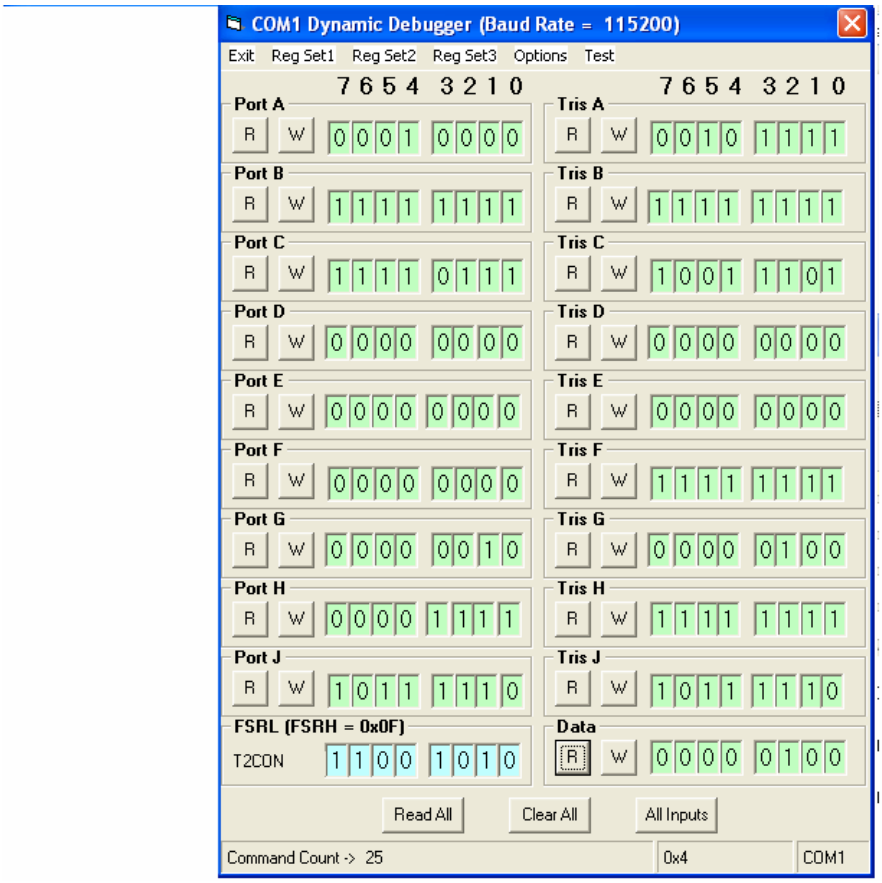
Innovation First, Inc.

IFI_Loader

and the

Dynamic Debug Tool



Introduction to the Dynamic Debug Tool

Beginning with version 1.0.8, the IFI Loader application has a new feature called the Dynamic Debug Tool (DDT). Besides its primary use of downloading code to your Innovation First Robot Controller and its secondary use of displaying textual feedback from the Robot Controller in the Terminal window, IFI Loader can now be used to directly read and modify memory locations and registers of the User processor. When used concurrently with the Microchip PIC18F8520 Data Sheet and optionally with an oscilloscope, the DDT can be a powerful tool for learning the microcontroller's architecture, for writing your custom code, and for debugging purposes.

If you just want to try out the DDT without modifying any code, you can simply download the Edu_DDT_default.hex file into your Mini-RC or the FrcCode_default_DDT.hex into your Full-size RC. These .hex files are included in the DDT code zip file which is available for download from the Innovation First web site. Once this is done, you can go directly to the instructions for connecting to and using the DDT.

Please keep in mind that Innovation First cannot provide technical support for the Dynamic Debug Tool or for the Robot Controller code (DDT Code) released to interface with it. They are simply offered as an example of the possibilities available to you with the powerful Innovation First Robot Controllers. The DDT Code is a stripped down version of the User Default Code and should not be used in Competition or as a replacement for the User Default Code. It should only be used as a tool to learn about the micro-processor architecture and how to write code to interface with it.

Connecting to the Dynamic Debug Tool

1. Connect your Robot Controller's PROGRAM port to a PC serial port.
2. Start IFI_Loader (version 1.0.8 or higher) on your PC.
3. Select the appropriate serial port from the PortSettings menu.
4. Download the FrcCode_default_DDT.hex code which includes the serial driver to your Robot Controller.
5. Close the Terminal Window. In the Options menu of the IFI Loader, select Dynamic Debug Tool. The DDT will open in a new window.
6. Click the "Read All" button at the bottom of the DDT window.
    a. If most spaces turn from white to green, you have successfully connected.
    b. If most spaces remain white and it says "Read Timeout" in the lower left corner, you failed.
        i. Check that your serial cable is connected to the correct port.
        ii. Check that you selected the correct COM port for IFI_Loader.
        iii. Check that you downloaded the correct code to your Robot Controller.

NOTE: Printf functions will not work while using the DDT from port1 on the RC. Printf functions are directed to the TTL Serial port 2. If you want to connect your PC to the TTL port for either printf functions or for connecting to the Dynamic Debug Tool, you will need to use an RS-232 to TTL level shifter to convert the signals to the appropriate voltages.

Using the Dynamic Debug Tool

Here are some examples of what you can do with your Robot Controller using the DDT:

1. Change an input pin to an output, or vice versa.
2. Change an output pin from high to low.
3. Read an input pin to tell whether it is high or low.
4. Configure and start a timer.
5. Configure and start a PWM signal.
6. Enable or disable any interrupt.

Here are brief instructions on using the DDT:

1. Click an "R" button to read the current state of a register. The register will turn green if not already green.
2. Click on a bit to toggle it from 0 to 1 or vice versa. The bit will turn yellow.
3. Click the "W" button next to the register you modified to send the change to the microcontroller. The register will turn red.
4. Click the "R" button to read the register again. It should be in the same state that you wrote, unless you tried writing to a port configured as an input or a protected register.
5. Right-click on a "W" button to make every bit in a register 1. Left-click "W" to send the change.
7. Right-click on a "R" button to make every bit in a register 0. Left-click "W" to send the change.
8. Whenever you change a register, the value of that register (in hexadecimal) is displayed at the bottom of the window as 0x**.

We will discuss only a few screens to the DDT that can be used. They are selectable from the Test menu. When you first start the DDT you will be on the Port Config screen. This enables you to configure the I/O of the microcontroller, as well as read inputs and write outputs. To change a pin to an input you must set its Tris bit to 1. Then the corresponding Port bit, when read, will represent the input level of that pin. To set the pin as an output you must set its Tris bit to 0. Now changing the Port bit will change the output level of the pin.

The FSRL register near the bottom of the DDT window can be used to address any register you like. You can either look up the address of a register on pages 50-51 of the PIC18Fxx20 data sheet or you can select a register from the Reg Set menus at the top of the DDT. For example, referring to page 50 of the data sheet, you can see that the TRISC register has an address of 0x0F94. If you set FSRL to equal 0x94 (since FSRH is already set to 0x0F) and click the "R" button, you will see that the Data displays the same value as the Tris C register above.

You use a similar method on the Memory Locations screen selected from the Test menu. The third screen available from the Test menu, EE PROM allows you to read and write to the first 18 memory locations in the EEPROM of the microcontroller. Any values you store in the EEPROM will be retained even after power is turned off. You can also control PWM Outputs.

Example Using the Dynamic Debug Tool

There are many uses you might find for the Dynamic Debug Tool, and we will not try to list them here. Rather, we will show you one example to get you started.

1. Download the EDU_default_DDT.hex code into a Mini Robot Controller (or the FrcCode_default_DDT.hex code into a Full-size Robot Controller).
2. Monitor the PWM OUT 1 pin of the Mini RC (or the PWM 13 pin of the Full-size RC) on an oscilloscope.
3. Bring up the DDT window.
4. Select CCP2CON from the Reg Set1 menu.
5. Write 0x3C to the register. Read to verify, if not the same re-write.
6. Select PR2 from the Reg Set1 menu. Write 0xF9 to it. Read to verify.
6. Select CCPR2L from the Reg Set1 menu. Write 0x7F to it. Read to verify.
7. Select T2CON from the Reg Set1 menu. Write a 0x04 to it. Read to verify.
8. You should now see a 40kHz square wave on the oscilloscope.

Using Section 16 of the data sheet, try to understand what those registers are for, and play around with writing different values to them and see what happens.

To see how the microcontroller pins are mapped to the Robot Controller I/O, you must refer to the ifi_aliases.h file in your project.