YDRIVE-IT-RZT1 RZ/T1 Solution Kit – Getting Started Guide

Description: This guide will help in connecting the YDRIVE-IT-RZT1 Solution Kit with Motor, Encoder, Power Supply and Host Communication Cables. It will walk you through a simple dual core example, an EtherCAT sample program, the installation of the RZ/T1 Motion Utility, connection to the RZ/T1 Motion Controller and spinning the motor. Finally, this guide provides instructions on how to change the motion parameters and display graphs representing the motion accuracy.

Table of Contents

Introduction: Here we give some general remarks on the YDRIVE-IT-RZT1 solution kit that should help you to go through the examples shown here as smoothly as possible.

- The example programs and the IAR tool chain version in this kit have been checked to work properly with each other. You may find newer versions of tool chain as well as example programs on the Internet, however we recommend not to mix software components from this package with newer versions.
- If the kit undergoes a software update, we will update the complete package and you should not mix software components from different package versions.
- The solution kit CD provides you a snapshot of software and documents that are up to date at the time of release of the kit. Please check our website for additional software and documents.

1 Hardware Setup – Connect RZ/T1 Motion Controller

The first step of getting started with the RZ/T1 Solution Kit is connecting the power supply, the motor, the encoder and the communication cable. Follow the steps using the cables and the motor included with the kit. Not all of the connections are required for all experiments.

1.1 Connecting the Logic and Motor Power

The Logic Power should be between 12V and 24V DC. The power consumption when the controller is idle is around 0.18A but can go up to 0.30A depending on the connected encoders, sensors and other loads.

The motor that is delivered with the kit is prepared for 24V operation. Power is supplied via 5-pin connector J2 on the inverter board as shown in *Figure 1-1*. The 5-pin plug for J2 is included in the kit. We recommend to use separate power supplies for logic and motor.

The Motor Power should not exceed 48V and the load dependent current consumption must not go beyond 8A. The inverter will shut off the power stage when the current exceeds the limits. The MOSFETs will withstand even a short between any two motor windings.

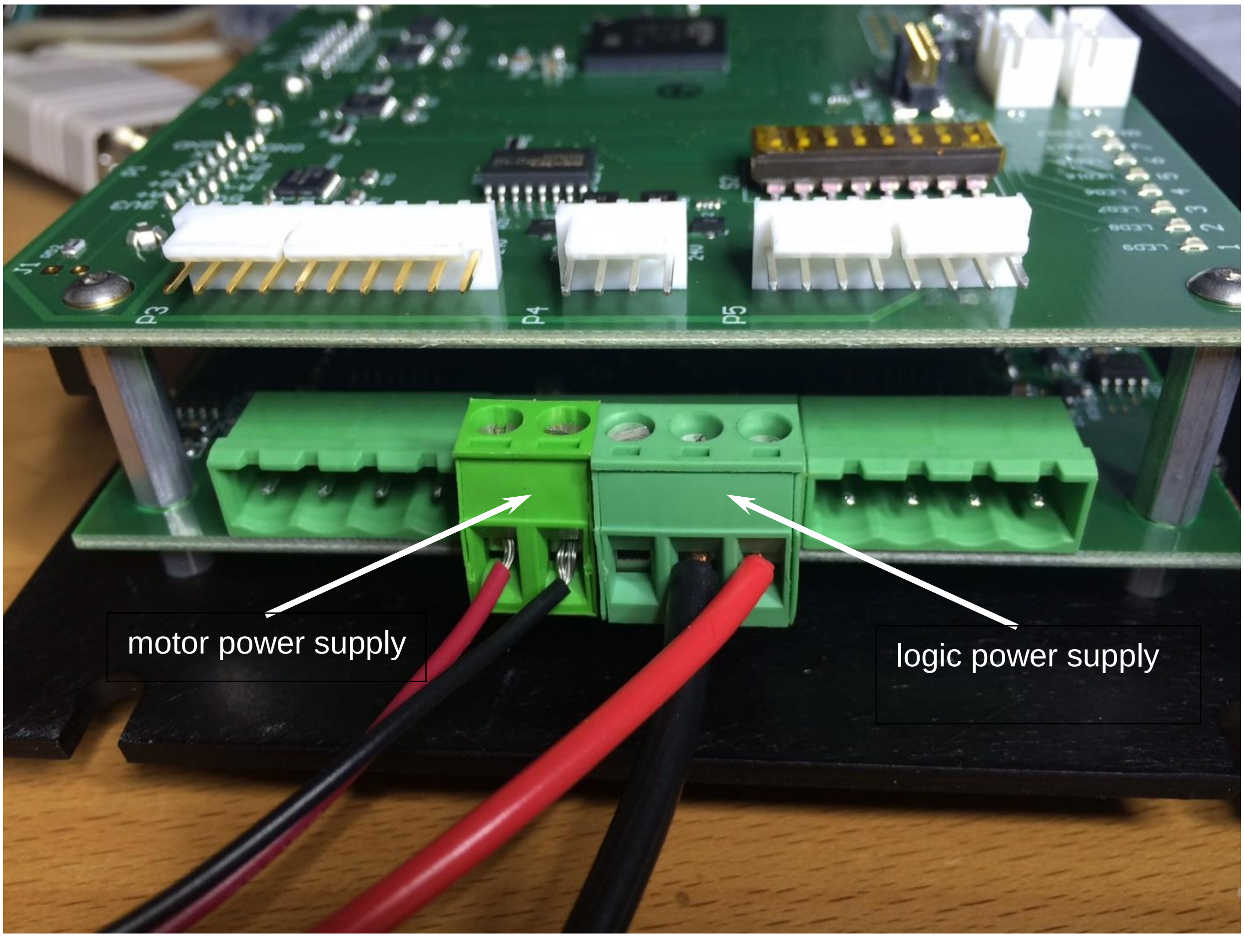


Figure 1-1: power connection via J2

1.2 Connecting the Motor Windings

The motor windings are connected using J1 and J3 on the inverter board. Select the connector on the left (J1) corresponding to the first channel; a second motor (not included in the kit) can be connected to J3 if desired. The connection for the motor in the kit is ready-to-use and you don't need to consider the pin assignment. Figure 1-2 shows the position of the connections for the motor windings (cable colors may vary).

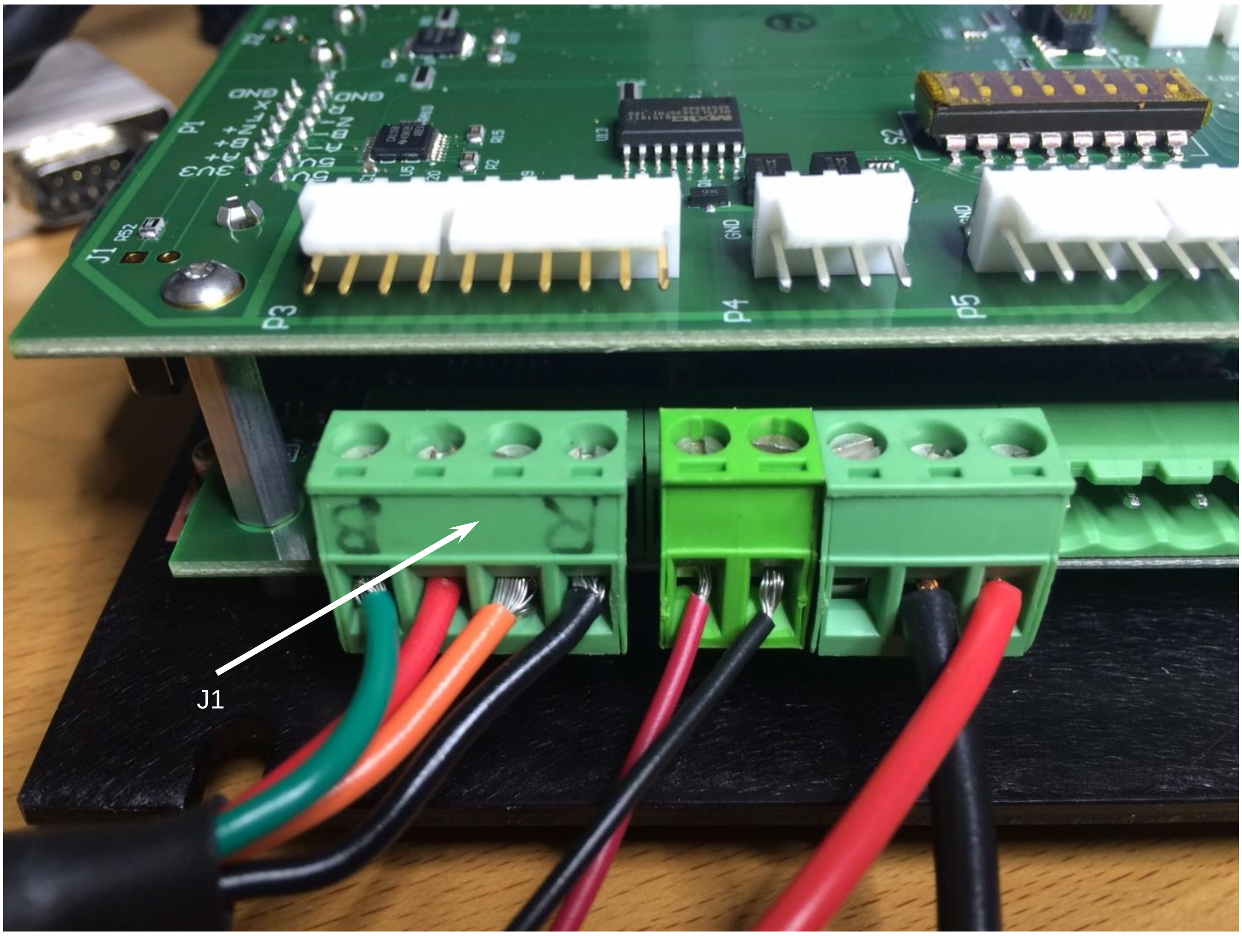


Figure 1-2: Connecting the motor windings to J1 (cable colors may vary)

1.3 Connecting the Encoder

The encoder is connected to P1 on the controller board via a DB 15 connector. A second encoder (not included in the kit) can be connected to P2 if desired. The position of P1 on the controller board is illustrated in *Figure 1-3*. The encoder of the motor that is supplied with the kit is prepared for immediate connection.

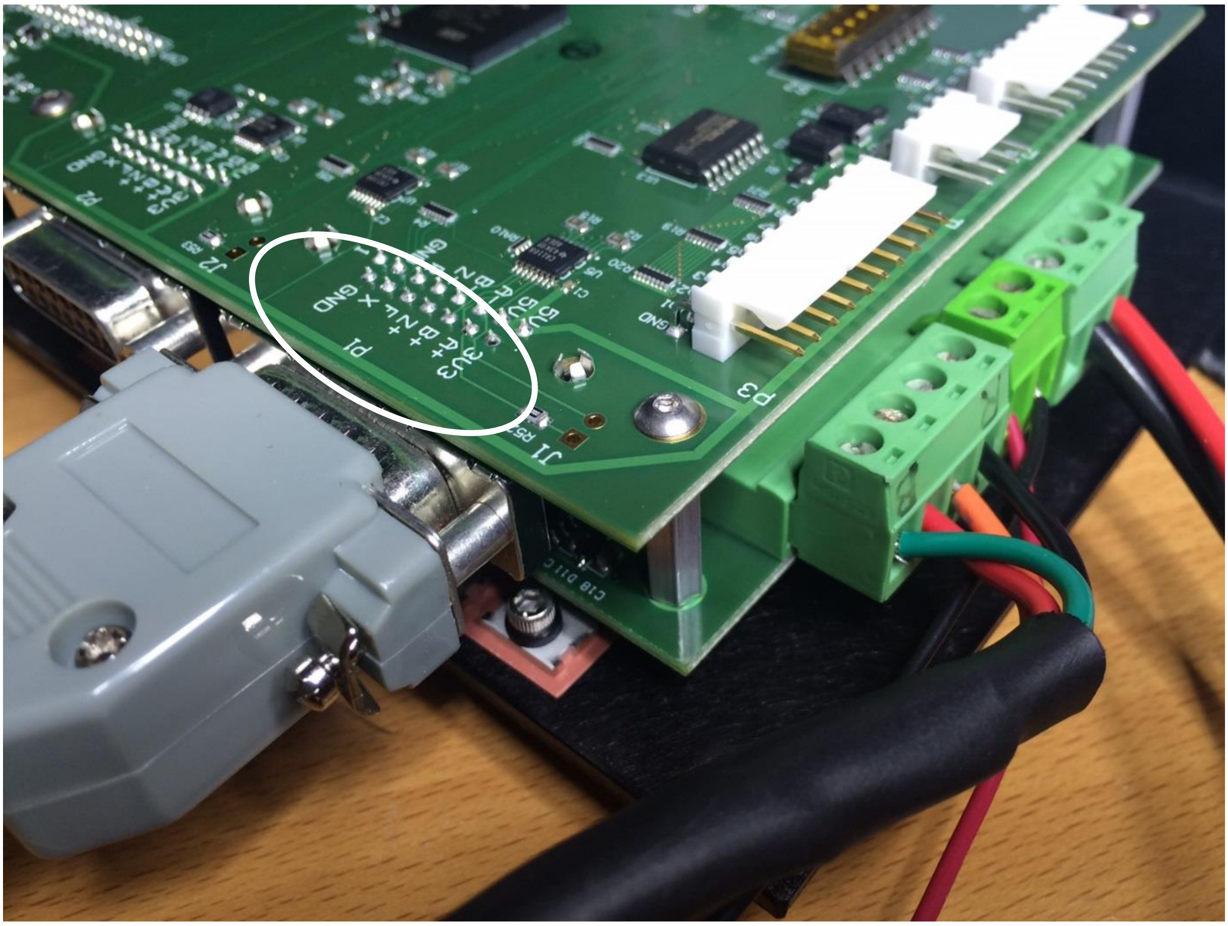


Figure 1-3: Connecting the encoder

1.4 Connecting the Hall Sensors

The YDRIVE-IT-RZT1 kit allows two ways of connecting Hall sensors to the board. They can either be connected the separate connector P5 or together with the encoder at P1. The setting of the DIP-switch S2-0 defines which connection is actually effective.

- S2-0 OFF: connection via P1
- S2-0 ON: connection via P5

The motor included in the kit is prepared for connection via P5; therefore prepare the connection as shown in Figure *1-4* and make sure that S2-0 has been set to ON. The cable colors may be different than in Figure 1-4; in the picture the shield connection is missing.

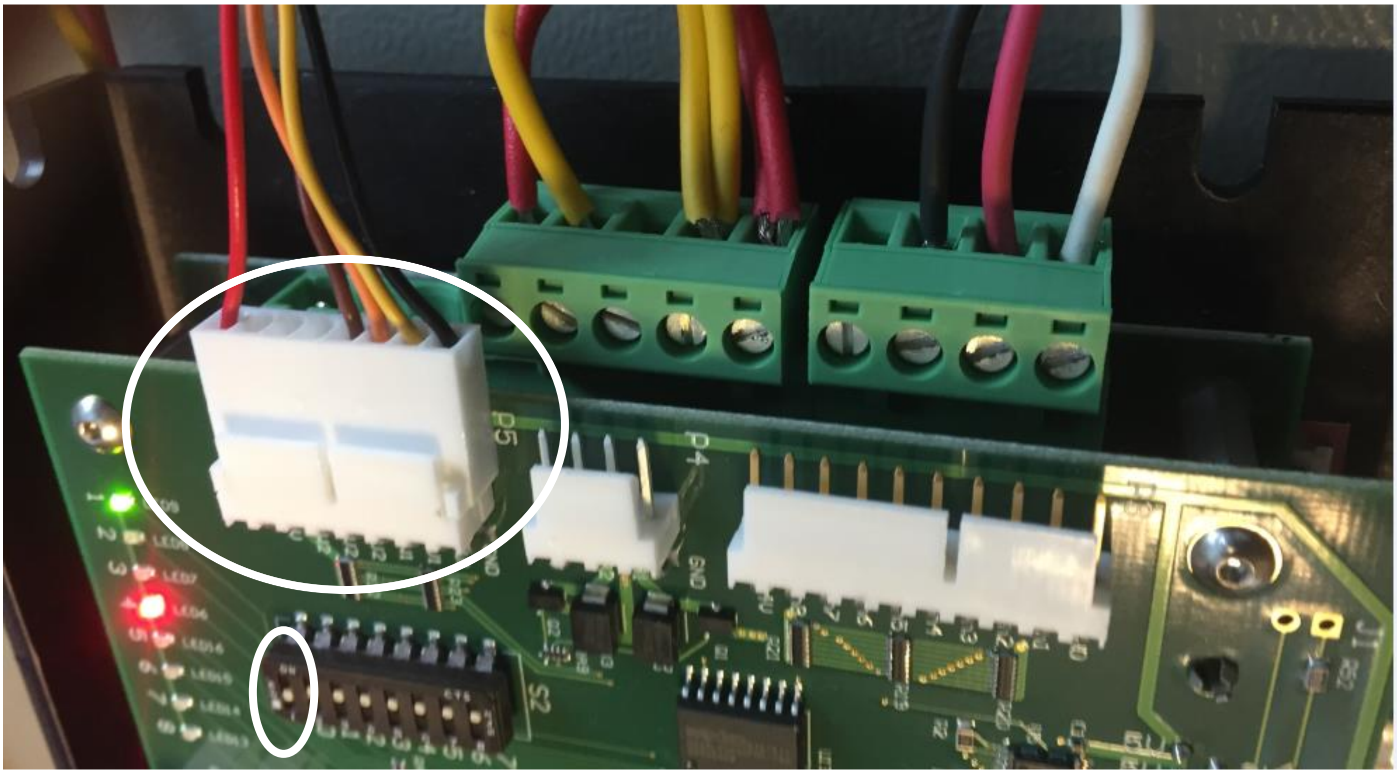


Figure 1-4: Hall sensor connection (cable colors may vary)

1.5 Connecting the RS232 communication cable

Communication with the host PC is done via a serial interface available on P11 of the controller board. The communication cable interfaces with a DB9 RS232 port using straight-through wiring (one male and one female DB9 connector). A suitable serial cable is included in the kit. If your PC does not provide a serial interface, we recommend using a serial-to-USB converter. The connection is shown in Figure 1-5.

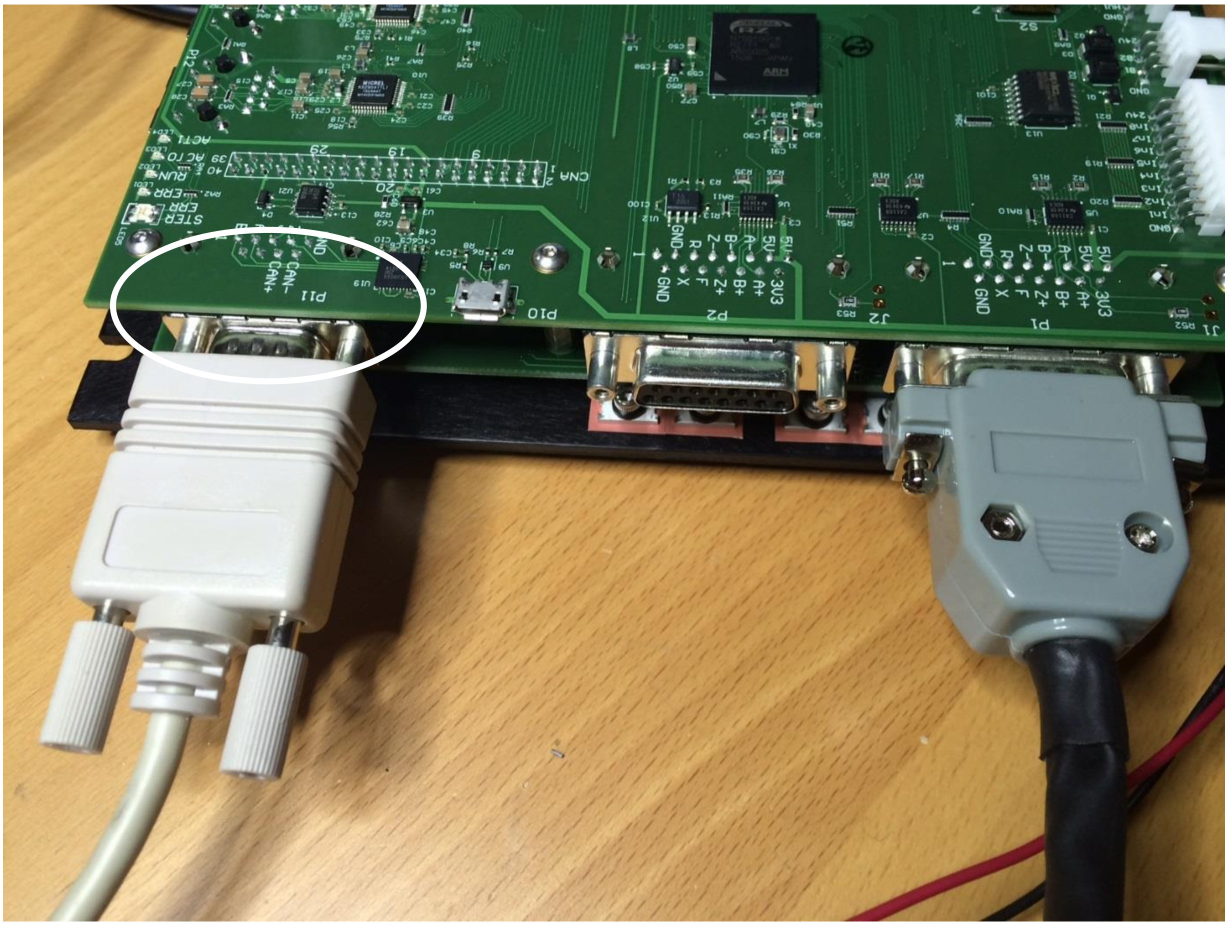


Figure 1-5: Serial connection to host PC

1.6 Connecting the JTAG interface

This connection is required for development purposes or in case new firmware has to be loaded. The JTAG connector is using a MIPI20 connector (as specified by IAR). The pins are spaced 1.27mm apart and the connector is polarized.

The RZ/T1 Solution Kit comes with an IAR I-jet Lite debugger that is ready to connect at P9 on the controller board as shown in Figure 1-6.

Caution: Be careful in plugging the connector to avoid bending pins!

Note: The YDRIVE-IT-RZT1 kit is delivered with the software for spinning the motor pre-installed. The JTAG connector is therefore not needed for running the motor. However the other examples (dual-core operation and EtherCAT) require this connection.

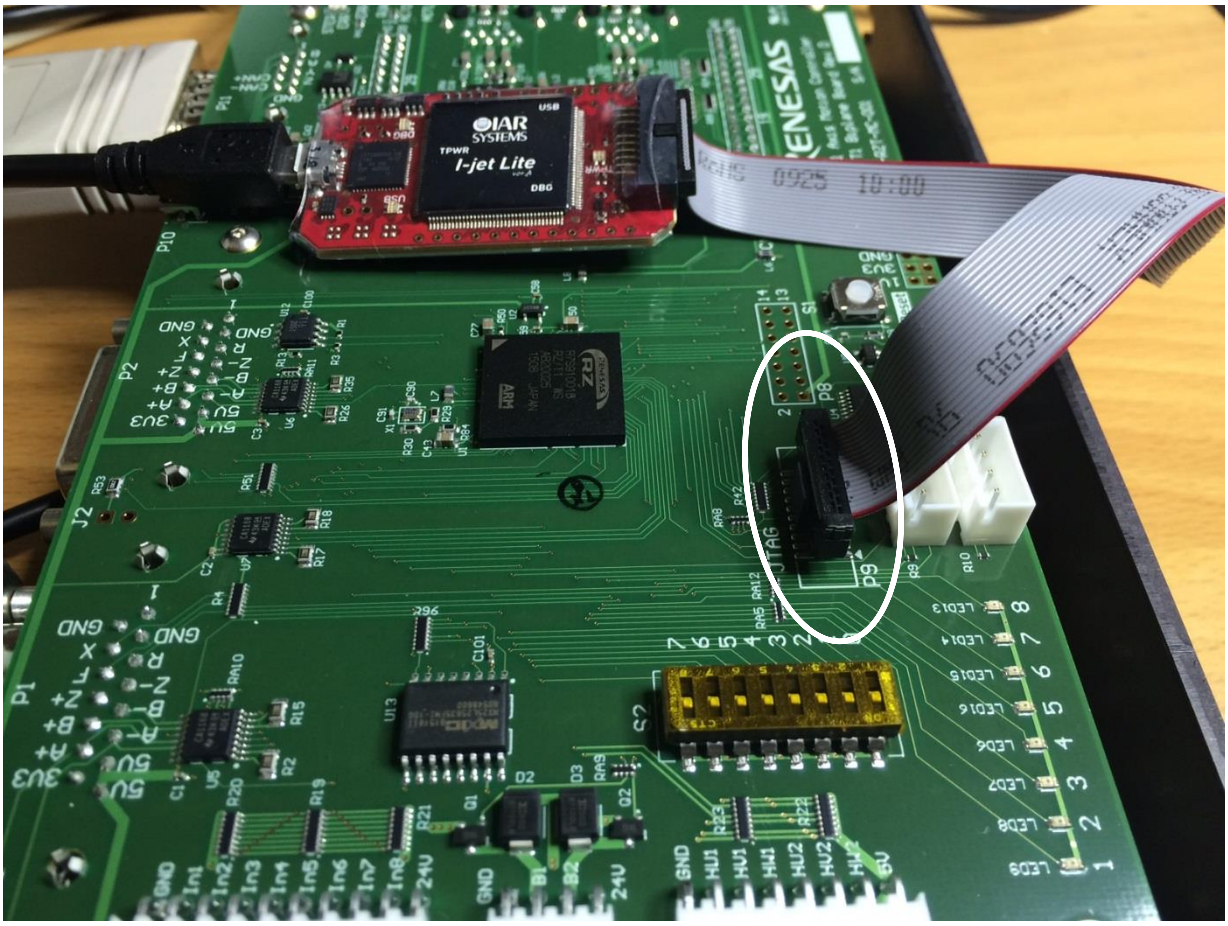


Figure 1-6: Hardware debugger connection

1.7 Connecting the Ethernet Interface

An Ethernet connection to the PC or a PLC can be prepared using one of the two RJ-45 connectors P12 or P13. For running the EtherCAT sample program you can use any of the two connectors.

Note: This connection is only required for operating the EtherCAT sample program. The other examples (dual-core operation and spinning the motor) do not require this connection.

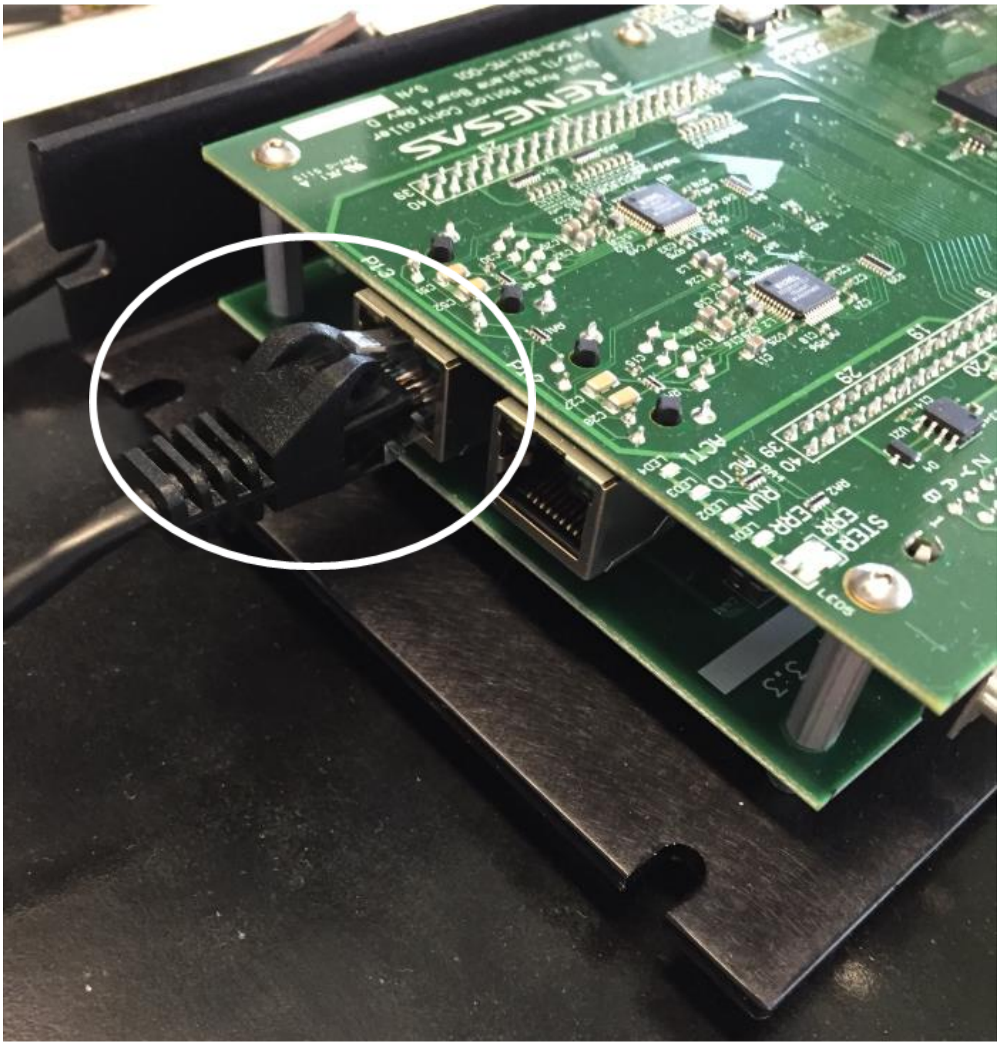


Figure 1-7: Ethernet connection

1.8 Sanity check

The YDRIVE-IT-RZT1 Solution Kit comes with preinstalled firmware. Once the logic power is supplied (see chapter 1.1), LEDs #1 to #4 (LED6…9 in the circuit diagram) should turn on as shown in *Figure 1-8*. LED #2 should blink with 1 s period.

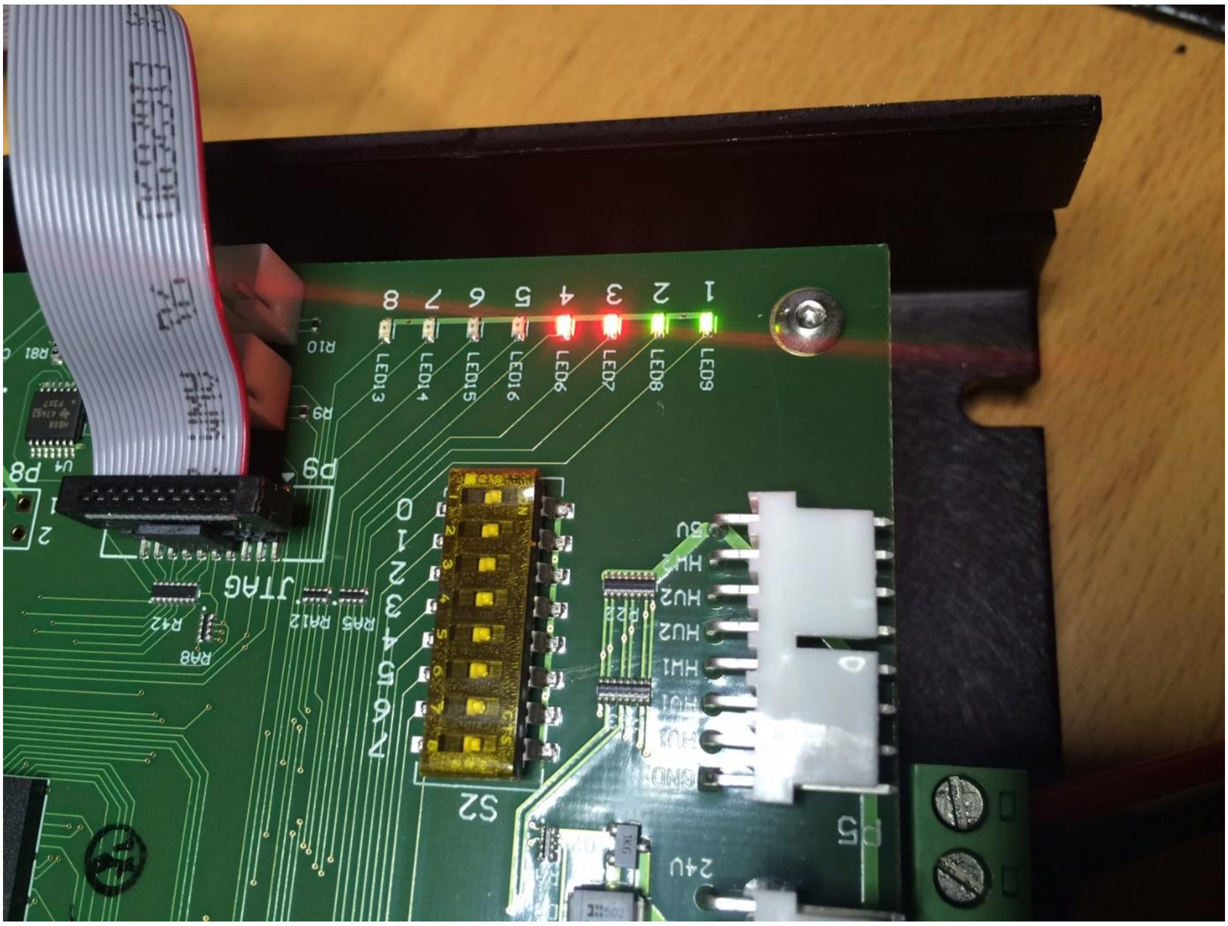


Figure 1-8: LEDs on the controller board after power-up

2 Running the Motion Control Software

In this chapter we will run the pre-installed motion control firmware package on the RZ/T1 and a PC-based user interface software that we refer as RZ/T1 Motion Utility.

Note: To run the motion control software all connections described in chapters 1.1 to 1.5 need to made and the supply voltages for logic and motor must be attached.

2.1 Software Setup – Install RZ/T1 Motion Control Utility

The RZ/T1 Motion Control Utility is compatible with Windows XP and the newer versions up to Windows 10. It requires Microsoft.NET 4.0 which will be installed if needed.

The installation is possible with two options:

- Running the Setup.exe file
- Running the RZ_T1_Utility.msi

Both options lead you to the dialog box in *Figure 2-1*.

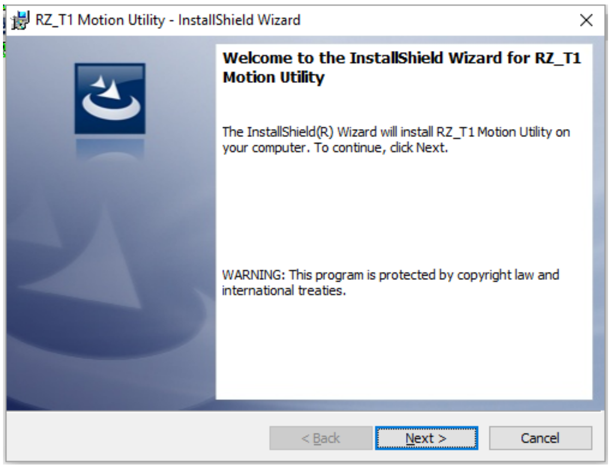


Figure 2-1: RZ/T1 Motion Utility installation

Press 'Next' and you will be asked for target folder as shown in *Figure 2-2*. Technically any location is OK but we recommend to stick with the proposed location.

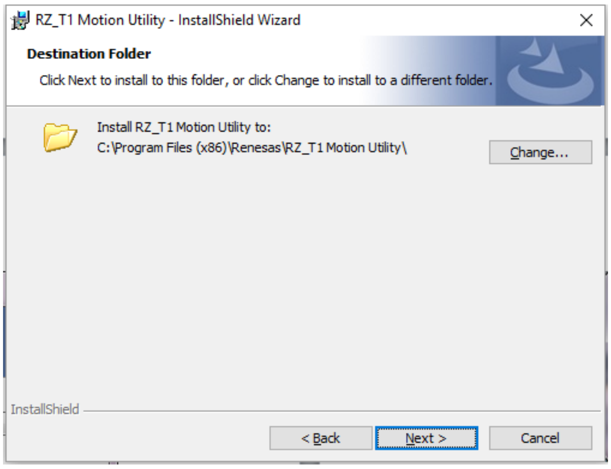


Figure 2-2: Target folder selection

Start the newly installed utility. The following screen should be displayed:

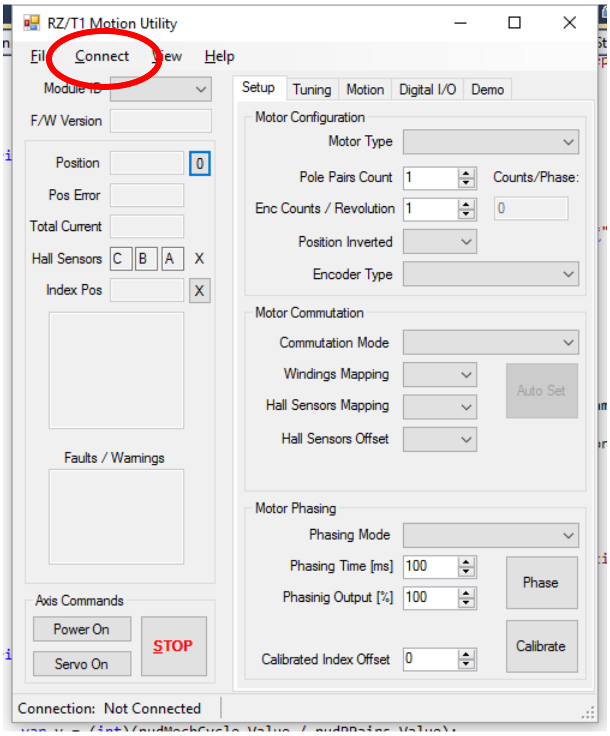


Figure 2-3: RZ/T1 Motion Utility start-up screen

Click the 'Connect' option from the main menu to select the serial interface to be used for communication with the RZ/T1 motion controller. Select the RS232 tab and then the appropriate serial port number from the drop-down list as illustrated in Figure 2-4. You may have to check the port number in the MS-Windows device manager.

Note: The name of the serial port can be selected from the list of predefined entries. If your serial COM port is not listed, then you can enter any name by typing it in the control.

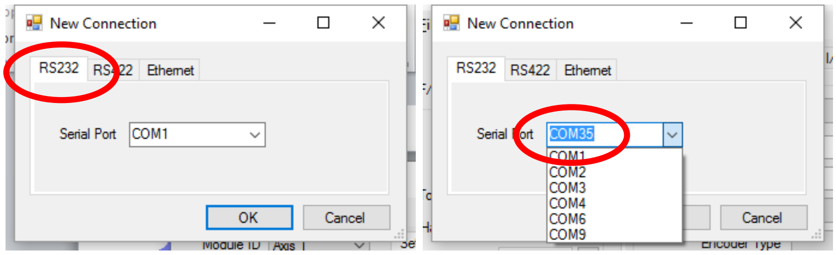


Figure 2-4: Selecting the serial COM port number

Once the COM port is confirmed the motion utility window should look as shown in *Figure 2-5* and report an established connection in the lower left corner of its window.

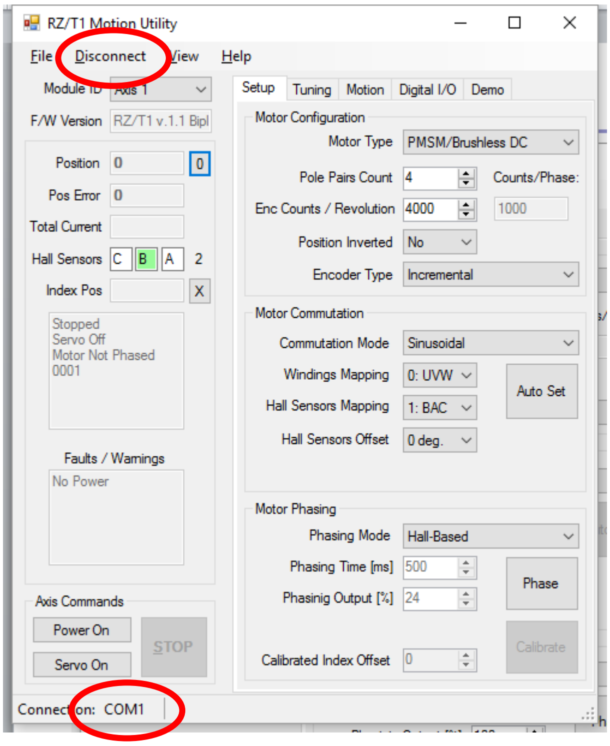


Figure 2-5: Established serial connection

Note that the 'Connect' menu entry changes to 'Disconnect' and the connected port is displayed in the bottom left corner.

2.2 Check the Hall Sensors and the Encoder Feedback

Turn the motor axis manually and observe the change in the numbers next to the ‘Position’ label. When the numbers change, this is a good indication that the encoder works properly.

Confirm that the Hall sensor indicators also change and at any time there is at least one of the signals green. They should never be all white (off) or all green (on).

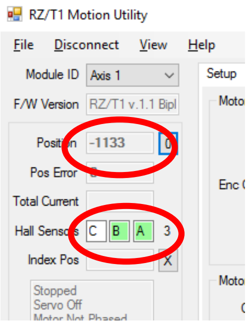


Figure 2-6: Controlling encoder and Hall sensors

2.3 Check Configuration Parameters

The motion control firmware that is pre-installed on the RZ/T1 Solution Kit can handle a variety of different motor types and configurations. Several tabs in the Motion Control Utility software allow to enter the required parameters. A working set-up can be saved as well as re-loaded in the ‘File’ menu.

All settings for the motor that is delivered with the RZ/T1 Solution Kit are stored in the Speeder.Motion.160406.mtr file on the CD. Re-load this file into the utility and check as shown in the subsequent steps whether the all settings have been restored properly.

2.3.1 Check the Motor Configuration

The Motor Type, Pole Pairs Count and Enc. Counts / Revolution should match the numbers in Figure 2-7.

Note: The above parameters reflect the characteristics of the motor and the encoder provided. In case motors other than the one included in the kit is used – these parameters may need to be changed.

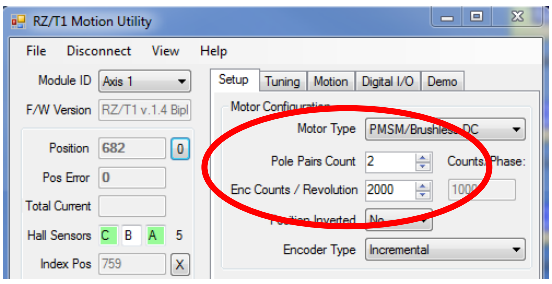


Figure 2-7: Checking the motor configuration

2.3.2 Check Motor Commutation Parameters

Please check whether the motor commutation parameters are set as shown in Figure 2-8.

Note: The above parameters reflect the current wiring of the motor and the hall sensors. In case motors other than the one included in the kit is used – these parameters may need to be changed.

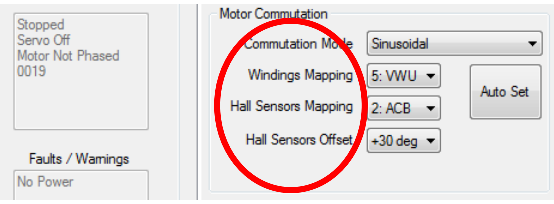


Figure 2-8: Motor commutation parameters

2.3.3 Check Motor Phasing Parameters:

Please check whether the motor phasing parameters are set as shown in Figure 2-9.

Note: If HALL sensors are not installed then different Phasing Mode must be used. The Algorithmic phasing is preferable if there are no obstructions to the rotor.

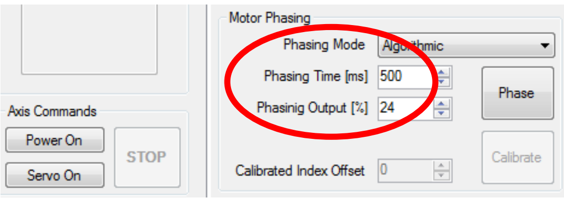


Figure 2-9: Checking the motor phasing parameters

2.4 Enable Inverter Power

The Motion Control Utility has two buttons to control the motor power. First use the 'Power On' button in the lower left corner as shown in Figure 2-10.

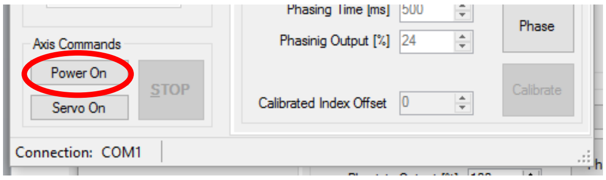


Figure 2-10: Using the Power On button

Note that the LED7 on the controller board should turn off. This indicates that everything is OK with the power stage of the inverter.

2.5 Move Motor in Open Position Loop

Navigate to the ‘Motion’ tab of the RZ/T1 Utility to spin motor Open Loop. Press the ‘Forward’ or ‘Reverse’ button as shown in Figure 2-11 to initiate a spin of the motor.

Note: Pressing the ‘Forward’ button should result in increasing position reported by the encoder. If this is not the case then the configuration parameters must change to reflect the actual wiring or motor parameters.

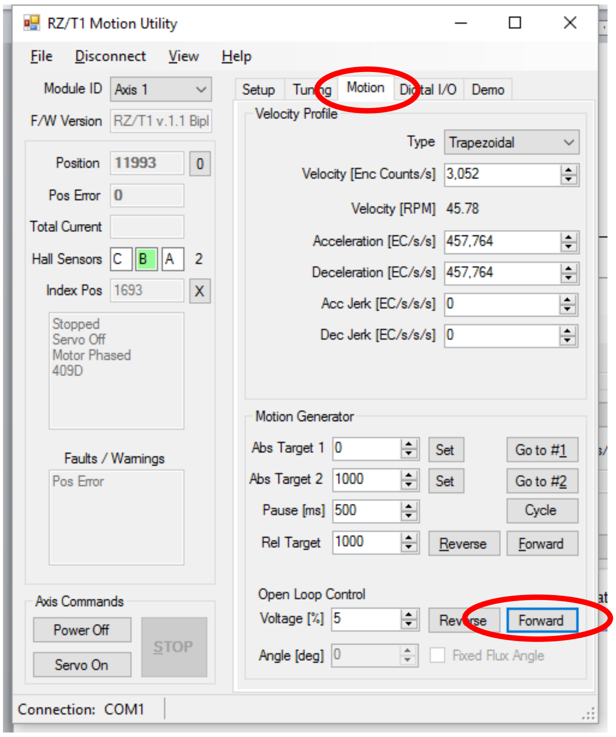


Figure 2-11: Moving the motor position forward

2.6 Servo Control Usage

Turn on the Servo Control by pressing the button ‘Servo On’.

Note: At this point the shaft of the motor should resist any manual attempt to be turned. This is an indication of normally operating servo control at steady state.

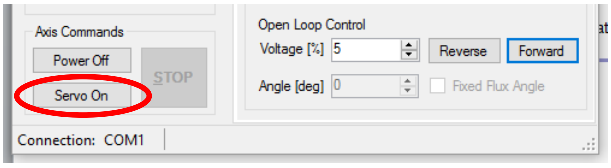


Figure 2-12: Activating Servo Control

2.6.1 Execute motion to relative or absolute target position

Use the buttons in the Motion Generator group to start motion to a relative or absolute target position. Two positions can be entered in the ‘Abs Target 1’ or 2 fields. The values can be typed manually or changed with the scroll wheel of the mouse. The ‘Set’ buttons can be used to copy the current position in the controls to the left of them. The ‘Go to #1’ and ‘Go to #2’ buttons will spin the axis to the pre-defined positions as indicated in Figure 2-13.

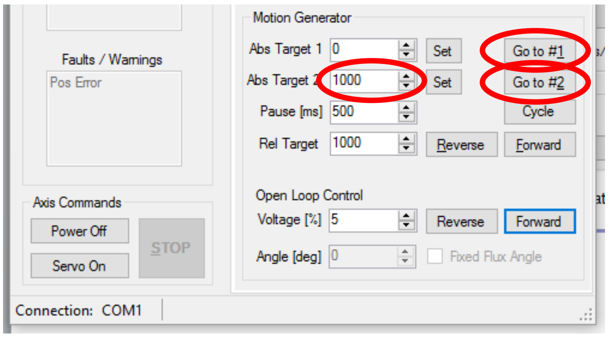


Figure 2-13: Move to absolute target positions

Alternatively you can make forward and reverse moves to a relative position specified in the ‘Rel Target’ field.

Note: Observe whether the current position reaches the defined target.

2.6.2 Change the speed and the acceleration of the requested move

Use the controls at the 'Velocity Profile' group to enter new values for velocity and acceleration. Observe the change in the execution motion when you repeat moves to pre-defined absolute or relative positions.

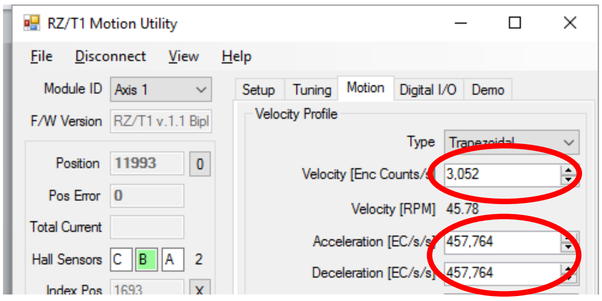


Figure 2-14: Changing velocity and acceleration

Note: If the motion request causes an error or any of the built-in interlocks is triggered then the servo control will be turned off and the subsequent commands will be ignored. Use the 'Servo On' button to enable the serve before any further motion request.

2.6.3 Cyclic Motion from Point to Point

You can define two target positions and start a cyclic motion between them. As an example enter two positions about 10000 counts apart and press the 'Cycle' button as illustrated in Figure 2-15. The motor should start moving between them.

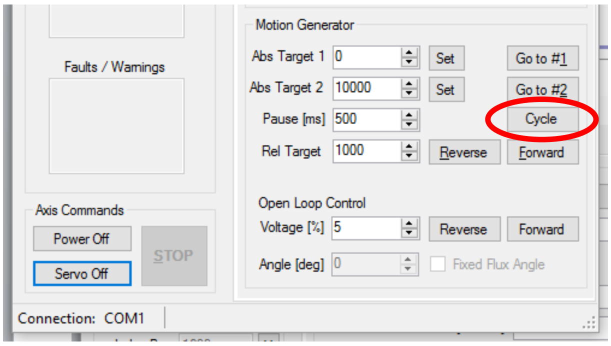


Figure 2-15: Initiating a cyclic movement

2.7 Using the Motion Scope

Open the Motion Scope by selecting ‘Motion Scope’ in the ‘View’ menu. The screen in Figure 2-16 should be displayed.

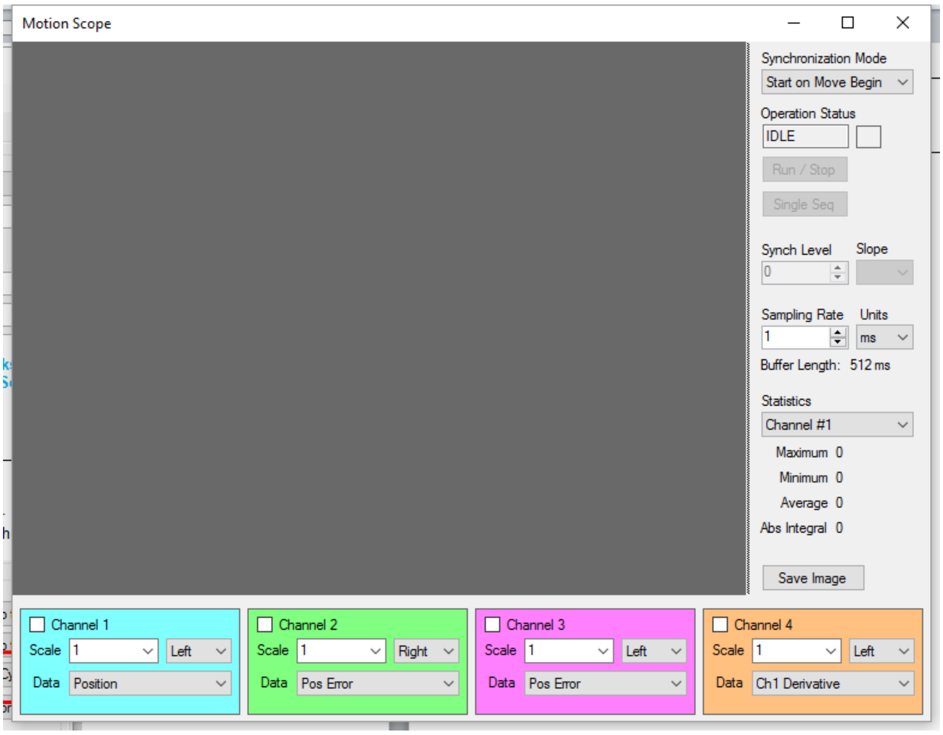


Figure 2-16: Starting the Motion Scope function

The screen resembles a four-channel oscilloscope; for each channel you can select data from a number of characteristics to be displayed.

2.7.1 Configure the Motion Scope channels

Configure the Motion Scope channels 1 and 2 like shown in Figure 2-17.

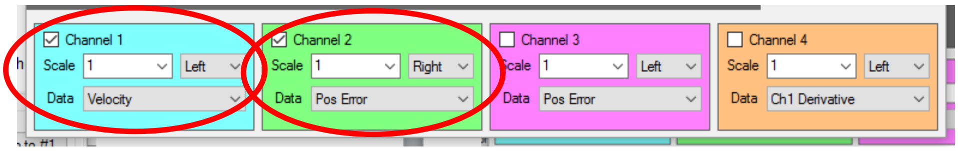


Figure 2-17: Selecting functions for channels 1 and 2 of the Motion Scope

The picture in Figure 2-18 should be displayed after one cycle of motion.

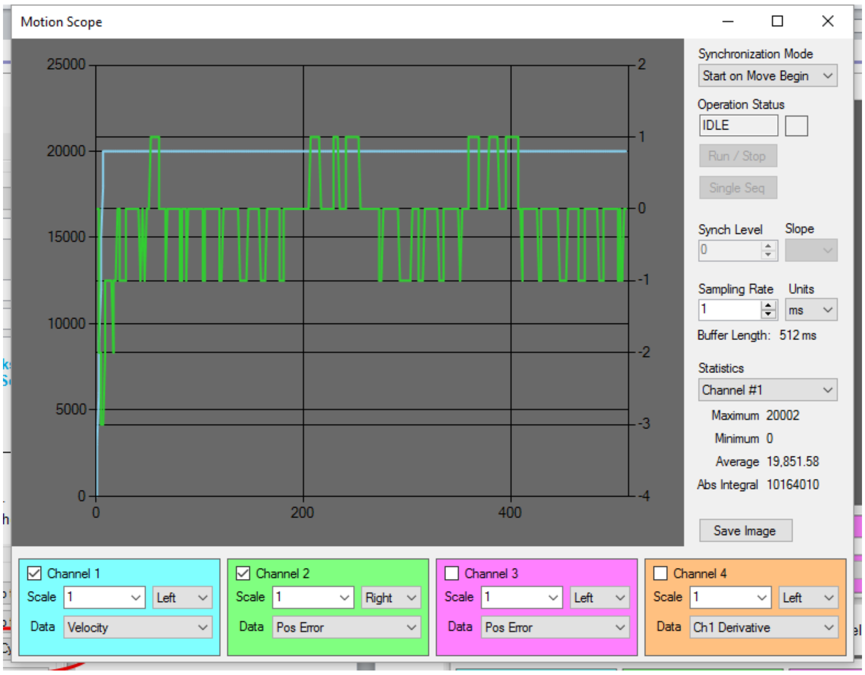


Figure 2-18: Motion Scope reading after one motion cycle

Now change the velocity to 25,000 and the acceleration to 450,000. The diagram for the velocity and the position error should change as shown in Figure 2-19.

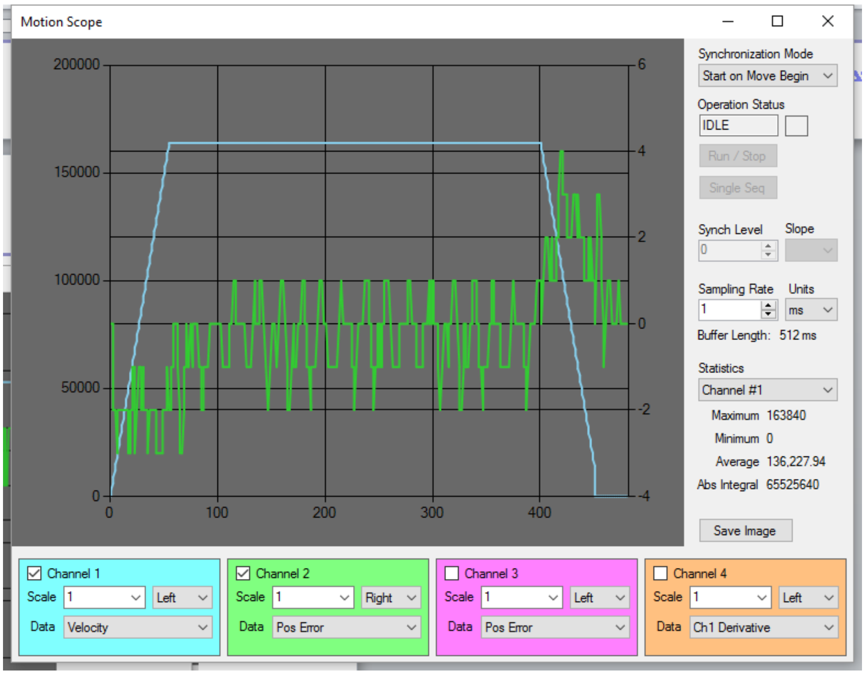


Figure 2-19: Motion Scope reading with changed velocity and acceleration parameters

2.7.2 Change PID Parameters

Changing the PID parameters of the control algorithm will result in significant changes of the position error. In this chapter we will go through some examples to make such changes visible.

2.7.2.1 Evaluate the role of the different PID gains

Navigate to the ‘Tuning’ tab and change the ‘KP’ parameter from 250 to 25 as shown in Figure 2-20.

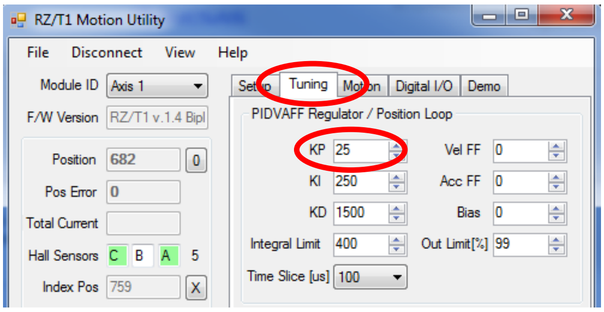


Figure 2-20: Changing the ‘KP’ parameter

In the Motion Scope you can now observe the change in the position error oscillating much more than before as can be seen in Figure 2-21.

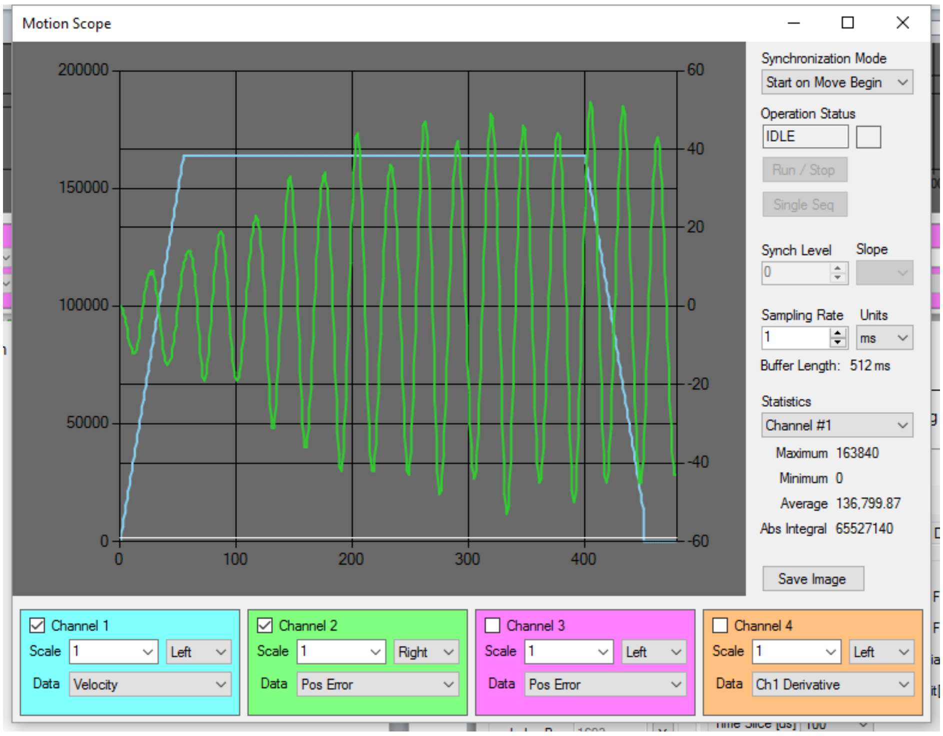


Figure 2-21: Motion Scope reading after changing the KP parameter (actual reading may vary)

Now change the ‘KI’ parameter to 25 as well as shown in Figure 2-22. Note the difference in the sound the motor makes with different KP/KI parameters.

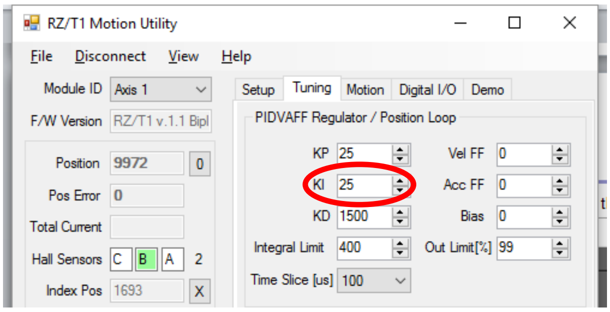


Figure 2-22: Changing the ‘KI’ parameter

The Motion Scope reading reflects the position error proportional to the commanded acceleration and deceleration as can be seen in Figure 2-23.

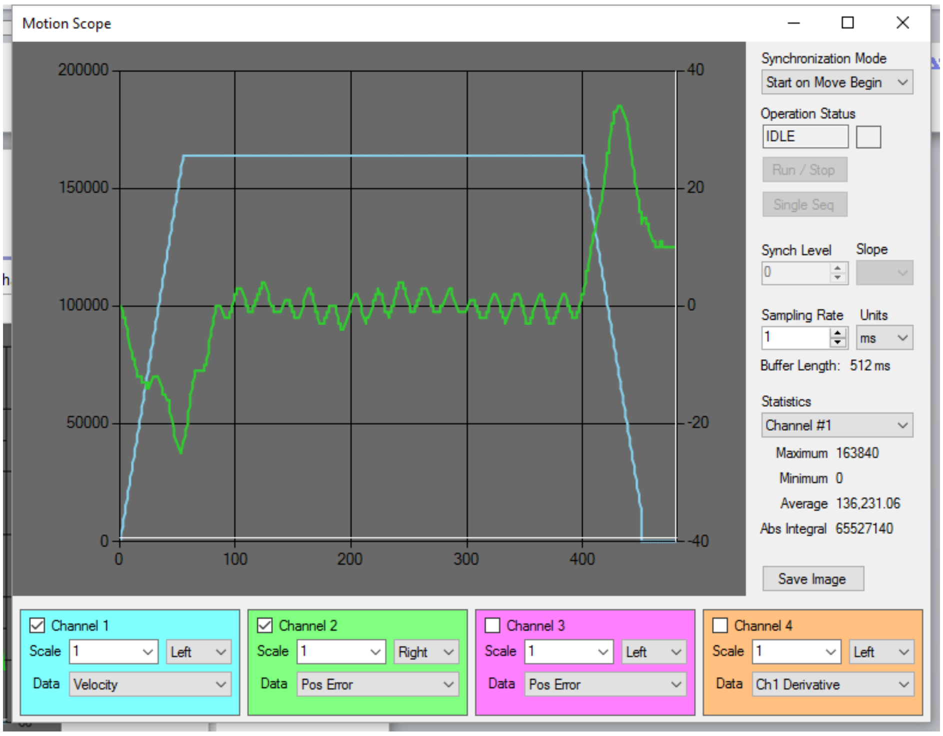


Figure 2-23: Motion Scope reading after additionally changing the ‘KI’ parameter (actual reading may vary)

2.8 Recover from Motion Errors

In this chapter we will reduce the maximum position error to cause a fault. Set the position error limit in the 'Tuning' tab to 10 as shown in Figure 2-24. The motion will stop and the status will indicate the fault caused by exceeding the maximum position error.

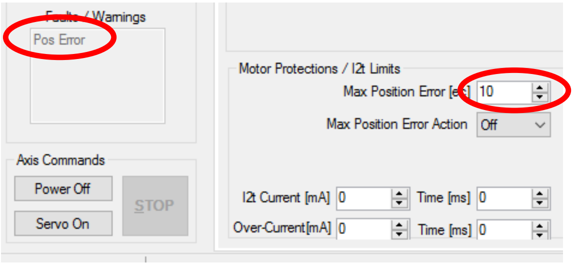


Figure 2-24: Changing the maximum position error

Now restore the Max Position Error to 800 and press the 'Servo On' button to recover from the fault as shown in Figure 2-25.

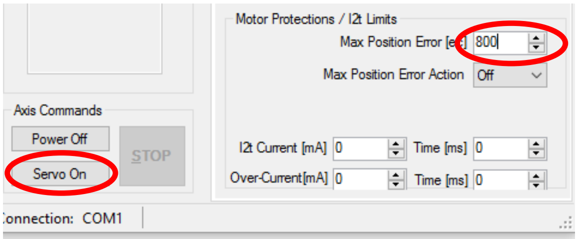


Figure 2-25: Recovering from position error

3 Simple Dual Core LED Blink Project

A good way to understand the RZ/T1 is by starting a very basic program from scratch. In this section we will describe how to start with the RZ/T1 example program provided on the CD that comes with the RZ/T1 Solution Kit. This example program had been originally written for the RZ/T1 RSK+ board, and some small modifications were introduced for the Solution Kit Board version on the CD. The program flashes two LEDs, one is operated by the Cortex-R4F core in the RZ/T1, the other one is operated by the Cortex-M3 core.

Copy the project from the CD to your working directory on the PC. Then open the project with a double-click on the workspace file 'RZ_T1_init.eww' as shown in Figure 3-1. This will open EWARM and give you access to the different versions of the example project.

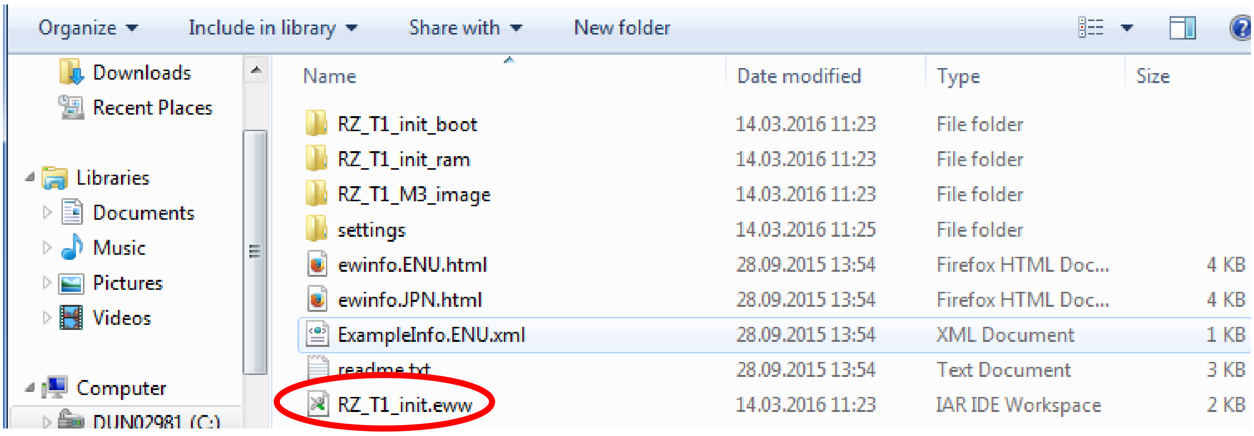


Figure 3-1: Opening the example project

IAR EWARM will now open the project and in the EWARM window you will see the example workspace child window similar to Figure 3-2.

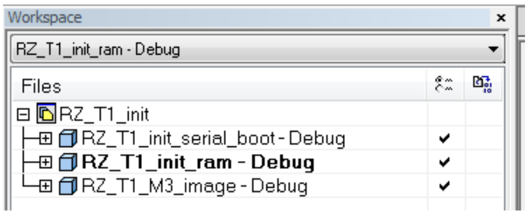


Figure 3-2: Updated project overview

The 'RZ_T1_init_serial_boot – Debug' project and the 'RZ_T1_init_ram – Debug' project are projects for the Cortex-R4F core of the RZ/T1. The first one is meant to be stored in the QSPI Flash and the second one is for internal RAM. Finally the 'RZ_T1_M3_image –Debug' is only for the Cortex-M3 core without using the hardware accelerator for industrial Ethernet communication.

3.1 Using the ‘RZ_T1_init_ram – Debug’ project

In this section we will explain how to run the example project on the RZ/T1 Solution Kit. The following connections have to be made to the board:

- Connect the 5-pin connector that is delivered with the kit to a 12-24V DC power supply and plug it to the J2 position of the inverter board. For this example it is sufficient to power the control part of the Solution Kit.
- Moreover, it is needed to connect the I-Jet Lite debugger to the connector P9 on the controller board and to a USB port from the host computer.

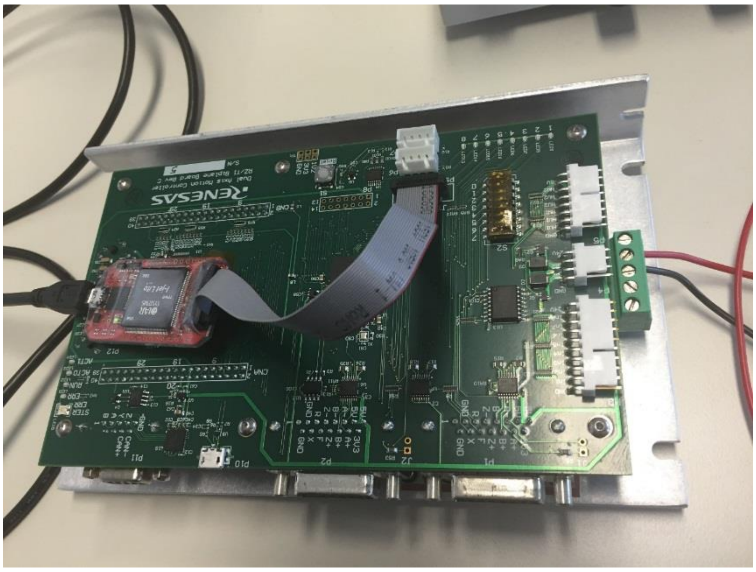


Figure 3-3: Connections for running the blinking LED example

The ‘RZ_T1_M3_image –Debug’ program is ready to be downloaded, so please ensure that you are running one core operation, by selecting ‘Debug’ in the workspace and by clicking the ‘Download and Debug’ button as shown in Figure *3-4*.

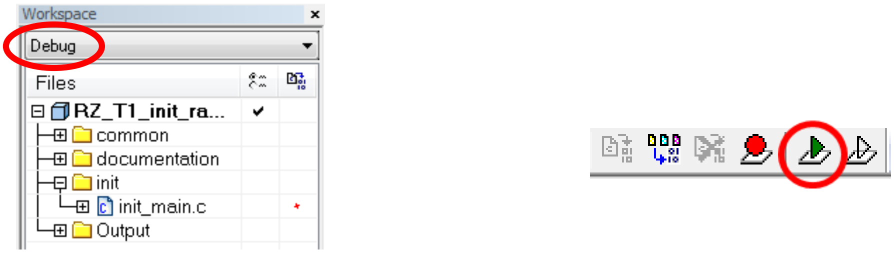


Figure 3-4: one core project option and ‘download and debug’ button

Note: If you like you can modify the program by changing the LEDs to be used. The subsequent table shows how LEDs are connected to RZ/T1 port pins on the Solution Kit board.

LED number	RZ/T1 port
LED9	P46
LED8	P45
LED7	P44
LED6	P43
LED16	P42
LED15	PT5
LED14	PT1
LED13	PT3

3.2 Dual Core Debugging Preparation

The RZ/T1 version used in the RZ/T1 Solution kit provides a dual core architecture. The RAM example program provides also the possibility to do dual core debugging. By selecting the 'Debug DualCore' version in the workspace as shown in Figure 3-5 it is possible to change it to a dual core project. It is the same code but some parameters for the configuration change.

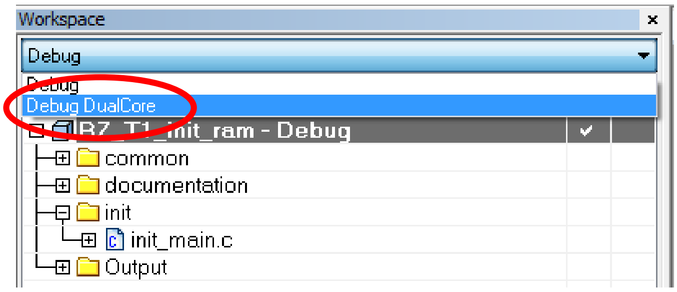


Figure 3-5: Select the dual-core project

The dual core operation on the RZ/T1 works as follows: The dual core versions of the RZ/T1 provide a Cortex-R4F processor for Real Time processing and a Cortex-M3 core with a RIN engine for communication using various industrial Ethernet standards. The Cortex-R4F core behaves during the boot process as a master core and realizes the booting routine while the Cortex-M3 core is kept in Reset. Figure 3-6 Illustrates how this is then handled by the software.

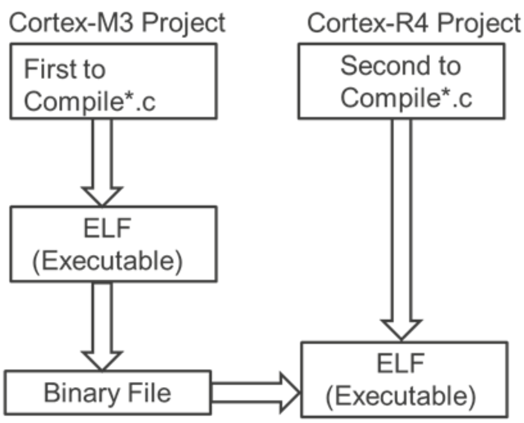


Figure 3-6: Software handling for dual-core operation

The Cortex-M3 project is converted to a binary file which is included to the Cortex-R4F Project. In the example project the configuration is already prepared, but let's highlight how it is done. By clicking to 'Project' and then to 'Options…' in the EWARM menu bar the 'Option' window in Figure 3-7 opens. In the 'Linker' and 'Debugger' section you can see the modifications done for this purpose.

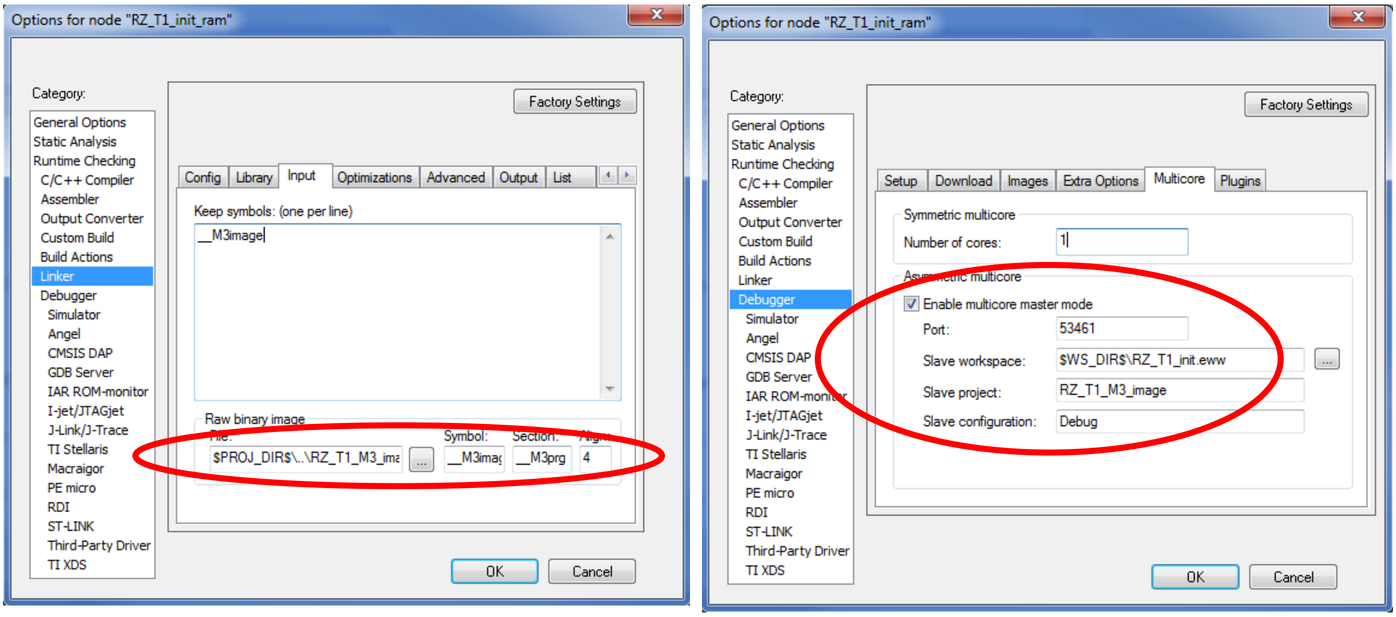


Figure 3-7: Linker and debugger options for a dual core project

With these settings the Cortex-M3 code is embedded into the Cortex-R4F code and the Cortex-R4F copies the code to the Cortex-M3 memory as part of the boot process. After the configuration is done, the Cortex-M3 core remains in Reset. In order to change it, it is required to modify the SWRR2 register as follows.

```
Core wake up.
#ifdef DUAL_CORE

    /* release M3 */
    SYSTEM.PRCR.LONG = 0x0000A502U;
    SYSTEM.SWRR2.LONG = 0;

#endif
```

This code is already included in the Cortex-R4F program.

3.3 QSPI Flash Related Changes

By the IAR v.7.40 it is needed to do also some configurations options must be changed. The main difference between the RZ/T1 RSK+ board and the Solution Kit board is the QSPI size: 512 Mbit for the RSK+ board and 256 Mbit for the Solution Kit board.

On the Solution Kit CD we provided a special project called ‘flashloader’. This ‘flashloader’ project must be copied into the ‘RZ_T1_init_Serial_boot’ project.

Now some project settings must be changed to make sure that files from the ‘flashloader’ project are used in the build process. Open the project options in IAR and select ‘Debugger’ and ‘Download’ and change the settings as shown in Figure 3-8 There you can see that the ‘Use flash loader(s)’ and ‘Override default .board file’ boxes are checked and that the board file path points to the ‘flashloader’ project. These options provide the flashing in the RZ/T1.

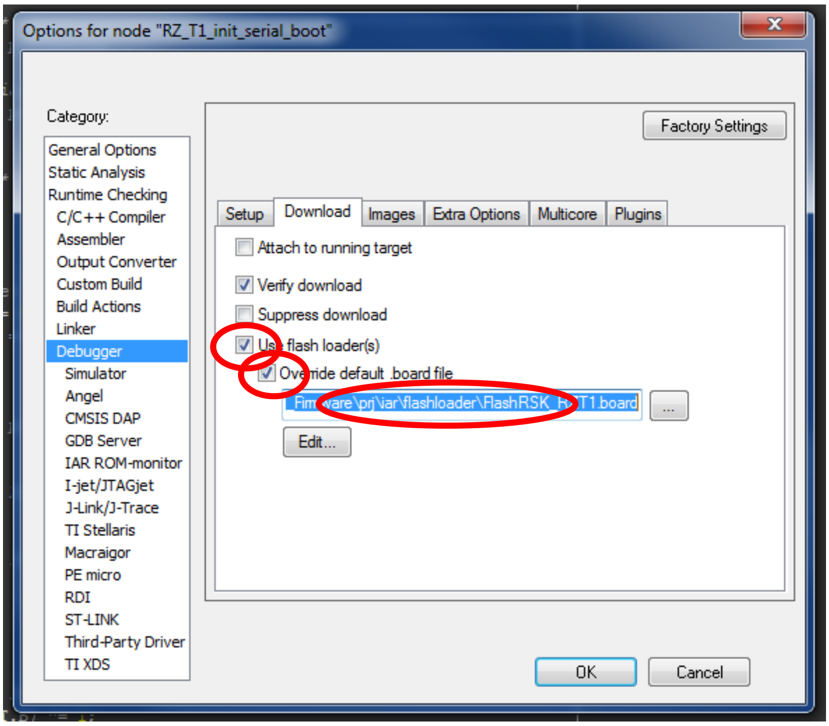


Figure 3-8: Debugger Download option change

Finally the .mac file needs to be changed in ‘Options’ ‘Debugger’ ‘Setup’ as shown in Figure 3-9. Similar to the last step the .mac file from the ‘flashloader’ project must be used for our simple LED flash project.

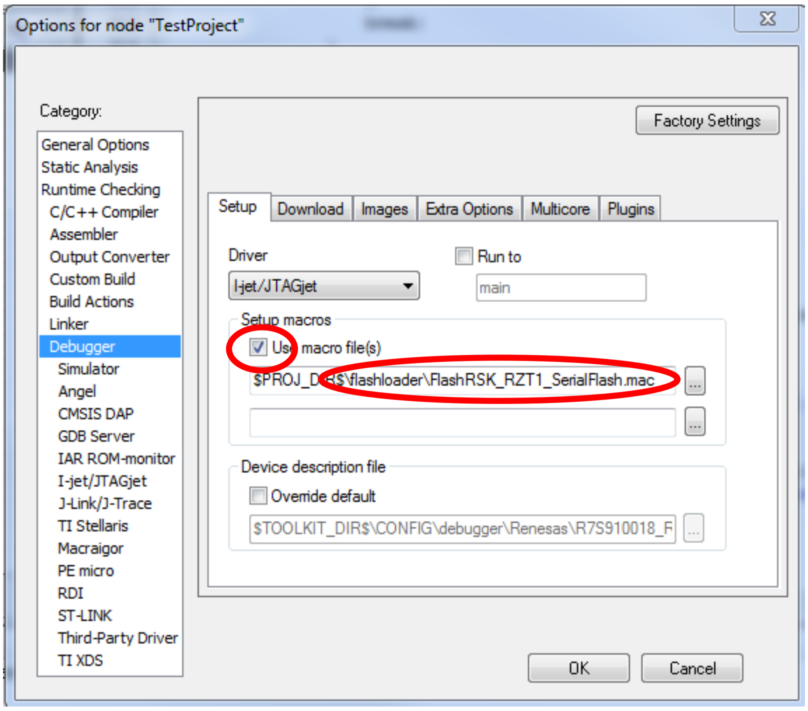


Figure 3-9: Debugger Setup option change

3.4 Dual Core Debugging

You can now download the code into the target processor by clicking to the ‘Download and Debug’ in IAR EWARM; the download progress will be illustrated as shown in *Figure 3-10*.

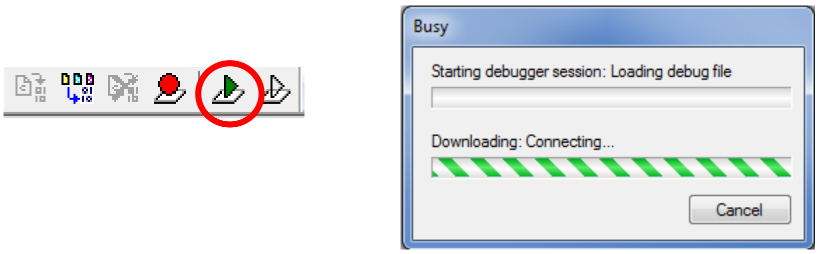


Figure 3-10: Downloading the example project

Now a second IAR EWARM instance is opened automatically for the ‘Slave’, namely the Cortex-M3 core in the RZ/T1, while the original window is used for the ‘Master’, the Cortex-R4F core in the RZ/T1. Furthermore you will see a couple of new buttons dedicated for multicore debugging in both EWARM instances.

The two instances can be differentiated by their title bars or by the way, in which the status of the respective cores is illustrated; Figure 3-11 gives you an example.

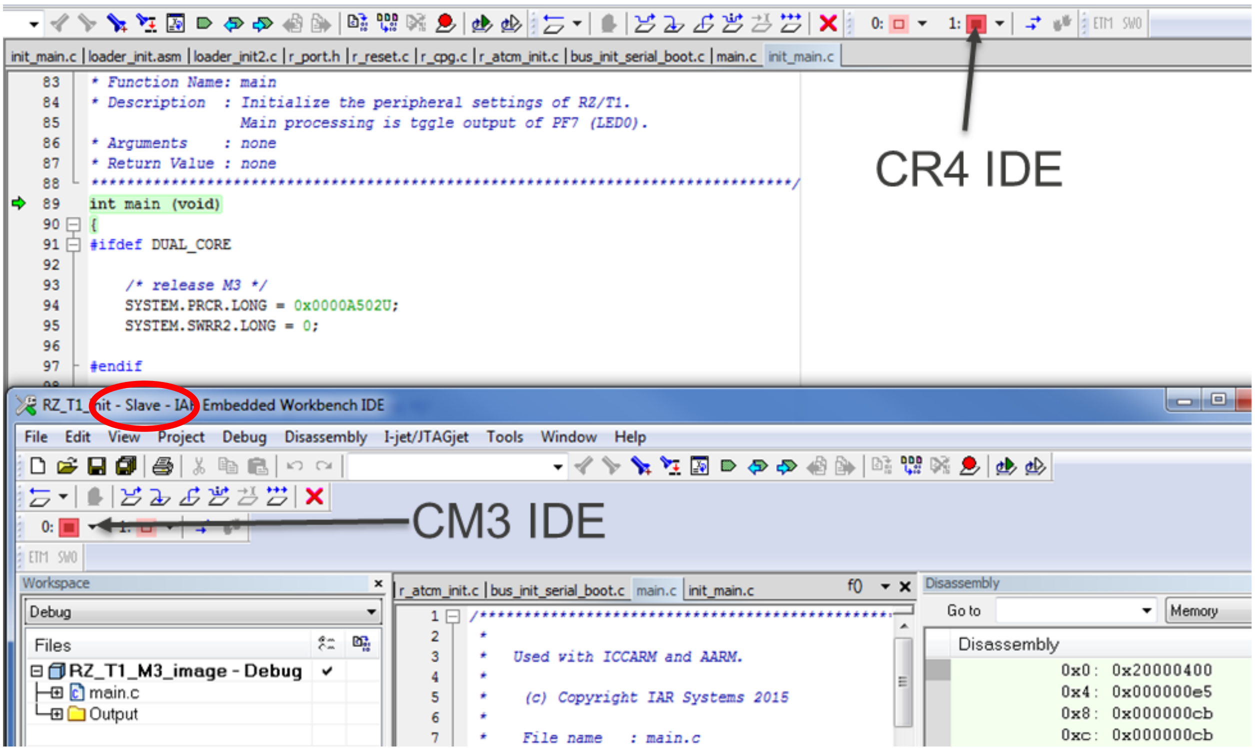


Figure 3-11: Differentiation of EWARM instances

Now you have to start all cores and then stop all cores with two mouse clicks shown in Figure 3-12. Here it does not matter, in which of the EWARM instances you issue these commands.

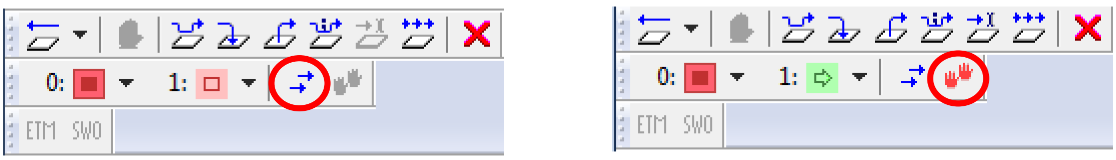
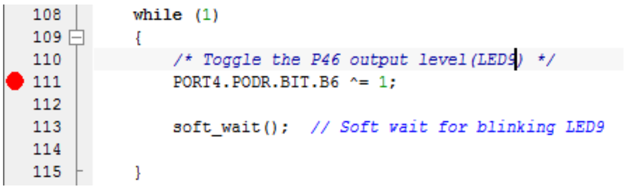
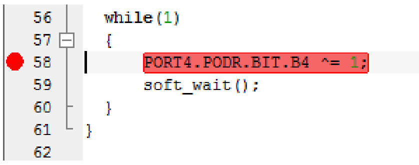


Figure 3-12: Start and then stop all cores

After that once again click ‘Start all cores’ as shown in Figure 3-12. This is needed in order to get an adequate start of both cores. Then you can see LED 9 and LED 7 start flashing. The independent operation (and debugging) can easily be checked by setting breakpoints for both cores inside RZ/T1. Breakpoint setting can be performed with a simple double click on the desired program line. Figure 3-13 shows, where the breakpoints have to be set for the respective cores.



for Cortex-R4F

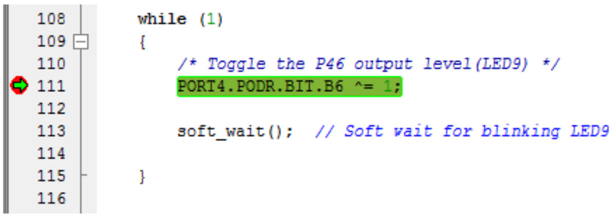


for Cortex-M3

Figure 3-13: Setting breakpoints for the cores inside RZ/T1

Note: Please observe that the breakpoints have to be set in the respective IAR EWARM instances.

After setting the breakpoints, the cores will be stopped and by restarting them individually you can observe how the cores operate the LEDs independent from each other. Figure 3-14 shows the situation on the Cortex-R4F core, when the breakpoint was hit. Then you can restart the core with the ‘Go” button.



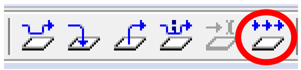


Figure 3-14: Breakpoint hit and processor restart for the Cortex-R4F core

4 Running the EtherCAT Sample Program

In this chapter we will show you how to run the EtherCAT sample program for the RZ/T1 Solution Kit. As a preparation for running the program you have to install the TwinCAT®2 software that serves as a kind of EtherCAT master on the PC and communicates to EtherCAT slaves using the PC's network interface.

4.1 TwinCAT®2 Installation

The TwinCAT®2 installation is straight forward and simple. You can download TwinCAT®2 from

http://www.beckhoff.com/english.asp?twincat/tcatdow.htm.

Please keep in mind that screen shots and operation method may change without notice for newer versions of the software. Above URL will lead you to a screen as shown in Figure 4-1.

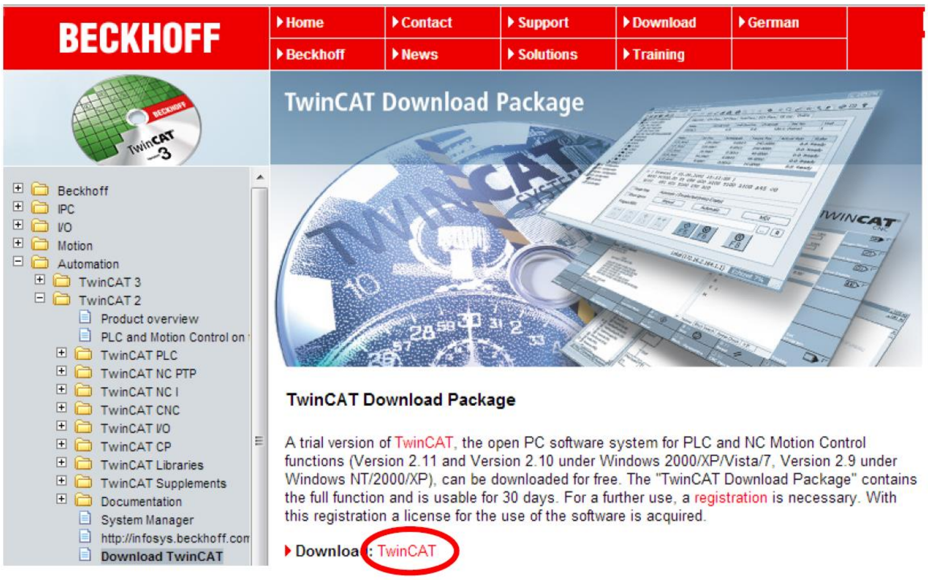


Figure 4-1: TwinCAT®2 Download Screen

Select the TwinCAT®2 download and register as requested. The registration is free of charge. You will then get an e-mail with a download link. After download, TwinCAT®2 can be installed on your PC. The installation procedure guides you through a number on steps that we will not detail here. We will only provide you a few recommendations for the installation:

- The installation procedure asks for a serial number. This field can be left empty.
- In the 'Select Installation Level' dialog select TwinCAT PLC – IEC 61131-3 PLC system
- In the 'Select Installation Type' dialog select 30 days demo version (the functions that are required to run our example last longer than 30 days)
- Install all features
- Target directory for the installation should be C:\TwinCAT

After installation some points should be checked in order to control whether the installation has completed properly.

4.2 Trouble shooting

Check correct network configuration: In the Control Panel go to ‘Network and Internet’ and then into ‘Network Connection’. Select your LAN adapter, right click and go to ‘Properties’. Then check if the TwinCAT® driver appears in the properties window and unselect the rest of the options as shown in Figure 4-2. If the TwinCAT® driver does not appear, you will have to install it manually.

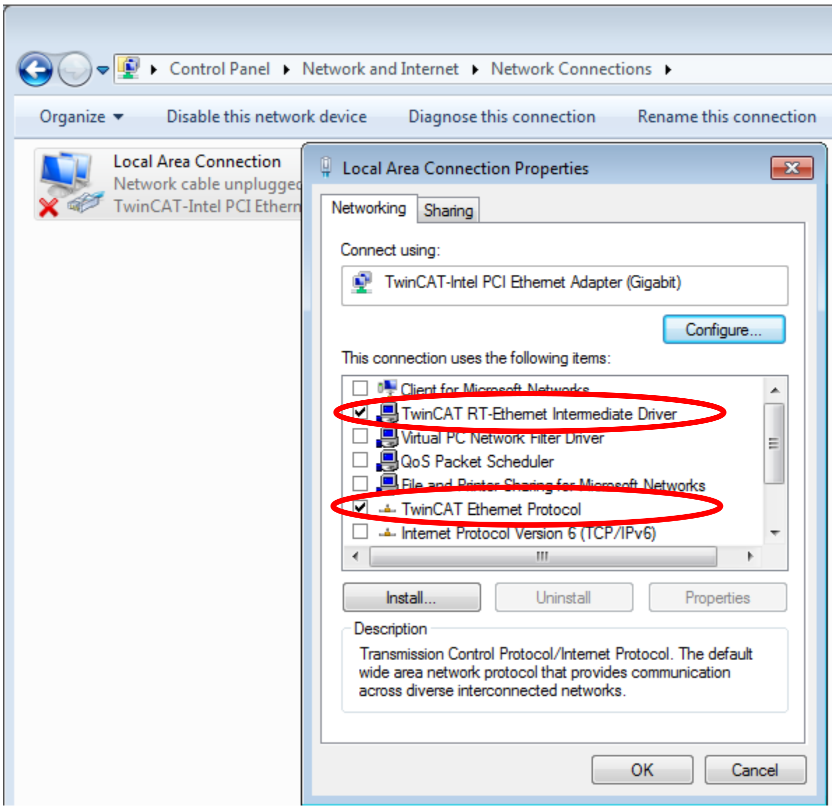


Figure 4-2: Required LAN connection properties

Another potential source of problems is the MS-Windows version. Check whether you have an MS-Windows 32-bits system type as shown in Figure 4-3.

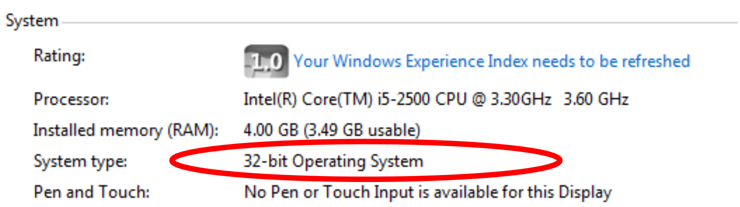


Figure 4-3: MS-Windows system type display

The TwinCAT® driver requires Intel PRO 100 compliant Ethernet adapters. TwinCAT® has an integrated compliance check routine that is described below. When you have started TwinCAT®2, select ‘Options’ and ‘Real Time Ethernet Compatible Devices…’ as illustrated below.

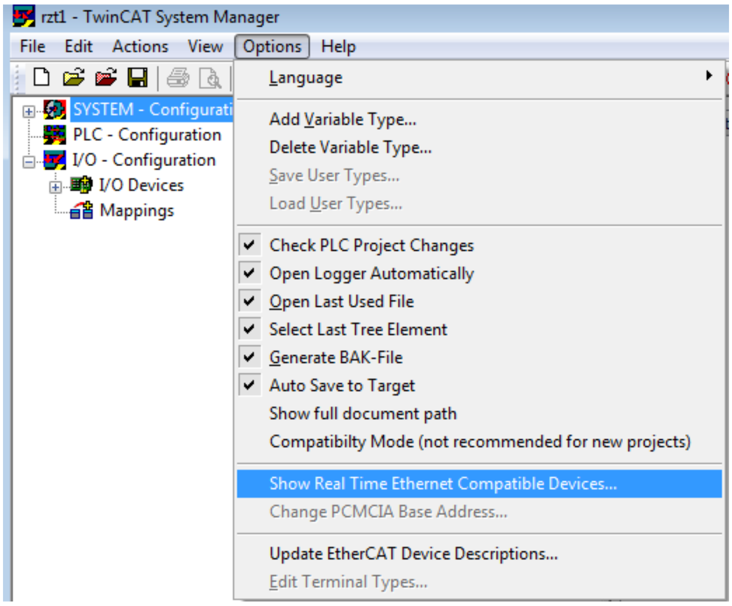


Figure 4-4: Activating TwinCAT built-in compliance test

In the next screen shot (Figure 4-5) you can see the driver state for the Ethernet adapters. It could be possible that there are no compatible devices in the computer. In this case you will have to install an additional Ethernet adapter and re-run this check.

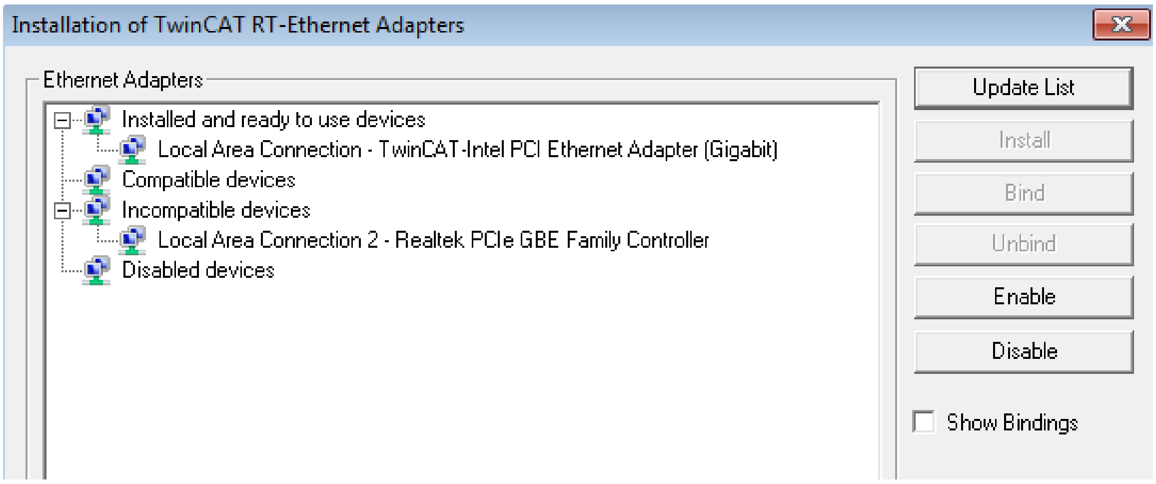


Figure 4-5: List of Ethernet adapters usable for TwinCAT®2

Then it may as well happen that the new adapter is listed as compatible (but not ready to use) device in the dialog from Figure 4-5. In this case select the new adapter and click ‘Install’ to install the missing TwinCAT® RT Ethernet driver. Afterwards the new adapter should be listed as ‘Installed and ready to use’.

Generally please make sure that you run TwinCAT®2 as administrator as shown below.

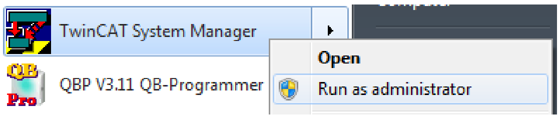


Figure 4-6: Running TwinCAT®2 as administrator

4.3 Generating the EtherCAT Slave Stack Code

To run the EtherCAT communication you must generate the EtherCAT slave code using a special code generation tool provided by Beckhoff. The slave code generation is performed based on the formal description of the slave properties in a so-called ESI file. The ESI code is part of the EtherCAT sample project on the CD and can be found under

…\RZT1_EtherCAT_sample_DualCore_with_HWRTOS_20160212\workspace\iccarm\Cortex-M3\Device\Renesas\RIN_Engine\Source\Project_Dual\EtherCAT_SSC_DC\RenesasSDK\ESI_File)

Copy the file RZT1-R EtherCAT demo [DC].xml into the Io\EtherCAT subdirectory of the TwinCAT® tool as shown in Figure 4-7.

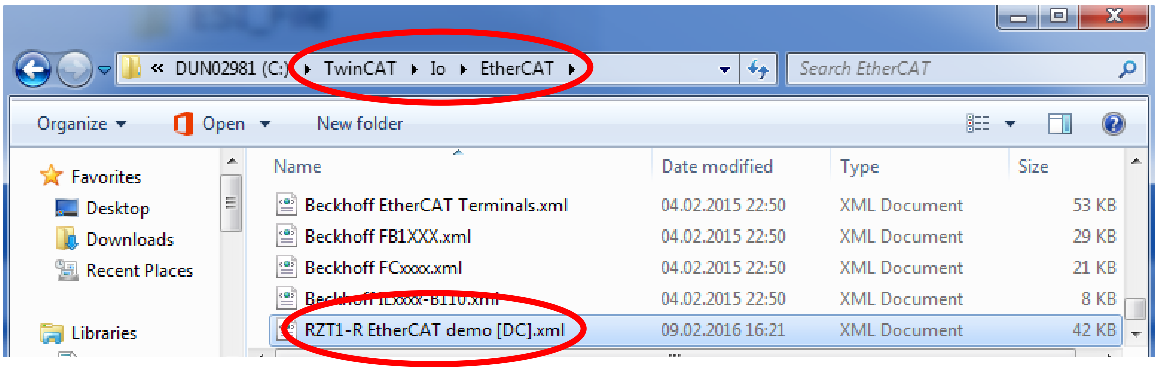


Figure 4-7: Copying the example ESI file into the TwinCAT® structure

Now you have to download the SSC Tool from the Beckhoff website – for example

https://www.ethercat.org/de/products/54FA3235E29643BC805BDD807DF199DE.htm

shown in *Figure 4-8*. You will be requested for login information for downloading the SSC. Please apply for ETG membership first – the membership is free of charge.

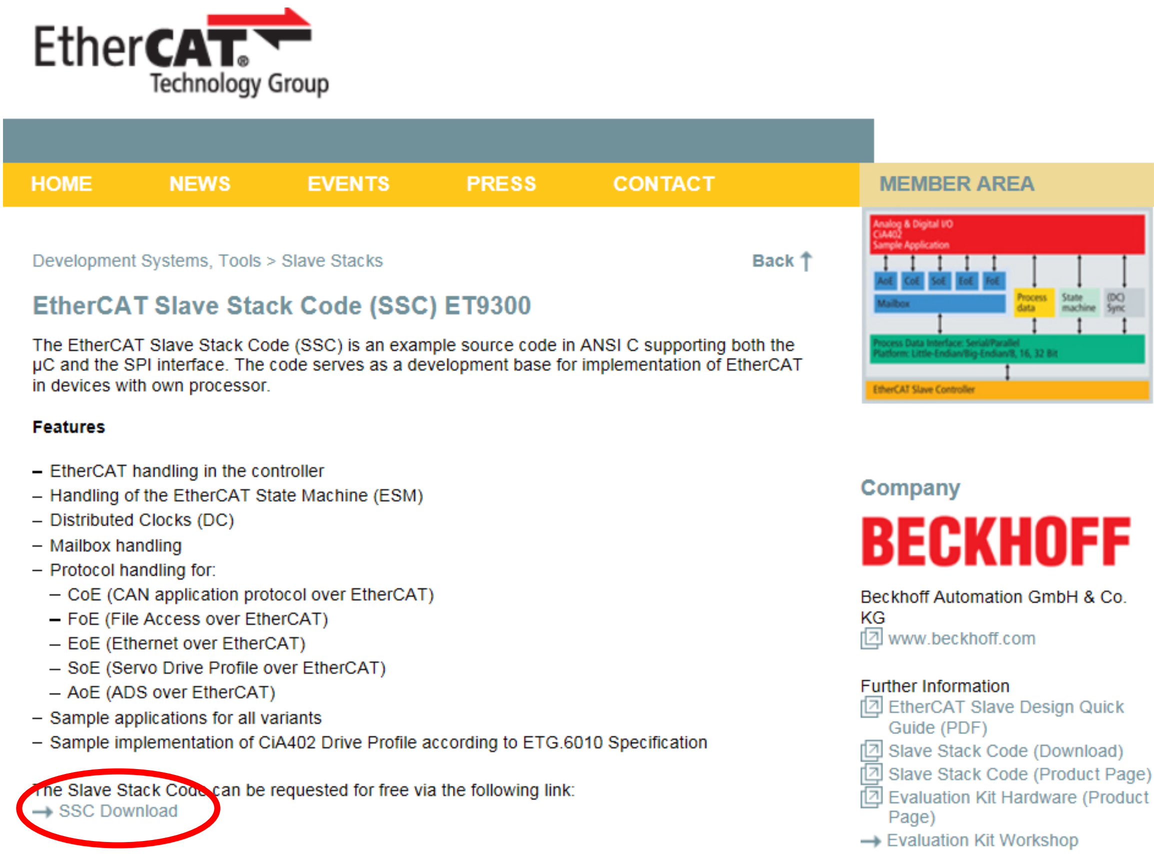


Figure 4-8: Downloading the EtherCAT slave code

Once the SSC is installed, go to

$Project_Directory\EtherCAT\RZT1_EtherCAT_sample_DualCore_with_HWRTOS_20160212\workspace\iccarm\Cortex-M3\Device\Renesas\RIN_Engine\Source\Project_Dual\EtherCAT_SSC_DC\RenesasSDK\ssc_project

and doubleclick to the .esp file (*Figure 4-9*). This will open the SSC tool.

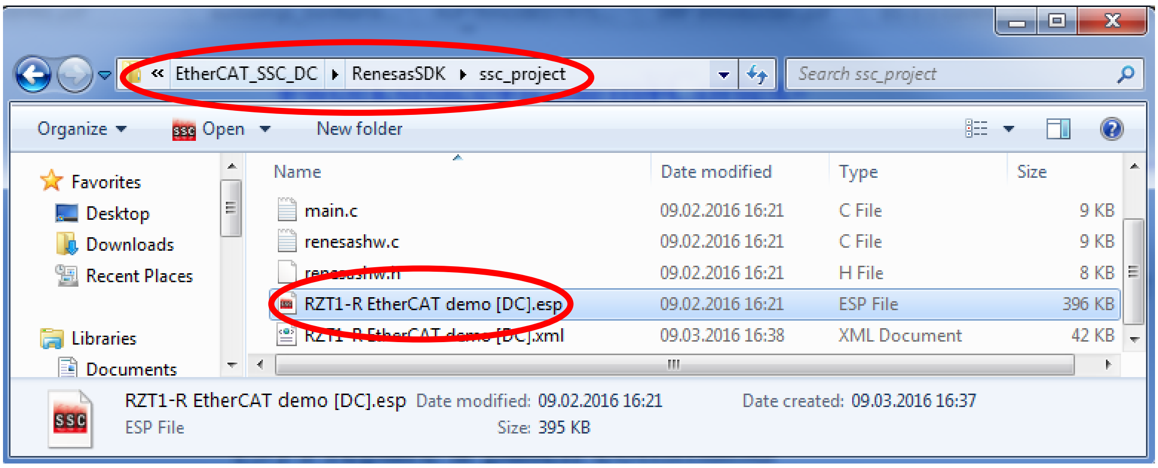


Figure 4-9: Opening the SSC tool

In the SSC tool you can now select 'Create new Slave Files' from the 'Project' menu (*Figure 4-10*) and once you have clicked 'Start', the SSC tool will start generating the slave source code for the RZ/T1 processor. Afterwards the tool can be closed.

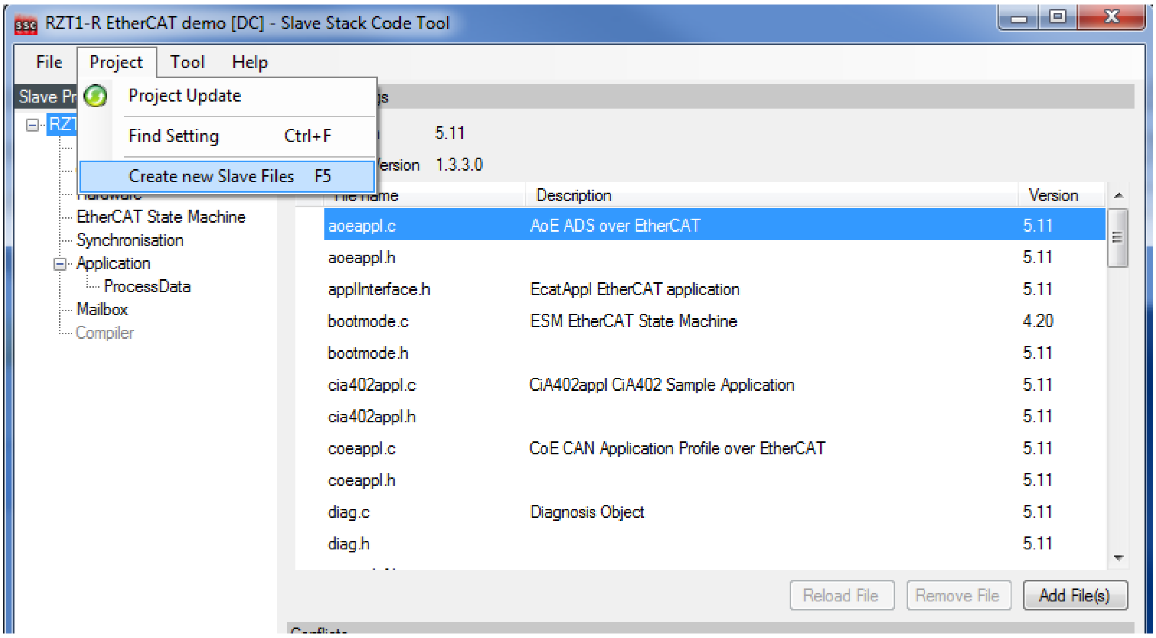


Figure 4-10: Creating the EtherCAT slave stack code

Last but not least it is required to install a patch file on your computer and to make it usable as an environment variable. Please download the GNU Patch program (version: 2.5.9 or later) from the following website:

http://gnuwin32.sourceforge.net/packages/patch.htm

Then, store the patch.exe file and add to storage path to an environmental variable: In the Windows Control Panel go to ‘System’ and then to ‘Advanced system settings’. In the ‘System Properties’ dialog select ‘Environment Variables’ as illustrated in *Figure 4-11*.

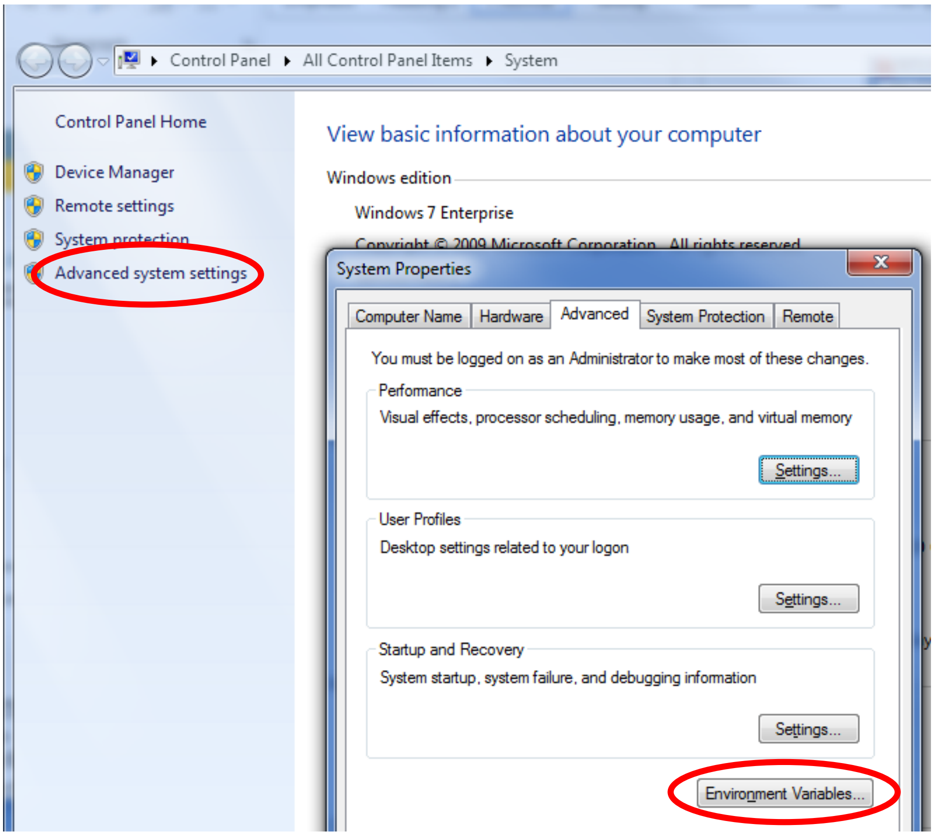


Figure 4-11: Adding a new environment variable

Select ‘Path’ in the list of system variables and then ‘Edit’. Add the path, where you have stored the patch file (C\Program Files (x86)\GnuWin32\bin), to the already existing search path. Click ‘Ok’ twice to accept the changes.

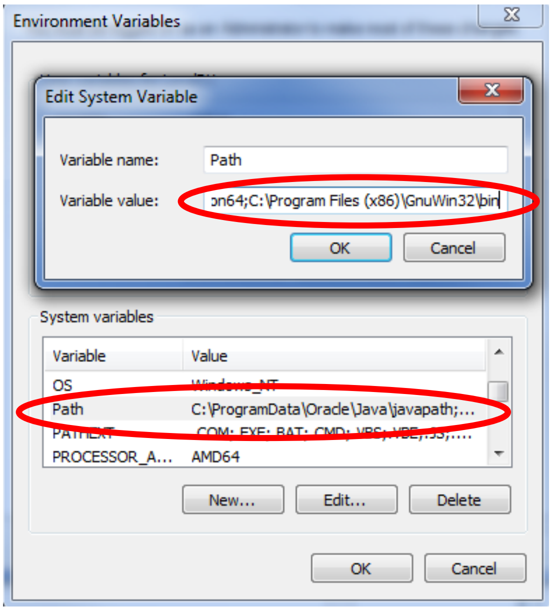


Figure 4-12: Editing the system variable ‘Path’

Next, the patch has to be executed once. Double-click the ‘apply_patch.bat’file from the Project directory.The script in this file moves the directory that contains the slave stack code, and then applies the patch that makes the corrections for the sample program.

If a "Patching file ..." message similar to *Figure 4-13* does not appear, the patch is not applied. In this case, right-click "apply_patch.bat", and then select "Run as administrator".

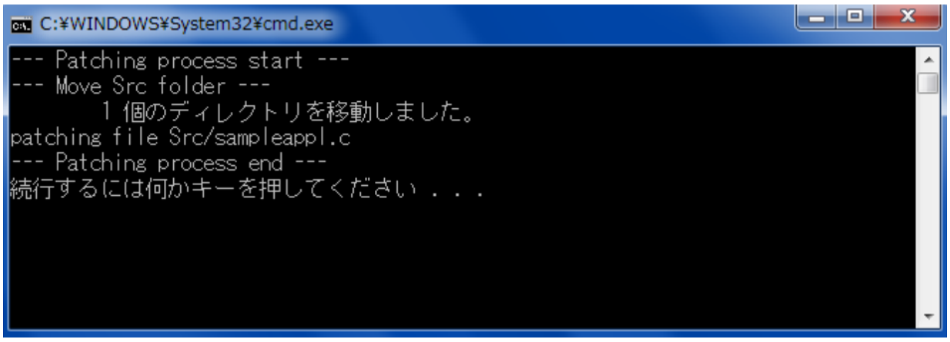


Figure 4-13: Patching file messages

Now the preparation for running the EtherCAT project in IAR EWARM are finally completed.

With the last steps, you have generated the files that the M3 Core Project needs to compile. So please open the M3 Project by doubleclicking the main.eww file in the

$Project_Directory\EtherCAT\RZT1_EtherCAT_sample_DualCore_with_HWRTOS_20160212\workspace\iccarm\CortexM3\Device\Renesas\RIN_Engine\Source\Project_Dual\EtherCAT_SSC_DC

directory and build the project. This will generate the binary file for the Cortex-M3 core.

Figure 4-14 explains how the Coretx-M3 binary file is included in the Cortex-R4F project.

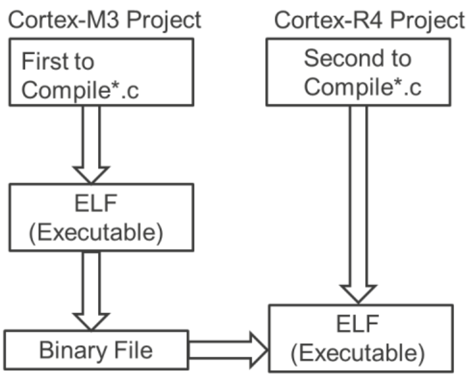


Figure 4-14: Integrating Cortex-M3 code into a Cortex-R4 project

After that, open the Cortex-R4 core Project by double-clicking the workspace file ‘RZ_T1_shm_serial_boot.eww’ in the

$Project_Directory\EtherCAT\RZT1_EtherCAT_sample_DualCore_with_HWRTOS_20160212\workspace\iccarm\Cortex-R4\EtherCAT_SSC_DC

directory, compile it, download it to the target and start program execution.

Now we will explain how to configure TwinCAT® in order to be to control the EtherCAT slave device realized in the RZ/T1.

For that reason, open TwinCAT®, select ‘I/O Devices’ with a right mouse click and then select ‘Scan Devices’ as shown in *Figure 4-15*. TwinCAT® will now look for EtherCAT devices connected to your PC’s network port.

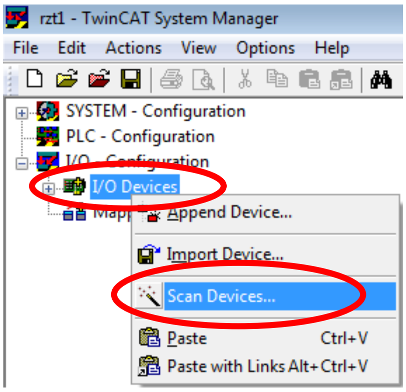


Figure 4-15: select ‘Scan Devices’ function

When TwinCAT® finds a new device you will be informed accordingly with the subsequent dialog box. Select the new device and click ‘Ok’. TwinCAT® will now ask you whether it should ‘Scan for boxes’. Click ‘Yes’ and accept the subsequent warning (if they occur at all) with ‘Ok’.

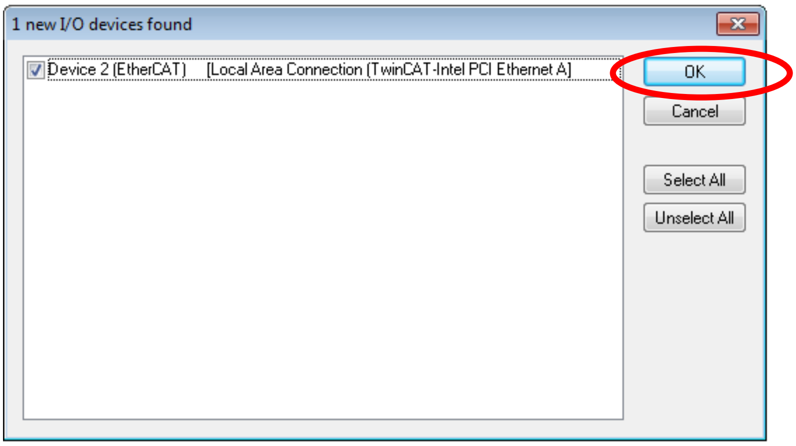


Figure 4-16: Found devices dialog

Finally select ‘Yes’ to activate the ‘Free Run’ option as shown in *Figure 4-17*.

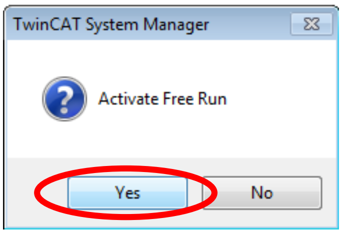


Figure 4-17: Activating the ‘Free Run’ option

An EtherCAT slave device holds identification data in an EEPROM. Before proper identification is possible, this EEPROM must be initialized once. The next screen shots will guide you through the EEPROM initialization.

In the TwinCAT® System Manager double-click on the new ‘box’ that was found in the recent ‘Scan for boxes’ process. TwinCAT® will now offer you several tabs in a new window; select ‘EtherCAT’ and then ‘Advanced Settings’ as illustrated in *Figure 4-18*.

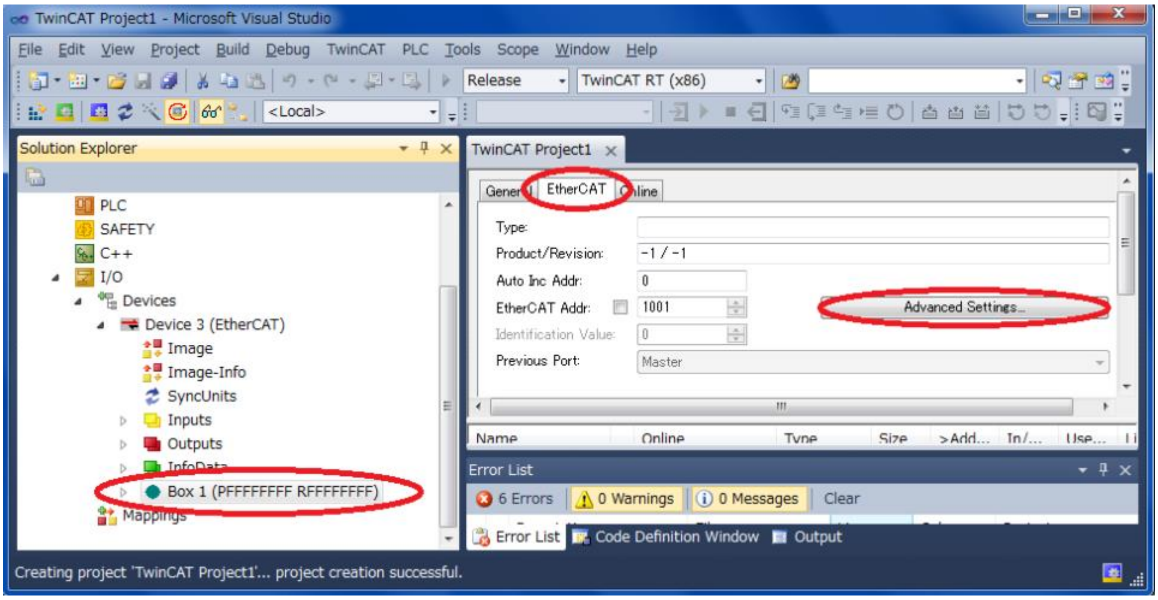


Figure 4-18: Preparation for EEPROM initialization

After arriving at the 'Advanced Settings' dialog, expand 'ESC Access' and then 'E²PROM'; now select the 'Hex Editor' and 'Download from List…' as shown in *Figure 4-19*.

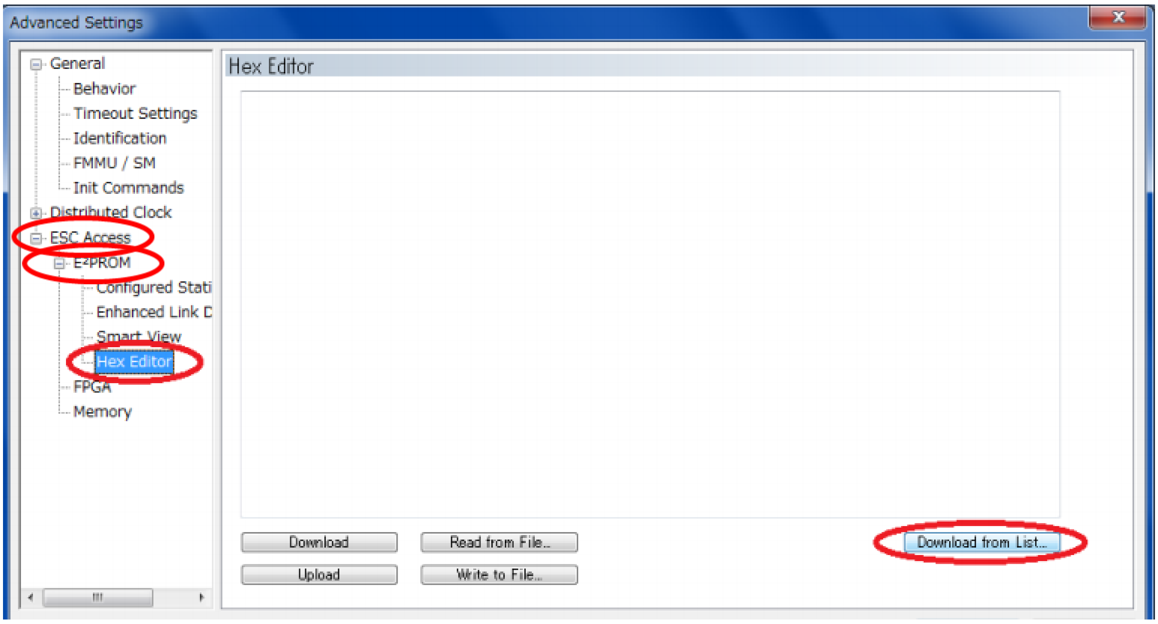


Figure 4-19: Further preparation for EEPROM initialization

You will now be offered a couple of possible download options. From the options select the 'RZ/T1-R EtherCAT demo [DC]' and click the 'OK'.

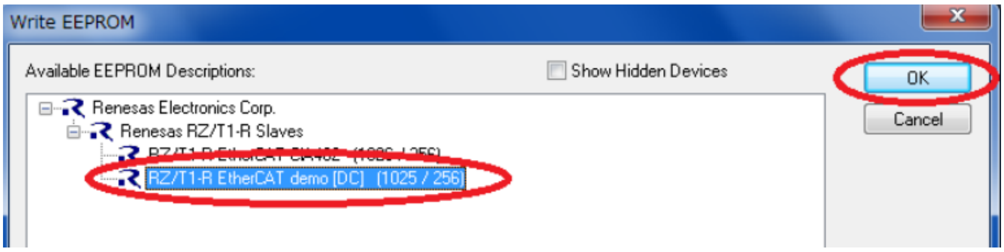


Figure 4-20: Selecting the proper download option

The selected EEPROM content is now loaded into the HEX editor and you must click ‘Upload’ to transfer the data into the EEPROM as shown in *Figure 4-21*. The EEPROM is now ready to go.

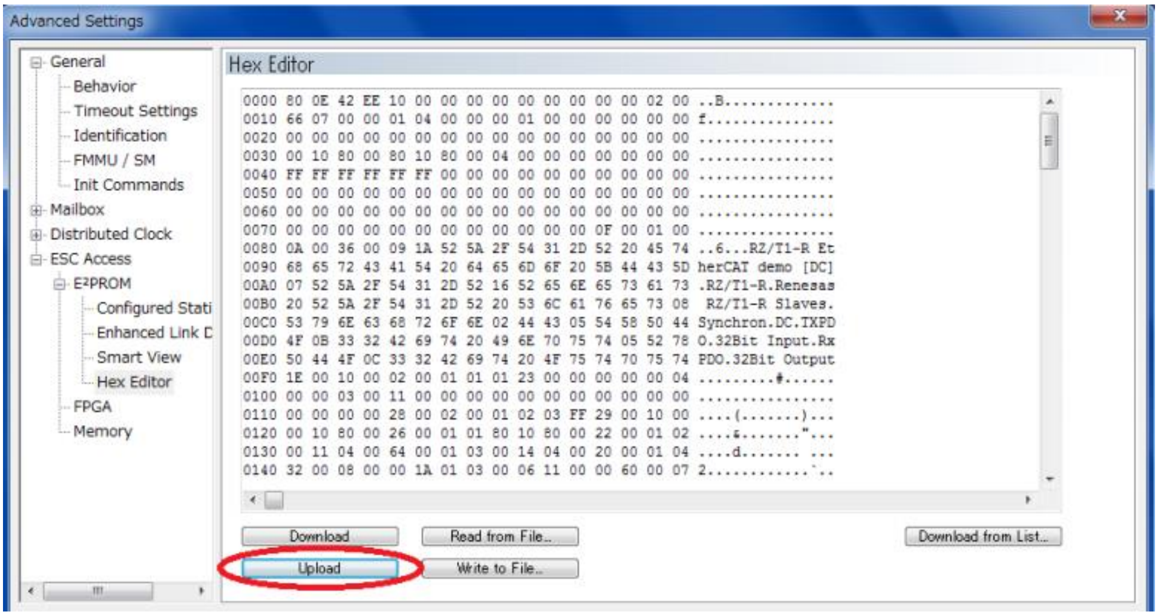


Figure 4-21: Uploading the EEPROM data

Now the ‘Scan Devices’ process must be repeated to show you the new devices with the proper identification data. Transfer the new box to the list of configured items with ‘Copy All’ and click ‘Ok’ as shown in Figure 4-22 and *Figure 4-23*.

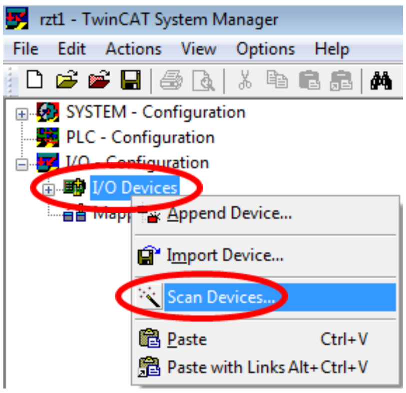


Figure 4-22: Execute Scan Devices again

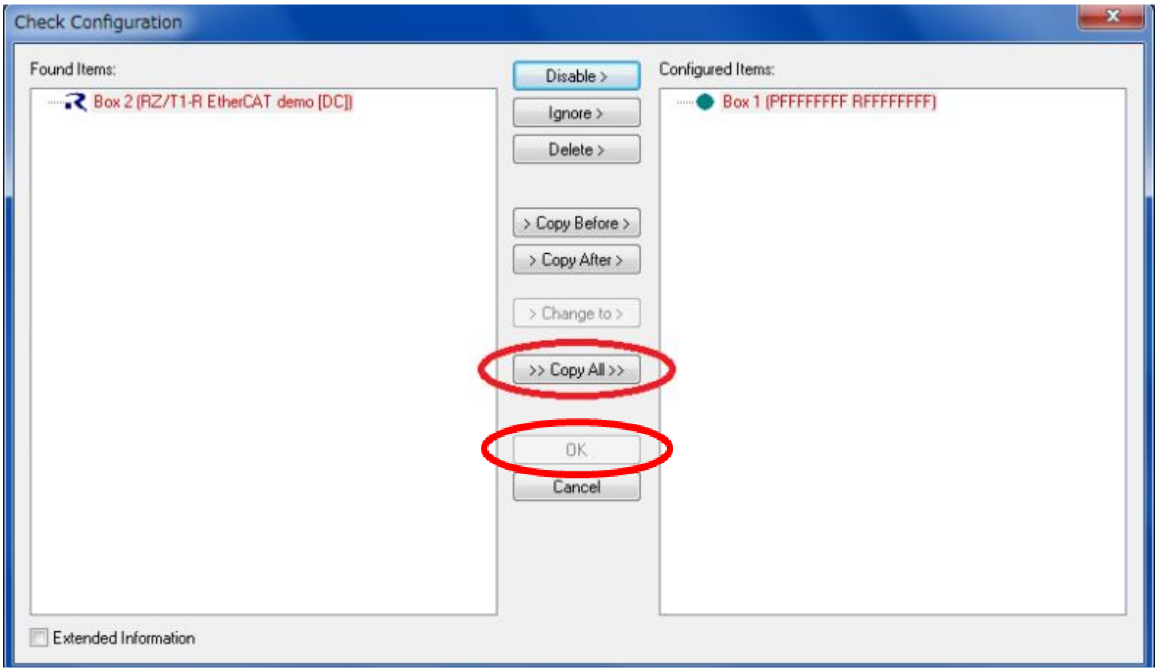


Figure 4-23: Transferring the new box to the actual network

The new box is now displayed in the directory style network representation of the TwinCAT® System Manager. When you select the new box, you must check on the 'Online' tab, whether its status is 'OP' (i.e. operational) as illustrated in *Figure 4-24*.

If this is not the case, select 'Set/Reset TwinCAT to Config Mode' from the 'Actions' menu and select once again 'Yes' to load I/O devices and 'Yes' to Active Free Run.

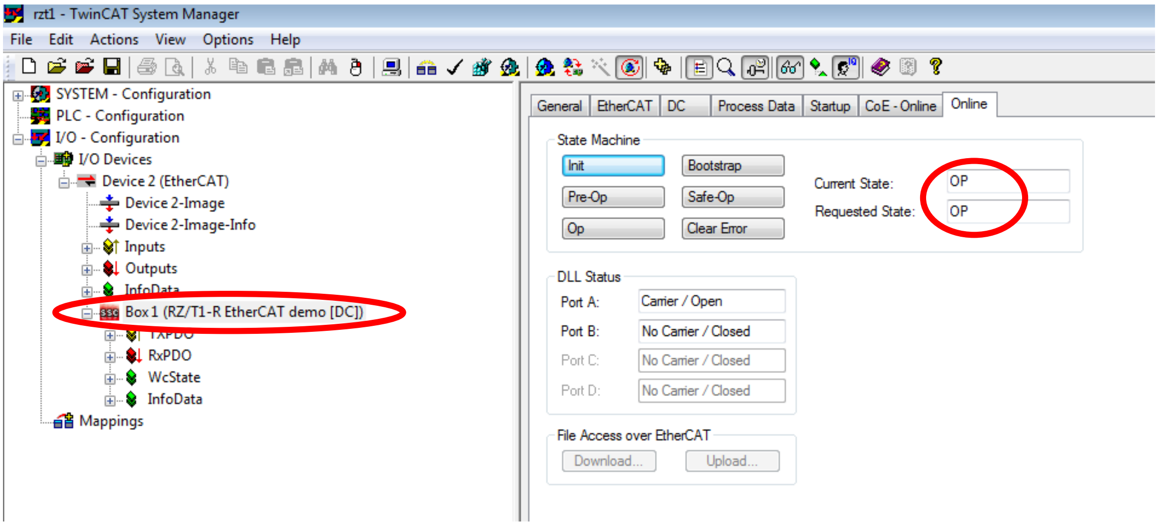


Figure 4-24: Check new box status

The EtherCAT network is now working and you can exchange data between TwinCAT® running on the PC and the EtherCAT slave running on the RZ/T1. This can easily be visualized by modifying the data that is output to the EtherCAT slave.

In TwinCAT® expand the box that represents the RZ/T1 based slave, expand process data output ‘RxPDO’ and select ‘32Bit Output’. On the ‘Online’ tab click ‘Write’ and enter a 4-bit value in the ‘Set Value Dialog’ as shown in the subsequent figures.

If everything is correct, then it is possible to modify the output to the Slave device by dual click in the RxPDO-> 32 bits Output and then modifying the value

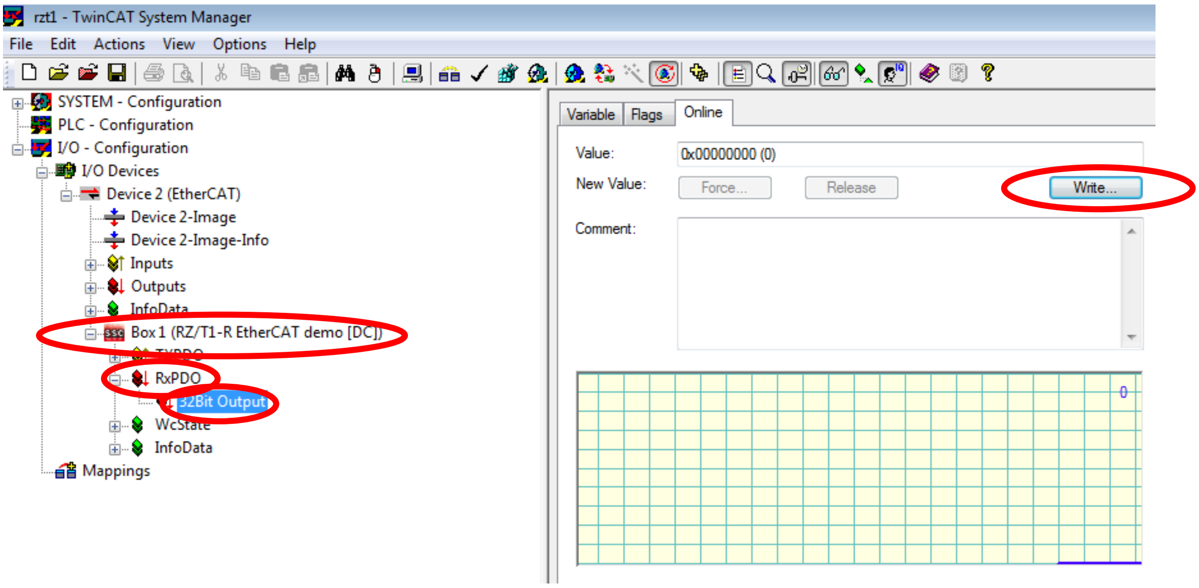


Figure 4-25: Writing data to the EtherCAT slave device

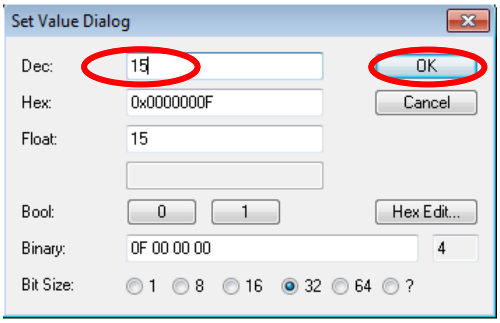


Figure 4-26: Setting the value to be written to the EtherCAT slave device

In case that you write 15 as shown in *Figure 4-26*, LEDs 3 to 6 on the Solution Kit board will light up (*Figure 4-27*). If you change the value to 5 (*Figure 4-28*), only LEDs 3 and 5 will light up.

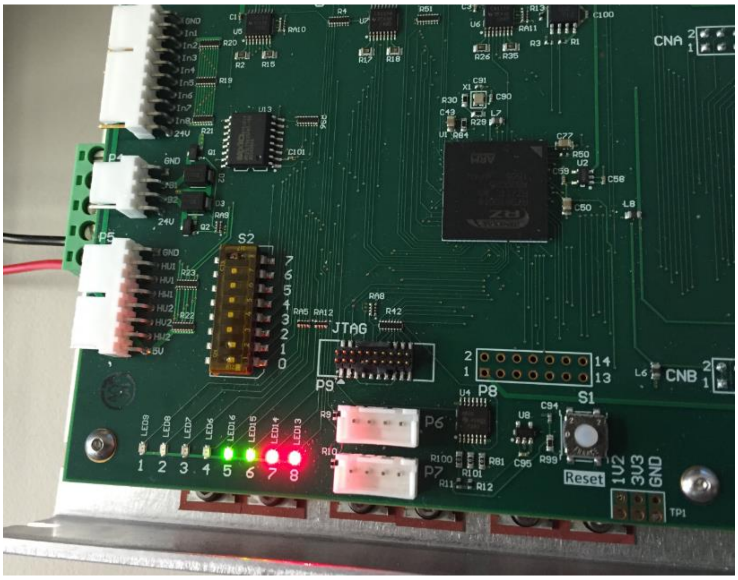


Figure 4-27: display of process data ‘15’

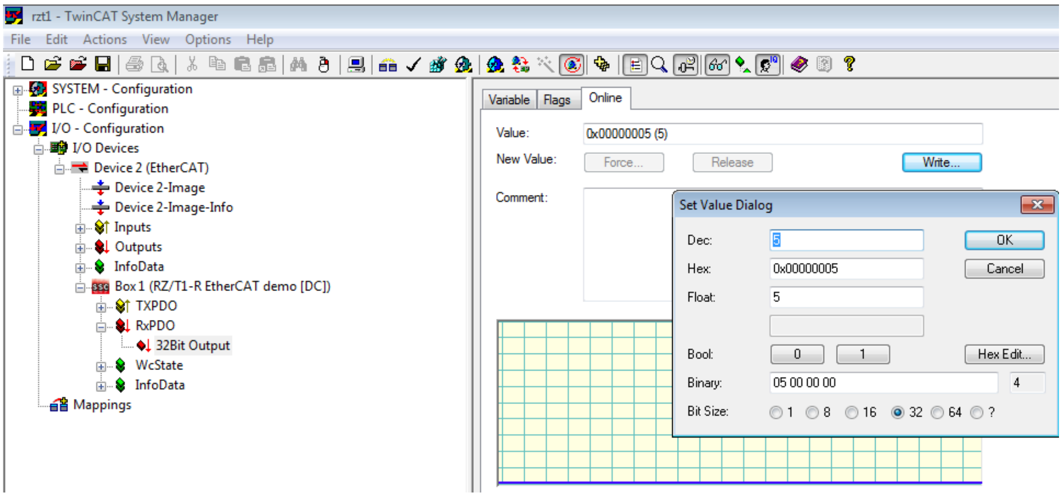


Figure 4-28: Changing process data to ‘5’

5 How to Return to Delivery Status

At some point of time you may find it useful to bring the RZ/T1 Solution kit board back to delivery status. Therefore this chapter describes how you write the motion control firmware back into the serial Flash on the board.

The RZ/T1 Solution Kit board is delivered with the motion control firmware pre-installed. When you execute the examples described in this document and when you execute these examples from serial Flash, the motion control firmware is overwritten. To return to the original status just connect the IAR I-jet Lite, as if you wanted to debug a program, open IAR with an arbitrary project. Then select ‘Download’ from the ‘Project’ menu and then ‘Download File’ as shown in Figure 5-1.

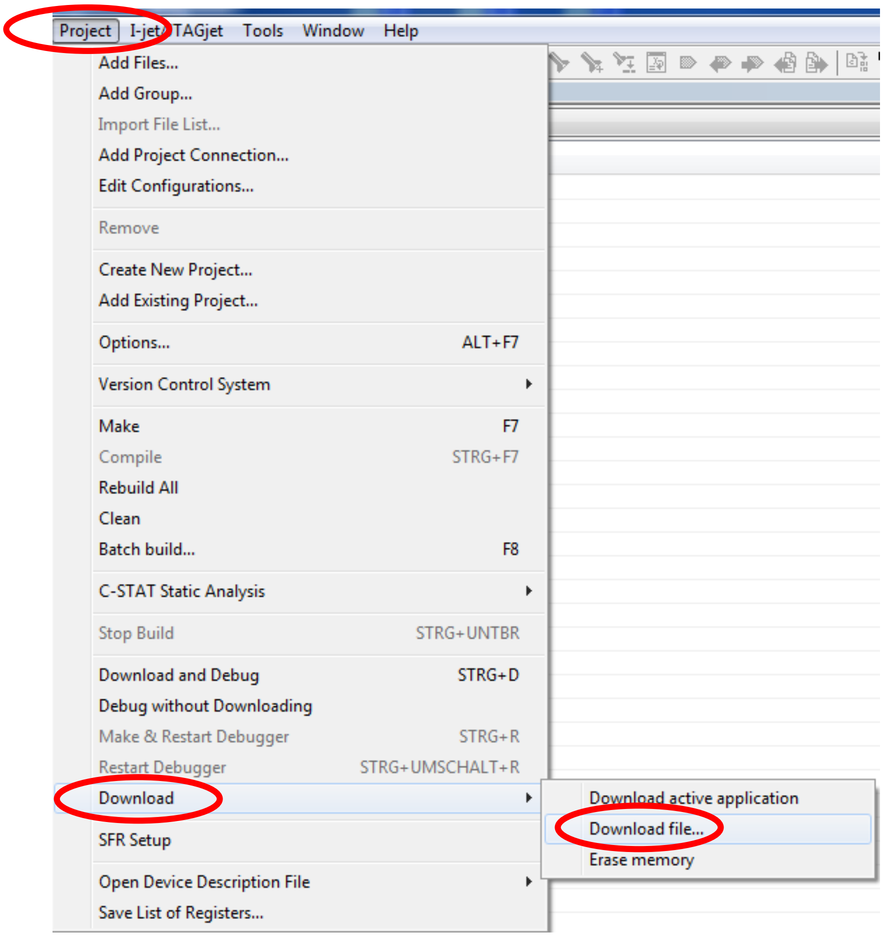


Figure 5-1: Selecting ‘Download File’ function

Now a file open dialog will pop up. In this dialog select the file ‘Motion_Control.out’ from the YDRIVE-IT-RZT1 CD.

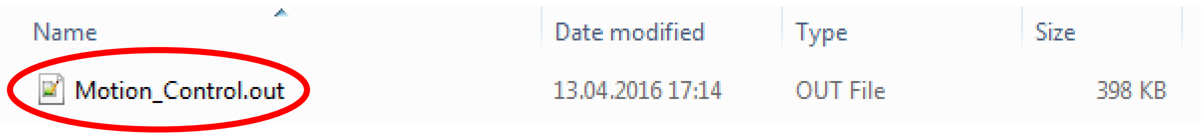
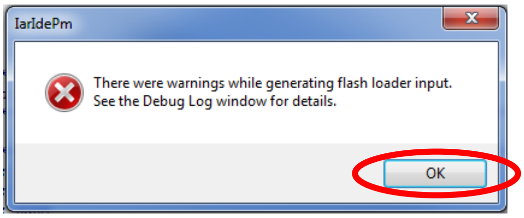


Figure 5-2: Selecting a file for download

You will now see an error message that you can ignore. After clicking ‘Ok’ downloading and Flashing will start as shown in Figure 5-3.



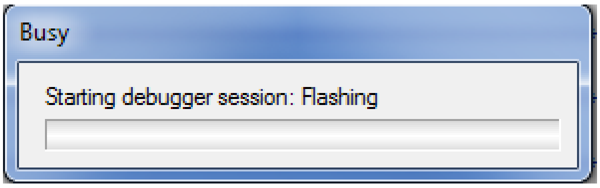


Figure 5-3: Start flashing a program

You can execute an easy sanity check as follows:

- Power off the board
- Unplug the IAR I-jet Lite
- Re-apply power to the board

Now LEDs LED9, LED7 and LED6 will be on and LED8 will start blinking and this is an indication that the motion control firmware has taken over control again.

Revision History

		Description	
Rev.	**Date**	**Page**	**Summary**
Rev. 1.0	May 4th, 2016		1st release
Rev. 1.01	June 10th, 2016		Changed .pdf security settings

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Handling of Unused Pins

Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.

- The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

The state of the product is undefined at the moment when power is supplied.

- The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
  In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.
  In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

Access to reserved addresses is prohibited.

- The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

After applying a reset, only release the reset line after the operating clock signal has become stable.
When switching the clock signal during program execution, wait until the target clock signal has stabilized.

- When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

- The characteristics of an MPU or MCU in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 5.0