflict Example



Oh No! There are Conflicts!

- **Relax, breathe deeply**
- **What is a "Resource"?**
  - An interrupt, register, or pin
- **What is a "Resource Conflict"?**
  - Two or more conflicting demands on the same resource

Conflict-Resolution Tips

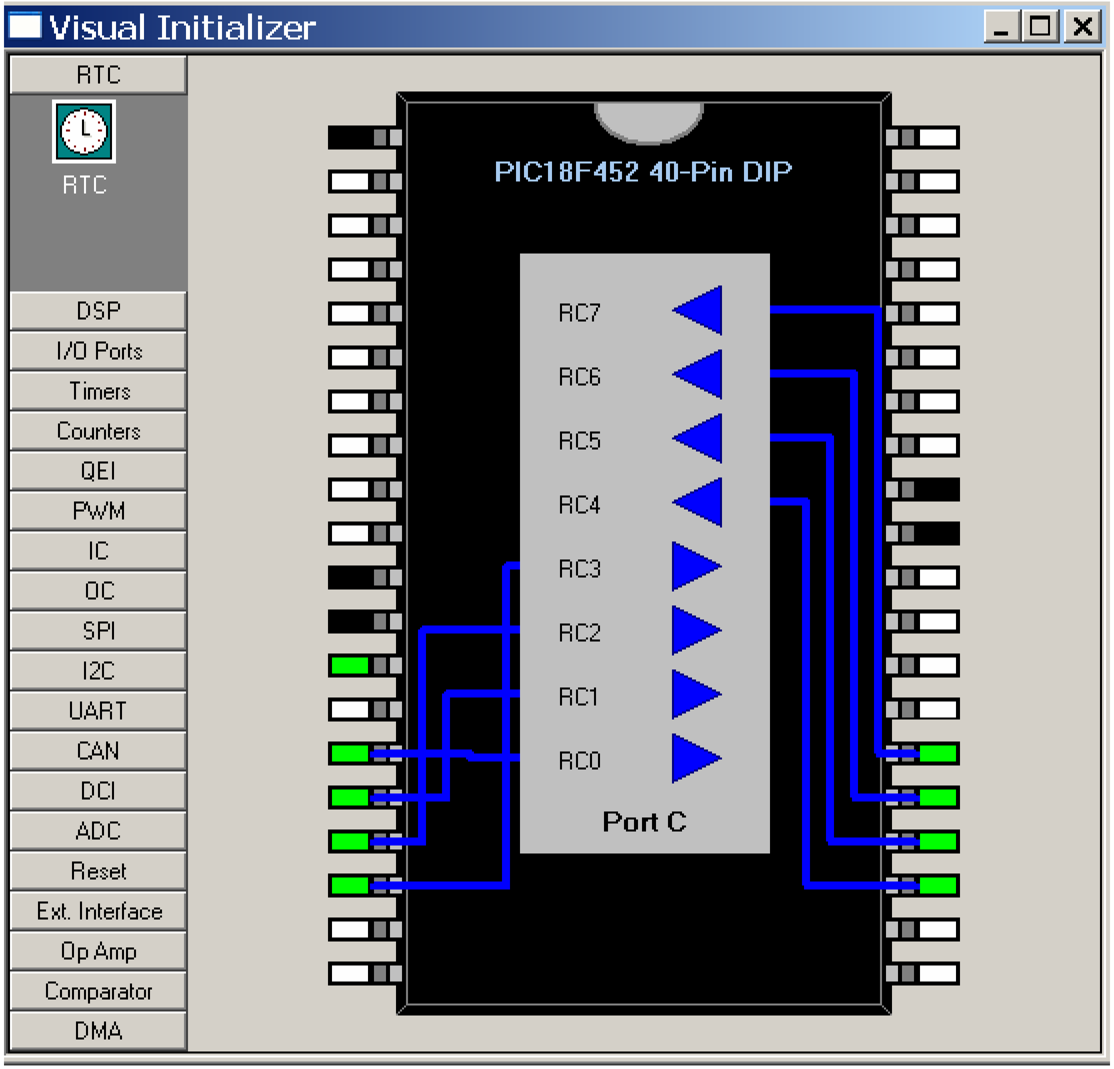
- **Find the conflicted resource**
  - Gather information from MPLAB® IDE output window
- **Is the conflict really a problem?**
- **VDI doesn’t force a solution**
  - VDI’s conflict checking is “static”
  - VDI doesn’t lock you out of conflicting configuration choices
- **Configure less-flexible features first**
- **Apply configuration changes incrementally**

Feature Menu Revisited

- Right-click on a feature icon on chip
  - Brings up a context menu of possible actions
- Configure
  - Same as clicking a gray or green feature icon
- Error
  - Show conflict information
- Zoom
  - Shows connections between features and pins
- Help

Zoom

Shows connections between features and pins



Initialization-Code Generation

How to Generate Code

- **Select the “Visual Initializer” menu**
- **Select “Code Generation Options”**
  - Choose options that fit your application
- **Select “Code Generation”**
- **Click the “Save to Project” button**

The Generated Code...

- **Initialization code for selected features**
  - Assembly code for optimal performance
  - Callable from C and Assembly
- **Files automatically added to project**
  - C-callable assembly source code
  - Header file with function prototype

...Calling It From C

- `#include` the .h file
- Call `VisualInitialization();`
- Your PIC® MCU is in the requested state

...Calling It From Assembly

- `INCLUDE` the .inc file
- `CALL _VisualInitialization`
- Your PIC® MCU is in the requested state

What’s Left For You to Write?

- **Your application code!** ***;-)***
- **ISRs (Interrupt Service Routines)**
  - See PIC® MCU Programming Guide
  - See C18, C30 User’s Guide

Building & Running the Demo

- Add application-specific code
- Build the project
- Debug the project

Miscellany

Saving the VDI Configuration

- **VDI configuration is saved in the Workspace**
  - Select the File menu
  - Select the “Save Workspace” menu item

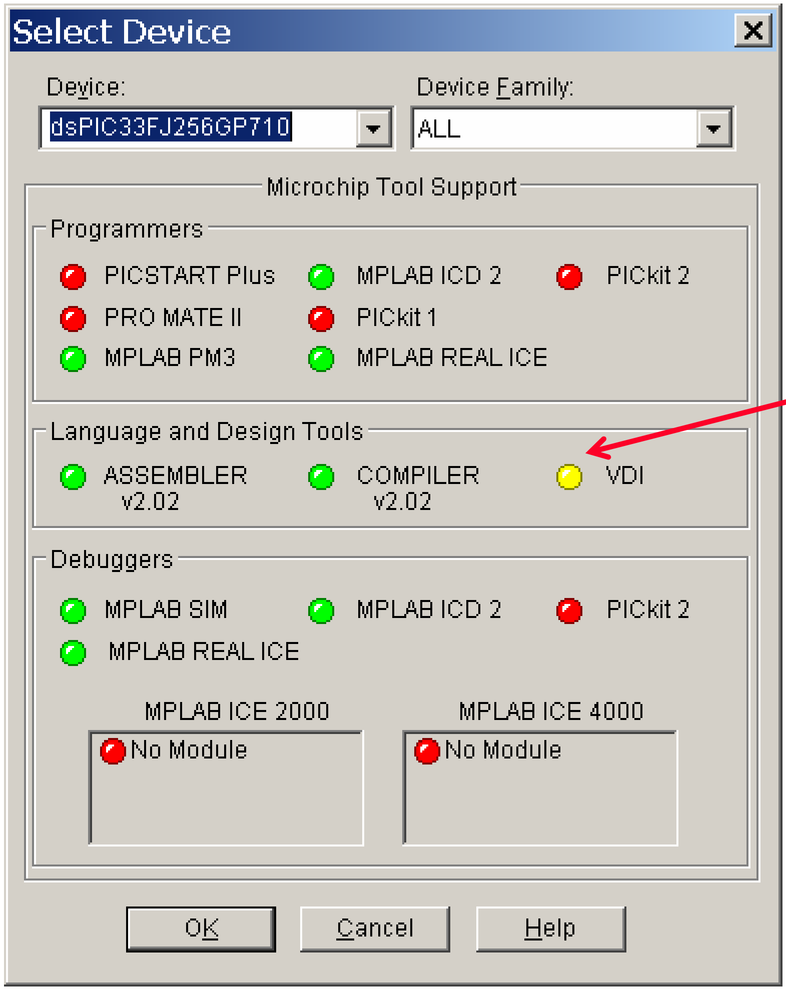
VDI Part Support

- **dsPIC® Digital Signal Controller Family**
  - 29 dsPIC30F devices
  - 19 dsPIC33F devices
- **31 PIC24F Devices**
- **46 PIC18 devices**
- **PIC16F785**

Checking If A Part Is Supported

- **Go to the Configure menu**
- **Select Device menu item**
- **Device dialog box will pop up**
- **Pick the device the you want to check from device combo box**
- **Check the VDI GYR (green, yellow, red) light**
  - Green: Full support
  - Yellow: Beta support
  - Red: No support

Select Device Dialog



Select Device

Device: dsPIC33FJ256GP710

Device Family: ALL

Microchip Tool Support

Programmers

🔴 PICSTART Plus	🟢 MPLAB ICD 2	🔴 PICkit 2
🔴 PRO MATE II	🔴 PICkit 1	
🟢 MPLAB PM3	🟢 MPLAB REAL ICE	

Language and Design Tools

🟢 ASSEMBLER v2.02	🟢 COMPILER v2.02	🟡 VDI

Debuggers

🟢 MPLAB SIM	🟢 MPLAB ICD 2	🔴 PICkit 2
🟢 MPLAB REAL ICE		

MPLAB ICE 2000: 🔴 No Module

MPLAB ICE 4000: 🔴 No Module

VDI GYR Light

What’s Next in VDI Development?

- **Any thing that will help you!**
- **Configurable Library in VDI**
  - Configurable Library is a collection of configurable library modules
  - The library module implements firmware functionality
  - It has parameters that you can configure
- **More devices support**
  - All dsPIC33F devices are expected to be supported by the end of this year
  - All PIC24HJ devices are expected to be supported by the end of this year

Benefits of Using VDI

- Interactively evaluate a PIC® MCU for fit
- Configure the whole device for your job
- Generate code for feature initialization
- Maintain device setup for later modification
- Access datasheets and errata
- Generate project reports

Where Can I Find VDI?

- **On the MPLAB® IDE CD-ROM**
- **On the Microchip Web Site**
  - Under Development Tools, Software
  - http://ww1.microchip.com/downloads/en/DeviceDoc/vdisetup.zip
- **From your local Microchip Sales Office**

Questions?

Lab 1 – Use VDI to Configure Osc, Port, Timer, and UART

- **Use dsPIC33FJ256GP710 MPLAB® IDE SIM**
- **Functionalities**
  - Increase the value of LATA by 1 every second
  - Send the value of LATA to UART
  - Enable/Disable Sending the value of LATA to UART by sending E/D to UART.
- **Features to be configured**
  - Oscillator
  - Port A
  - Timer 1
  - UART 1
- **See handout for detailed information**

Lab 2 – Build a Voltage Meter
Using dsPIC33FJ256GP710

- **Functionalities**
- **Use Explorer 16 Development Board (DM240001)**
  - Convert analog voltage input at RA5
  - Configure RTC
  - Display the voltage on the LCD
  - Display the time (on RTC) on the LCD
  - Send the value of the voltage to UART2
- **Features to be configured**
  - Oscillator
  - Port A
  - RTC
  - 12-bits ADC
- **See handout for detailed information**

References

- dsPIC33F Family Data Sheet
- Explorer 16 Development Board User’s Guide
- C30 Compiler User’s Guide
- C18 Compiler User’s Guide

Trademarks

The Microchip name and logo, the Microchip logo, Accuron, dsPIC, KeeLoq, KeeLoq logo, microID, MPLAB, PIC, PICmicro, PICSTART, PRO MATE, rfPIC and SmartShunt are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AmpLab, FilterLab, Linear Active Thermistor, Migratable Memory, MXDEV, MXLAB, SEEVAL, SmartSensor and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, ECAN, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, PICkit, PICDEM, PICDEM.net, PICLAB, PICtail, PowerCal, PowerInfo, PowerMate, PowerTool, REAL ICE, rfLAB, Select Mode, Smart Serial, SmartTel, Total Endurance, UNI/O, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.