VxWorks 7®
Documentation
Board Support Package
For Acromag® XCOM-6400 Products

8501076D

Table of Contents

1.0   Introduction

1.1   Overview

The VxWorks 7® Board Support Package (BSP) is a comprehensive package that simplifies the process of developing application for Acromag for the XCOM-6400 series hardware using VxWorks Work Bench.

The VxWorks 7® XCOM-6400 BSP has the following components:

- Process to develop a VxWorks 7 Source Build Project
- Process to develop a VxWorks 7 Image Project
- Process to build a VxWorks 7 Library
- Process to build a Real Time Process Project
- C Source code examples:

    o   Test Menu
    o   Initialize network ports
    o   Display network ports
    o   Read/Write to serial port
    o   Read Devices in the SmBus
    o   Read/Write to GPIO
    o   Read NCT6776 Super I/O
    o   Read/Write to USB stick

1.2   Provided Files

There are several folders provided by the VxWorks 7® BSP, all of the pertinent code files being inside the VxW7_64bit folder on the distribution media:

- 002_KernelIncludeFiles

    These are the files to be included in the Kernel Image Project, and to be built inside the Kernel.

- 003_DKMIncludeFiles

    These are API files and functions that give examples on how to access XCOM series hardware features such as GPIO access, SMBus access, Super I/O access, network port initialization, serial port access, etc. These files are now included in their own Downloadable Kernel Module (DKM), to be loaded at runtime on the target machine.

1.3   Preface

The information contained in this manual is subject to change without notice, and Acromag, Inc. (Acromag) does not guarantee its accuracy. Acromag

makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Further, Acromag assumes no responsibility for any errors that may appear in this manual and makes no commitment to update, or keep current, the information contained in this manual. No part of this manual may be copied or reproduced in any form, without the prior written consent of Acromag,

1.4   Trademarks, Trade Name, and Copyright Information

2.0   Build a VxWorks 7 Source Build Project

2.1 Launch New Project Wizard

Start VxWorks 7 Workbench.

Click <File><New><Project>.
Select <VxWorks 7> <VxWorks Source Build Project> as shown below.
Then click <Next>.
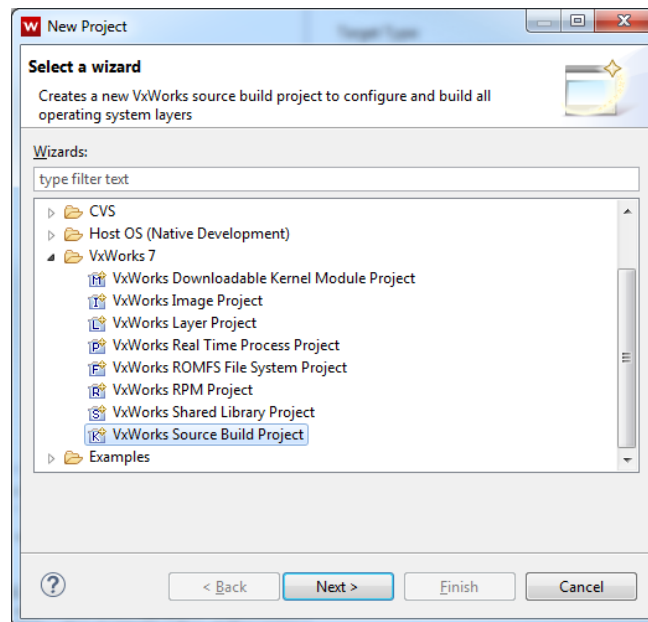
2.2   Assign a Project Name

It will then prompt you for a Project name.
Type *Intel64_Vx7_1_1_0_2* or whatever name suits you as shown below.
Then click <Next>
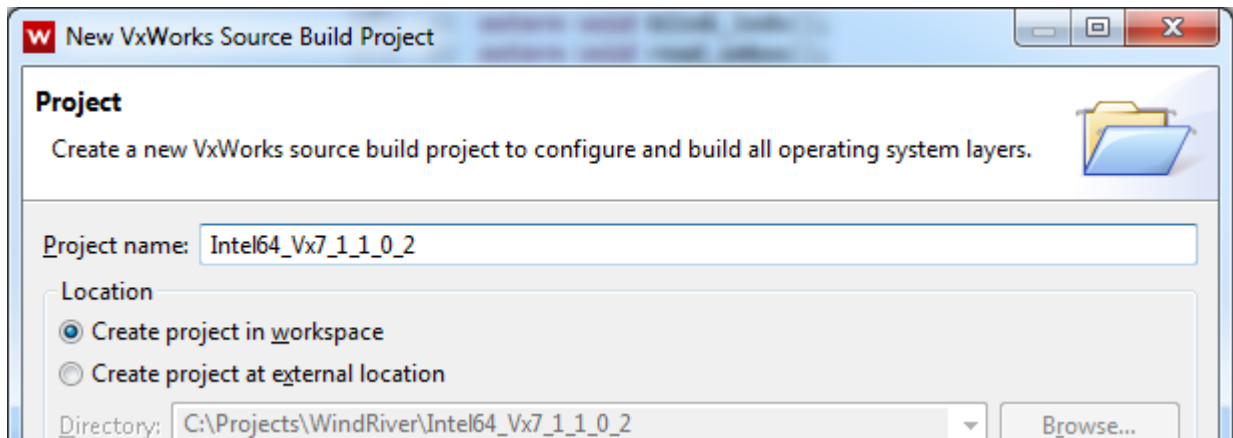
2.3 Select the appropriate Intel BSP

Based on: a VxWorks 7 board support package
BSP: itl_generic (latest version)
Active CPU: CORE
Baseline: (Nothing checked)
Address Mode: LP64 64-bit libraries
Compiler Support: Primary and secondary compiler
Floating point setting: hard
Endianess setting: little
VxWorks 6.9 compatibility: Disabled for this BSP
Processor mode: UP support in Libraries
Debug mode: Off, and normal compiler optimizations enabled
IP version setting: IPv6 and IPv4 enabled libraries
VSB profile: None

Click on <Finish>.

2.4 Confirm creation of the Source Build project

This will then add the newly created Source Build project in your Project explorer as shown below.
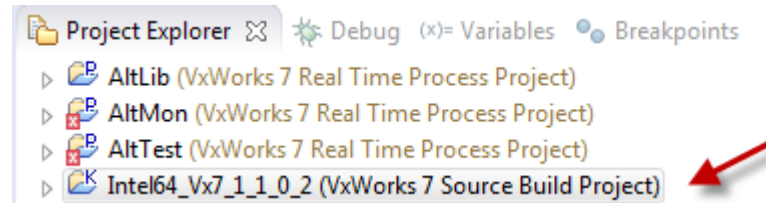
2.5 Enable Symmetric Multiprocessor Support (SMP)

Double click on the Source Build Configuration and locate the option for SMP and double click to enable as shown below.
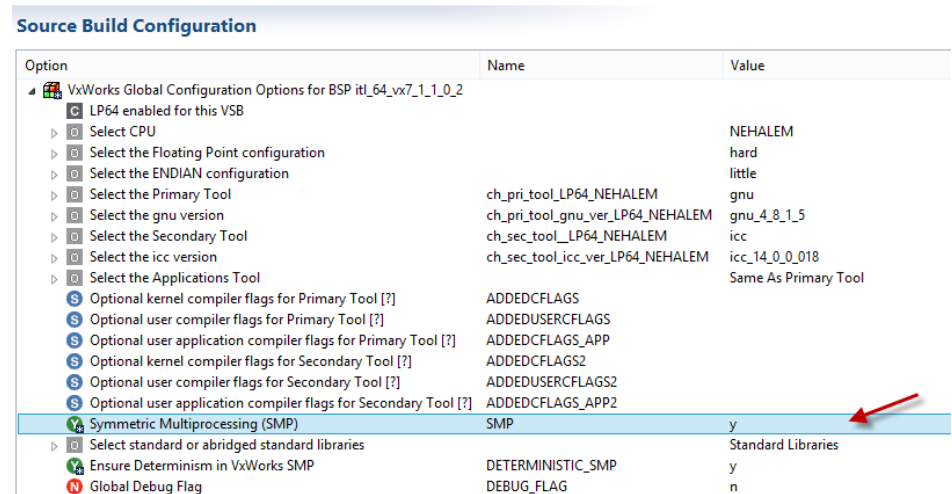
2.6 Build the Project

Right click on the Project name and select <Build Project>.
Grab a cup of coffee, this process may take a while to complete.

When it does finally finish, congratulations!
You have successfully built a VxWorks 7 Source Build project.

3.0 Build a VxWorks 7 Image Project

3.1 Launch New Project Wizard

Within VxWorks Workbench, click <File><New><Wind River Workbench Project>.
Then select <Kernel Image>.
Then click <Next>.
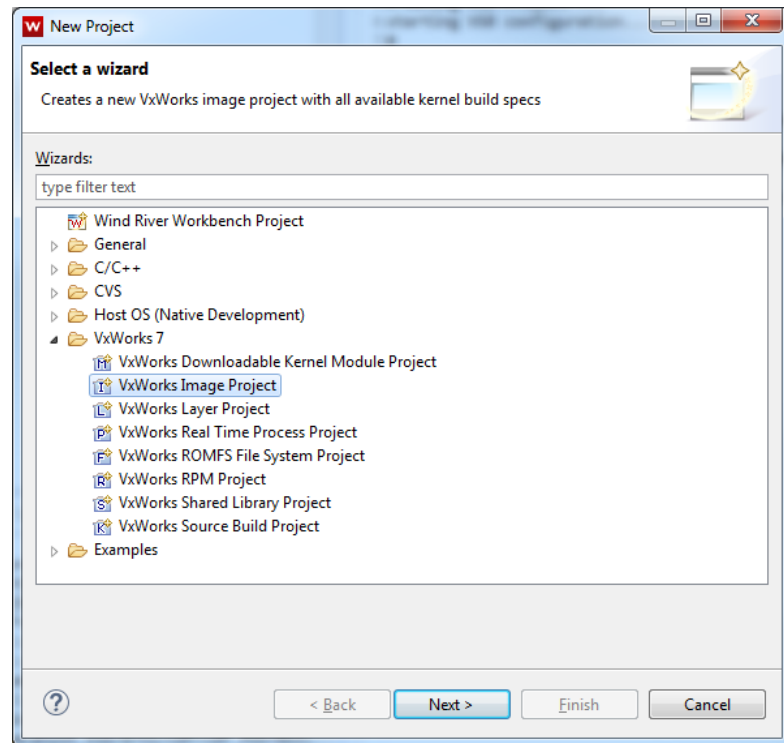
3.2 Assign a Project Name

It will then prompt you for a Project name.
Type *XCom6400_IntelGeneric_Vx7_1_1_0_2* or whatever name suits you as shown below.
Then click <Next>.
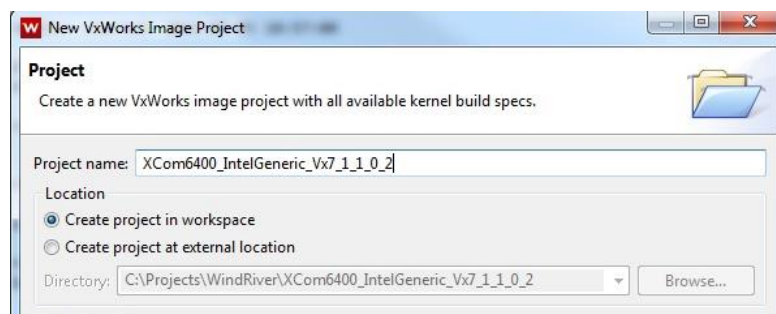
3.3 Select the Source Build Project

On the form below, select Based on to <a source build project>.
Then change the project name to the one you specified in Section 2.3.
Click <Next>.
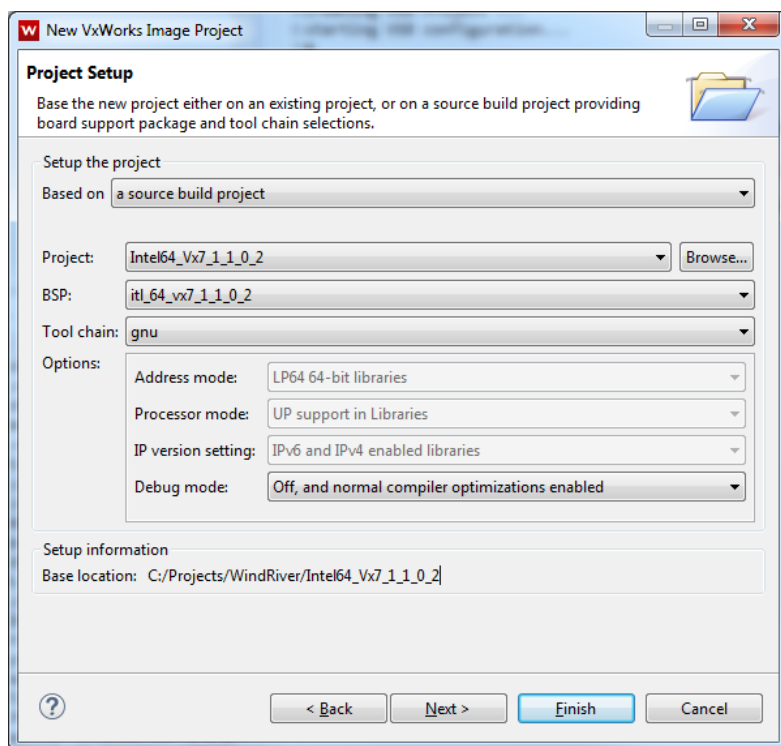
3.4 Select a Profile

For profile, select <PROFILE_INTEL_GENERIC> as shown below.
Then click <Finish>.
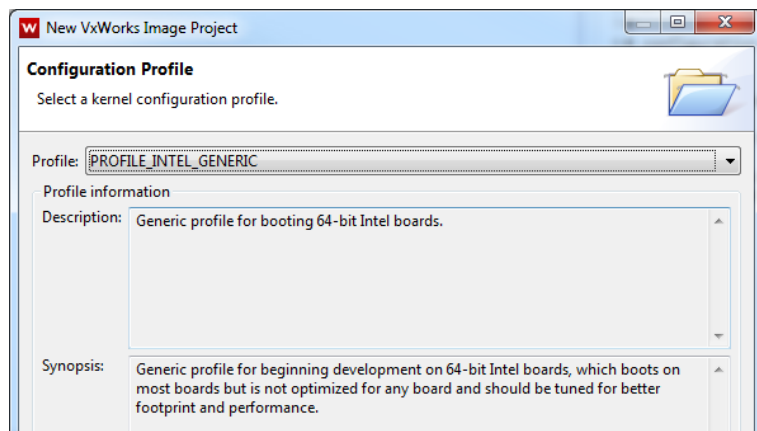
3.5 Confirm creation of Image Project

This will then add the newly created VxWorks 7 Image project in your Project explorer as shown below. Click on the triangle beside it to expand like below.
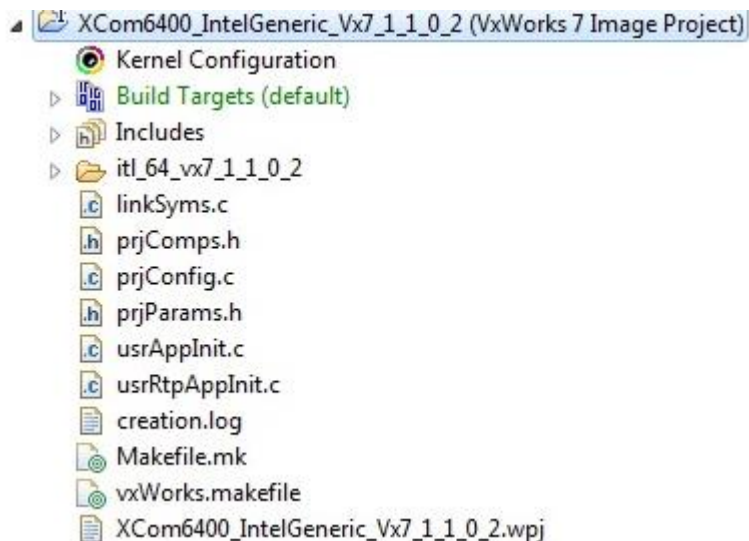
3.6 Change Set Active Build Spec

Right click on the project name and select <Build><Set Active Build Specs><default_romCompress>
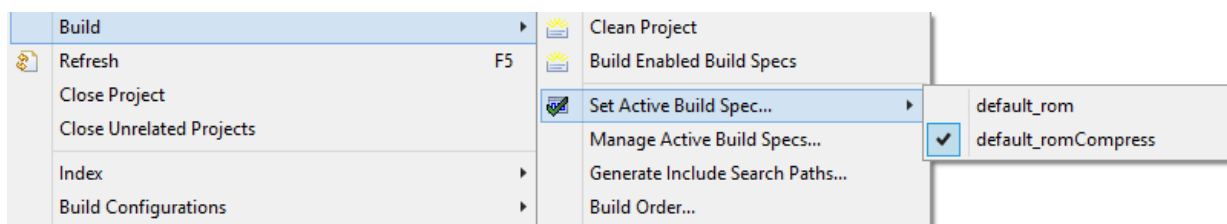
3.7 Edit the Project file

Double click on *XCom6400_IntelGeneric_Vx7_1_1_0_2.wpj* in Project Explorer as shown below.
This will bring up the editor for the project file.
Please note that this name will vary depending on what you used for project name.
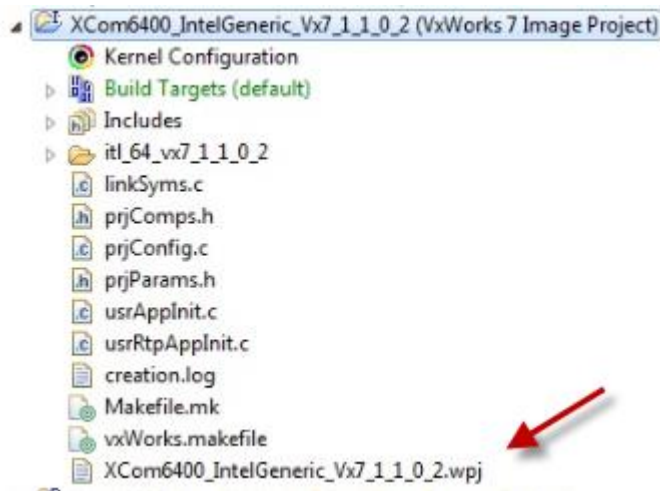
3.8 Modify <ROM_SIZE> and <ROM_TEXT ADRS>

Edit and locate each of the items below within the file and change to the values in red:

```
<BEGIN> BUILD_default_MACRO_ROM_SIZE
00800000 # number of bytes of ROM space
<END>

<BEGIN> BUILD_default_MACRO_ROM_TEXT_ADRS
00408000 # ROM entry address
<END>

<BEGIN> BUILD_default_romCompress_MACRO_ROM_SIZE
00800000 # number of bytes of ROM space
<END>

<BEGIN> BUILD_default_romCompress_MACRO_ROM_TEXT_ADRS
00408000 # ROM entry address
<END>
```

3.9 Save the File

Save the file. It will then prompt you to save the changes. Click <Yes>.

Afterwards, it will then prompt you to load the changes as shown below. Click <Yes> again.
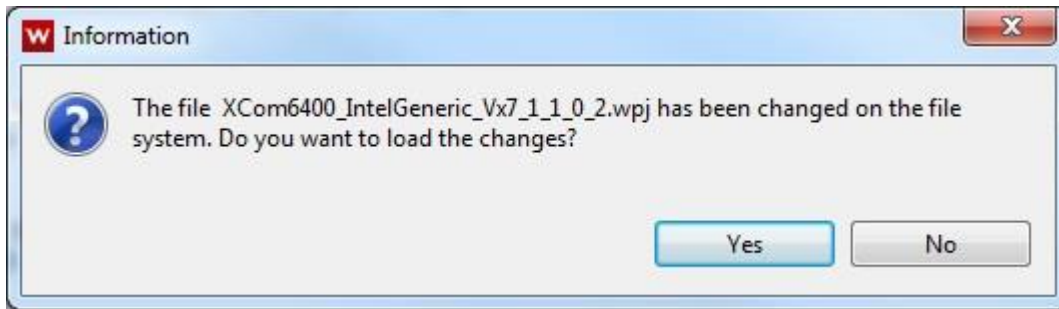
3.10 Edit file <20bsp.cdf>

Click on the triangle to the left of the folder icon for <itl64_vx7_1_1_0_2>.
Double click on *20bsp.cdf* as shown below.
This will bring up the editor for this file.
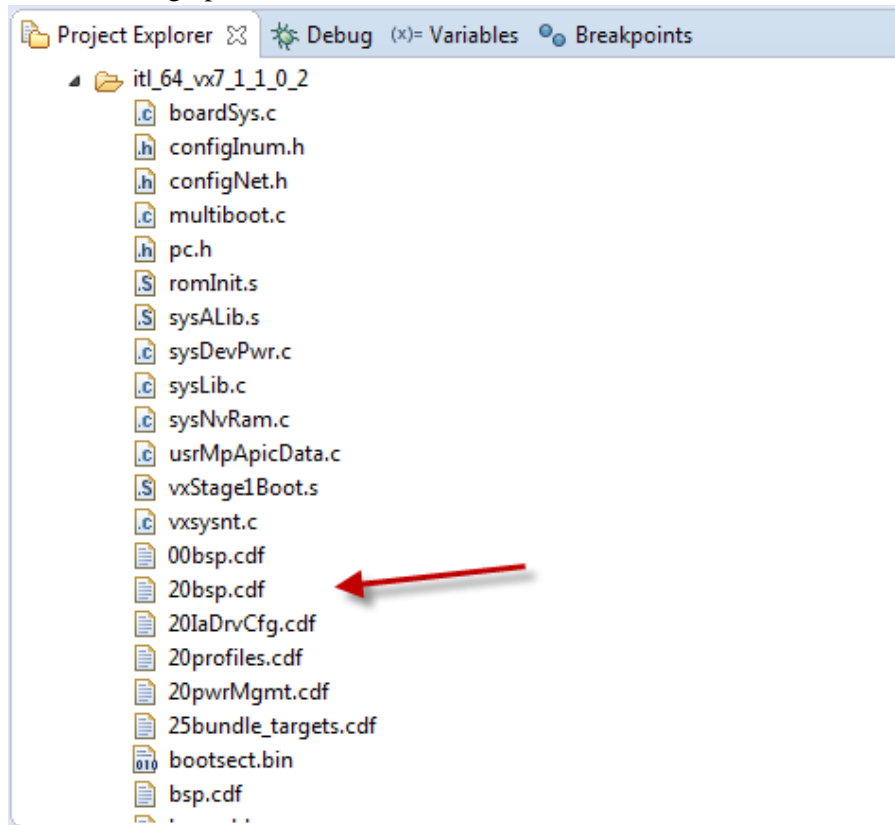
3.11 Modify serial port information in file

Edit and locate each of the items below within the file and change to the values in red. For XCOM-6400 change the default N_UART_CHANNELS to 2. Note that you may have to change the IP addresses in the DEFAULT_BOOT_LINE so they match your network topology.

```
Parameter N_UART_CHANNELS {
    NAME            N_UART_CHANNELS
    SYNOPSIS        Number of serial ports supported.
    TYPE            uint
    DEFAULT         (2)
}

Parameter CONSOLE_BAUD_RATE {
    NAME            CONSOLE_BAUD_RATE
    SYNOPSIS        Console baud rate.
    TYPE            unsigned long
    DEFAULT         (INCLUDE_CAMELBACK_MOUNTAIN)::(115200) \
                    (115200)
}

Parameter DEFAULT_BOOT_LINE {
    NAME        default boot line
    SYNOPSIS    Default boot line string
    TYPE        string
    DEFAULT     "gei(0,0)host:vxWorks h=10.0.1.119 e=10.0.1.200 u=target"
}
```

3.12 Save the file <20bsp.cdf>.

Save the file.

3.13 Edit file <20IaDrvCfg.cdf>

Double click on *20IaDrvCfg.cdf* as shown below.
This will bring up the editor for this file.
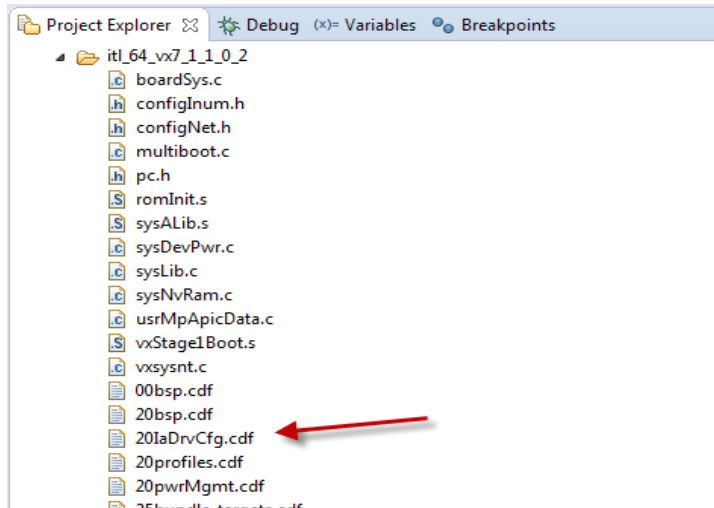
3.14 Edit the following lines

```
Parameter DRV_SIO_IA_NS16550_INSTANCE_0_TRIGGER {
    NAME                DRV_SIO_IA_NS16550_INSTANCE_0_TRIGGER
    SYNOPSIS            Trigger type for com1 interrupt, 0 is EDGE, 1 is LEVEL
    TYPE                uint
    DEFAULT             (INCLUDE_CAMELBACK_MOUNTAIN)::(1) \
                        (INCLUDE_WALNUT_CANYON_2)::(1) \
                        (INCLUDE_WALNUT_CANYON)::(1) \
                        (1)
}



Parameter DRV_SIO_IA_NS16550_INSTANCE_1_TRIGGER {
    NAME                DRV_SIO_IA_NS16550_INSTANCE_1_TRIGGER
    SYNOPSIS            Trigger type for com2 interrupt, 0 is EDGE, 1 is LEVEL
    TYPE                uint
    DEFAULT             1
}
```

3.15 Save the file <20IaDrvCfg.cdf>

3.16 Edit the Kernel Configuration

Double click on <Kernel Configuration> as shown below.
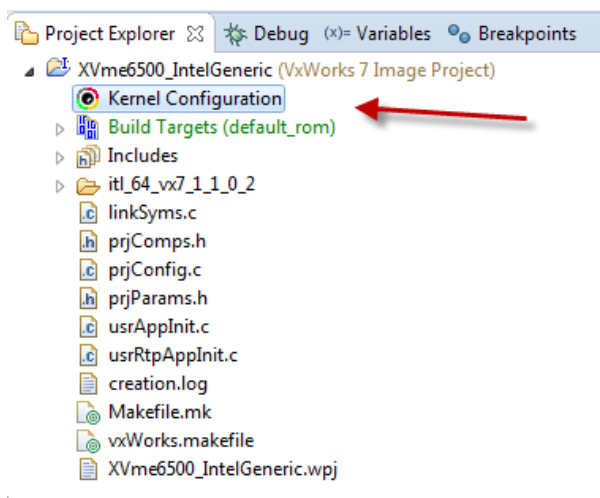
This will bring up the Components view as shown below.
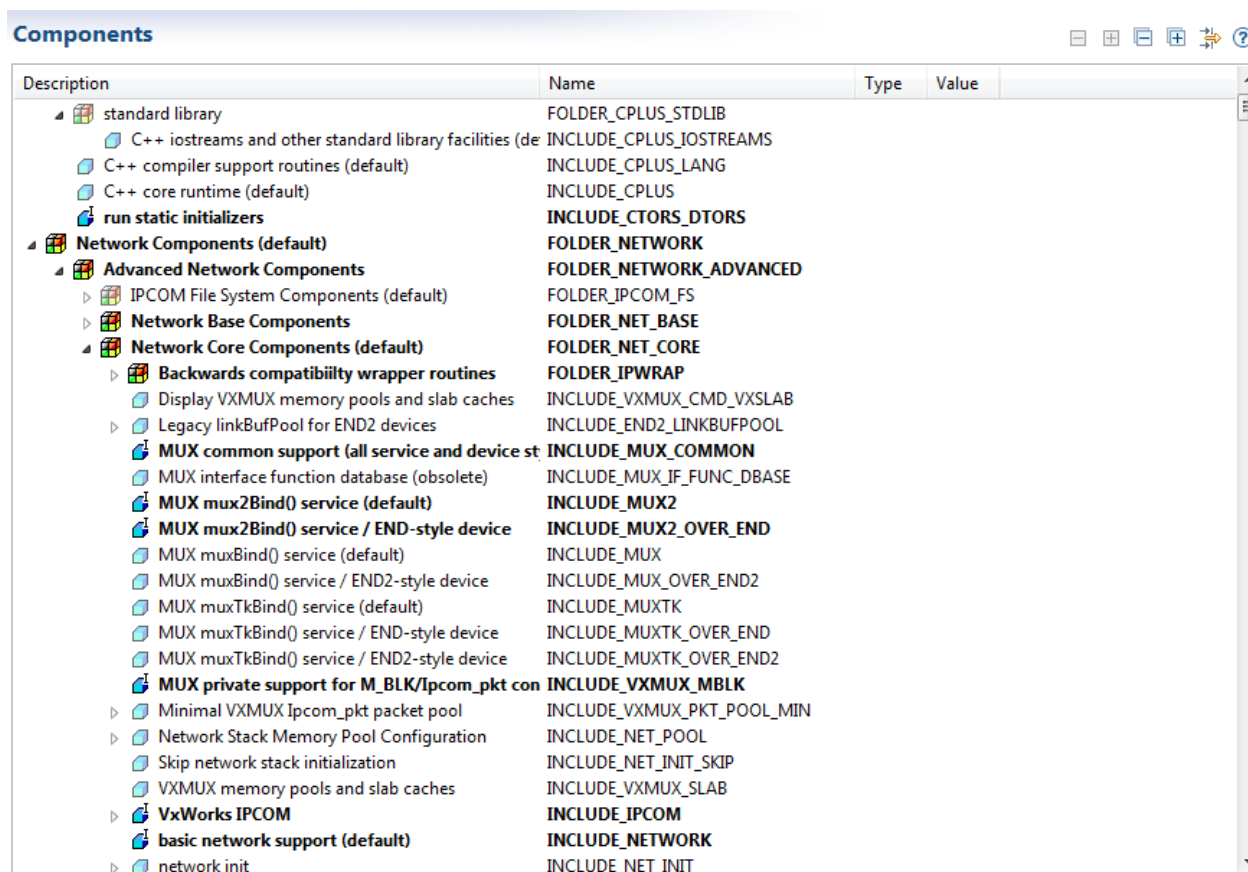
Include/Activate the following modules. Feel free to not include a module at your own risk and discretion.

INCLUDE_ACPI_CPU_CONFIG	Support for the CPU Configuration
INCLUDE_PCI_SHOW	Show PCI bus and PCI devices (BSP and Driver Guide)
INCLUDE_VXBUS_SHOW	Display all devices
INCLUDE_RTP	Realtime Process support (Programmer's)
INCLUDE_DEBUG_AGENT_START	Allows connecting to workstation via Workbench
INCLUDE_HW_FP_SHOW	Add floating point support
DRV_SIO_IA_NS16550_INSTANCE_1	Activate 2nd serial port
RAM_HIGH_ADRS	*0x00C08000* (Bootrom Copy Region)
ROM_BASE_ADRS	*0x408000* (ROM base address)
ROM_SIZE	*0x800000* (ROM size)
RAM_LOW_ADRS	*0xffffffff90008000* (Runtime kernel low address)
Exclude INCLUDE_FEI8255X_VXB_END	
INCLUDE_TELNET_CLIENT	implementation of the client side of the Telnet protocol

3.17 Log Report

The following is for debugging purposes only. The following was reported on the Log:

NOTE: the following component have been included for convenience: INCLUDE_GETADDRINFO_SYSCTL INCLUDE_HOST_TBL_SYSCTL INCLUDE_REMLIB_SYSCTL INCLUDE_SEM_EXCHANGE INCLUDE_SC_POSIX INCLUDE_RTP_DEBUG_COMMAND INCLUDE_RTP_SHELL_CMD INCLUDE_SC_SOCKLIB INCLUDE_RTP_SHELL_C INCLUDE_RTP_DEBUG INCLUDE_SC_SYSCTL INCLUDE_SYSCTL_HW
NOTE: the following component have been added as required to complete configuration: INCLUDE_NET_SYSCTL INCLUDE_SYSCTL INCLUDE_IPNET_SYSCTL INCLUDE_PIPES INCLUDE_MSG_Q_INFO INCLUDE_JOB_TASK INCLUDE_OBJ_OWNERSHIP INCLUDE_TASK_STACK_ALLOC INCLUDE_RTP_HOOKS INCLUDE_PG_MGR_LIB INCLUDE_MMAP INCLUDE_MEM_RTP INCLUDE_SC_KERNEL INCLUDE_SC_IOS INCLUDE_SC_MMAN INCLUDE_RTP_ARCH INCLUDE_VM_RTP
WARNING: also removing dependencies: INCLUDE_SC_POSIX INCLUDE_RTP_DEBUG_COMMAND INCLUDE_RTP_HOOKS INCLUDE_RTP_SHELL_CMD INCLUDE_SC_SOCKLIB INCLUDE_MEM_RTP INCLUDE_SC_KERNEL INCLUDE_SC_IOS INCLUDE_RTP_ARCH INCLUDE_RTP_SHELL_C INCLUDE_RTP_DEBUG INCLUDE_SC_SYSCTL
WARNING: the following components are no longer required and have been removed: INCLUDE_GETADDRINFO_SYSCTL INCLUDE_NET_SYSCTL INCLUDE_HOST_TBL_SYSCTL INCLUDE_REMLIB_SYSCTL INCLUDE_SYSCTL INCLUDE_IPNET_SYSCTL INCLUDE_PIPES INCLUDE_MSG_Q_INFO INCLUDE_JOB_TASK INCLUDE_OBJ_OWNERSHIP INCLUDE_SEM_EXCHANGE INCLUDE_TASK_STACK_ALLOC INCLUDE_PG_MGR_LIB INCLUDE_MMAP INCLUDE_SC_MMAN INCLUDE_SYSCTL_HW INCLUDE_VM_RTP
NOTE: the following component are no longer required and have been removed: INCLUDE_INTEL_GENERIC
NOTE: the following components have been included for convenience: INCLUDE_GETADDRINFO_SYSCTL INCLUDE_HOST_TBL_SYSCTL INCLUDE_REMLIB_SYSCTL INCLUDE_SEM_EXCHANGE INCLUDE_SC_POSIX INCLUDE_RTP_DEBUG_COMMAND INCLUDE_RTP_SHELL_CMD INCLUDE_SC_SOCKLIB INCLUDE_RTP_SHELL_C INCLUDE_RTP_DEBUG INCLUDE_SC_SYSCTL INCLUDE_SYSCTL_HW
NOTE: the following component have been added as required to complete configuration: INCLUDE_NET_SYSCTL INCLUDE_SYSCTL INCLUDE_IPNET_SYSCTL INCLUDE_PIPES INCLUDE_MSG_Q_INFO INCLUDE_JOB_TASK INCLUDE_OBJ_OWNERSHIP INCLUDE_TASK_STACK_ALLOC INCLUDE_RTP_HOOKS INCLUDE_PG_MGR_LIB INCLUDE_MMAP INCLUDE_MEM_RTP INCLUDE_SC_KERNEL INCLUDE_SC_IOS INCLUDE_SC_MMAN INCLUDE_RTP_ARCH INCLUDE_VM_RTP
NOTE: the following components have been included for convenience: INCLUDE_TIMER_OPEN INCLUDE_DISK_UTIL_SHELL_CMD

NOTE: the following component have been added as required to complete configuration: INCLUDE_POSIX_FTRUNC
INCLUDE_POSIX_PTHREADS INCLUDE_POSIX_SCHED INCLUDE_POSIX_TIMERS INCLUDE_SIGEVENT INCLUDE_DEBUG_AGENT
INCLUDE_DISK_UTIL INCLUDE_DEBUG_AGENT_BANNER INCLUDE_DEBUG_AGENT_HOSTFS
NOTE: the following components have been included for convenience: INCLUDE_HW_FP_MREGS
WARNING: the following components are no longer required and have been removed: INCLUDE_MBUF_UTIL2

3.18 Save the Kernel Configuration

3.19 Import the Kernel Include Files to the Kernel Image Project

Right click on the Image Project Name and select <Import>.
Select <File System> as shown below and click <Next>.
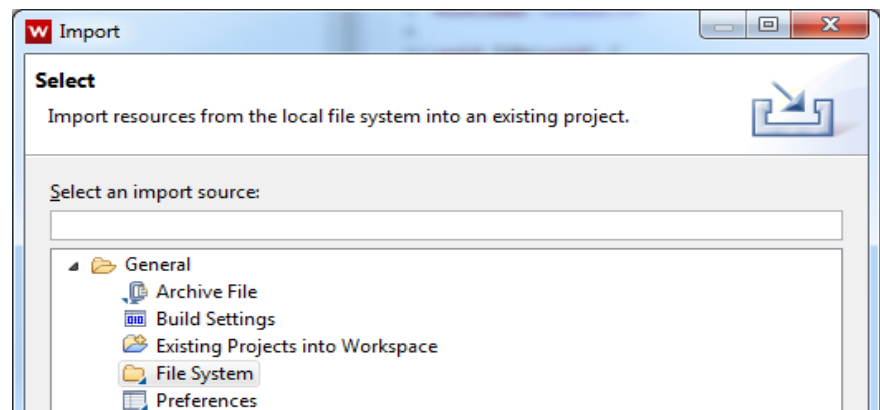
3.20 Select the 002_KernelIncludeFiles folder

Click on <Browse> and point to <TheDistributionMedia>/VxW7_64bit/002_KernelIncludeFiles and click OK.

As shown below, in the left pane of the Import Wizard, there will be the 002_KernelIncludeFiles folder that was just selected. Select the entire contents of that folder by checking the checkbox next to the folder, as shown below. Then click <Finish>.
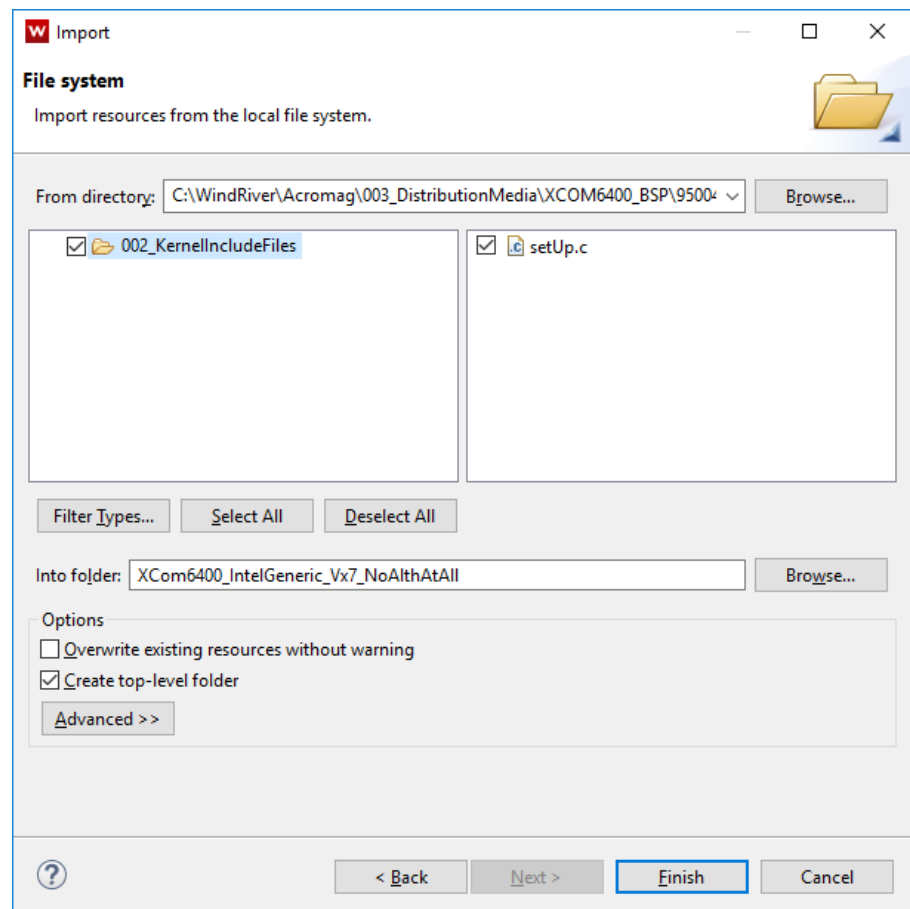
Once the import is finished, the files will show up in the Wind River Project Explorer, in the Kernel Project, under a folder called "002_KernelIncludeFiles". Select all of the files in this folder and drag them to the Kernel Image project itself. The files will then appear directly under the Kernel Image, and the 002_KernelIncludeFiles folder will be emptied. Delete the now-empty 002_KernelIncludeFiles folder.

3.21 Build the Project

Right click on the Image Project name and select <Build Project>.
Grab a cup of coffee, this process may take a while to complete.

When it does finally finish, congratulations!
You have successfully built a VxWorks 7 Image project.

4.0 Build a VxWorks 7 Downloadable Kernel Module (DKM))

4.1 Launch New Project Wizard

Within VxWorks Workbench, click <File><New><Wind River Workbench Project>.
Then select <Downloadable Kernel Module> and click <Next>.

4.2 Assign A Project Name

The Wizard will prompt you for a project name. Enter *XCom6400_Test_DKM* and click <Next>.

4.3 Select The Source Build Project

The Wizard will ask you for a Project Context. Select "Based On: A Source Build Project", and in the
"Project" dropdown, select the VxWorks Source Build Project that you just created. Then click
<Finish>.

4.4 Import The Acromag Files Into The DKM Project

Right click the new DKM project in the Project Explorer view, and click <Import>. Under <General>,
click <File System> and click <Next>.

Click the <Browse> button by the "From Directory" input box, and browse to the
<DistributionMedia>/VxW7_64bit/003_DKMIncludeFiles folder that you obtained from the
distribution media. Then click <OK>.

In the left pane of the Import Wizard, expand the 003_DKMIncludeFiles folder, and check all of the
folders inside. Then click <Finish>.

The files will appear in the Wind River Project Explorer, under the DKM Project, in a folder labeled
"003_DKMIncludeFiles". Click the arrow next to the 003_DKMIncludeFiles folder to view the
folder's contents. Highlight all of the folders inside, and drag them into the XCOM64_Test_DKM
folder one level up from the 003_DKMIncludeFiles folder. This will pull them out of their current

folder and put them in the XCOM64_Test_DKM folder. Then delete the now-empty 003_DKMIncludeFiles folder.

4.5 Import The Necessary VxW7 Files Into The DKM Project

Right click the new DKM project in the Project Explorer view, and click <Import>. Under <General>, click <File System> and click <Next>.

Click the <Browse> button by the "From Directory" input box, and browse to <YourWindRiverFolder>/vxworks-7/pkgs/net/ipnet-1.1.1.2/coreip-1.4.1.0/src/utils/src/lib. Then click <OK>.

In the right pane of the Wizard, check "ipProto.c", and click <Finish>.

In the Workbench Project Explorer window, under the DKM Project you created, expand the "Lib" folder. Drag ipProto.c from the lib folder into the DKM project folder, and then delete the now-empty "Lib" folder.

4.6 Build The DKM Project

Right click the DKM project in the Project Explorer view, and click <Build Project>.

5.0 Test the VxWorks 7 Image

5.1 Build a bootable UEFI USB Drive

- Build a bootable UEFI USB drive per the instructions in
    VxWorks 7 Boot Loader User's Guide, Edition 5

- IMPORTANT NOTE: If you have trouble building or booting from the bootable UEFI USB drive, be sure to consider the following troubleshooting notes:
  - o Before building the bootloader, to include the GNU compiler and the pre-requisite RPMs, run the "Wind River Product Maintenance" Utility, and select the "About This Installation" option to view the already-installed pre-requisites, and to install any that are needed and not already installed.
  - o When running make from the uefi-x.x.x.x folder, you must be in an Administrator Command Prompt with Wind River's special paths included in your path. To do this, open a Command Prompt from within Wind River Workbench:
    - ▪ In Wind River Workbench, right click a project in the Project Explorer View, and select "Open VxWorks 7 Development Shell"
    - ▪ The Terminal window will open, with a Command Prompt inside. In that Command Prompt, run the "runas" script/program with the following parameters:
      runas /profile /env /user:windowsAdminUser@domain.com cmd
      Then enter your password when prompted.
      This will open an Administrator Command Prompt with all of the required Wind River Paths included. Then from that Administrator Command Prompt, navigate to the uefi-x.x.x.x folder and run make as instructed.
  - o The binaries that get made will get created in your "<WindRiverInstallDir>/vxworks-7/workspace/" folder.
  - o Obtain a USB drive that you can format and format it as a FAT32 file system. (No special boot block is required.) Then create a "uefi/boot" folder, and place the two binaries that were made from the make command, place the binaries in the "boot" folder.
  - o Copy your VxWorks kernel image file into the root folder of the USB drive. Your kernel image file will be located at <YourWorkspaceFolder>/<KernelImageProjectName>/default_romCompress/vxWorks_romCompress. Once you've copied this file to the root folder of the USB drive, rename the file to "bootapp.sys" (without the quotes).
  - o In the same root folder on the USB drive, create an "obj" directory, and put the DKM.out file from the "<YourWorkspaceFolder>/<DKM Project Name>/<BSP_Name>_COREgnu_LP64_KERNEL_SMP/<DKM Project Name>/Debug/" directory into the "obj" directory on the USB drive. Rename the .out file on the USB drive to "acro.out" (without the quotes).
- On the target machine, once you've changed the boot order as instructed in the Boot Loader User's Guide, you may have to manually run "bootx64" from the (non-VxWorks) Shell that appears. Once you run "bootx64", either automatically or manually, then VxWorks should boot properly.

5.2 Boot to VxWorks 7 Image

If all goes well, your system should boot to a VxWorks 7 image screen with a prompt similar to the one below.

```
            VxWorks 7 64-bit

Copyright 1Status Code Available
DXE Status Code Available
ERROR: Type:2; Severity:80; Class:1; Subclass:1; Operation: 3
Target Name: vxTarget


            VxWorks 7 64-bit

Copyright 1984-2016 Wind River Systems, Inc.

    Core Kernel version: 1.0.11.2
            Build date: Feb 25 2016 14:22:55
                 Board: Intel(R) Haswell Processor Lava Canyon
        OS Memory Size: ~16223MB
     ED&R Policy Mode: Permanently Deployed


 Adding 8858 symbols for standalone.

->
```

6.0 Build the VxWorks 7 Acromag Library

This process is not applicable for the XCOM hardware series.

7.0 Build the VxWorks 7 Acromag Application

This process is not applicable for the XCOM hardware series.

8.0 Launch your VxWorks System

8.1 Boot your VxWorks 7 Image

If all goes well, your system should boot to a VxWorks 7 image screen with a prompt similar to the one below.

```
            VxWorks 7 64-bit

Copyright 1Status Code Available
DXE Status Code Available
ERROR: Type:2; Severity:80; Class:1; Subclass:1; Operation: 3
Target Name: vxTarget


            VxWorks 7 64-bit

Copyright 1984-2016 Wind River Systems, Inc.

     Core Kernel version: 1.0.11.2
             Build date: Feb 25 2016 14:22:55
                  Board: Intel(R) Haswell Processor Lava Canyon
         OS Memory Size: ~16223MB
      ED&R Policy Mode: Permanently Deployed


 Adding 8858 symbols for standalone.

->
```

8.2 Launch Acromag Test Application

At the VxWorks 7 prompt, type the following.

-> *setUpVxW("usb") <Enter>*

The VxWorks Shell should print that the DKM was loaded successfully. If it doesn't, then re-do this manual's Sections 3.19-4.6 and try again.

If the DKM module loads successfully, then enter the following command:

-> *xcom_menu <Enter>*

You should get the following screen:

```
Acromag XCOM VxWorks BSP Test Menu
----------------------------------
 1.   Read XCOM SmBus
 2.   Read XCOM NCT6776 voltages
 3.   Toggle GPOs high
 4.   Toggle GPOs low
 5.   Write Hello World to Com0 at 115200,8,n,1
 6.   Write Hello Earth to Com1 at 115200,8,n,1
 7.   Initialize network ports

 X.   Exit
```

You can look at the source code for this at

\Acromag.Files\Acromag.Image\testMenu.c

8.3 Read SmBus Devices

At the Acromag Menu, press the <1> key as shown below.
```
Enter choice: 1 <Enter>
```

The system should respond with something similar to the following:

```
The unsigned short SMBus Base Address is 0x4040
Disable Multiplexer: SlaveAddress=0xE0  IndexRegister=0x0
DataByte=0x0
Disable Multiplexer: SlaveAddress=0xE0  IndexRegister=0x0
DataByte=0x0
Temperature is   27.25000000 degrees Celsius
Temperature is   26.50000000 degrees Celsius
Enable LTC2991: SlaveAddress=0x9E  IndexRegister=0x1
DataByte=0xF8
0. Temperature is   26.62500000 degrees Celsius
1. V1 (DCIn) is   24.30424223V
2. V2 (DCIn) is   24.30170313V
3. V4 (12V Source) is   12.05846748V
4. V6 (5V Source) is    5.09589564V
5. V8 (3.3V Source) is    3.33012416V
6. Vcc (Source) is    5.09189374V
7. V3 (12V Load) is    2.38089229A
8. V5 (5V Load) is    0.18615980A
9. V7 (3.3V Load) is    0.88990488A
Enable Multiplexer: SlaveAddress=0xE0  IndexRegister=0x0
DataByte=0x4
Total time elapsed for read_smbus() is    3.00000000
```

The information returned are the temperatures of the DIMM board(s) installed on your XCOM system as well as the various voltages and current loads connected to the LTC2991. Please note that you need to supply the correct SMBus Address of the system in the source code.

You can look at the source code for this at

\Acromag.Files\Acromag.Image\testMenu.c
\Acromag.Files\Acromag.Image\testSmBus.c

Code Snippet

read_smbus("xcom"); /* xvme for xvme series; xcom for xcom series */

8.4 Read Super I/O NCT6776 Voltages

At the Acromag Menu, press the <2> key as shown below.

Enter choice: *2 <Enter>*

The system should respond with something similar to the following:

```
The following are Super I/O information.
Vddq  voltage is = 1.3520 V
Vcc5  voltage is = 5.1097 V
Vcc1  voltage is = 1.0720 V
Vcc3  voltage is = 3.3760 V
Vcc12 voltage is = 12.0849 V
Vcore voltage is = 1.8000 V
```

The information returned are the voltages reported by the NCT67766 Super I/O. You can look at the source code for this at

\Acromag.Files\Acromag.Image\testMenu.c
\Acromag.Files\Acromag.Image\testSIO.c

Code Snippet

read_voltage("xvme"); /* xvme for xvme series; xcom for xcom series */

8.5 Test GPIO Stage 1

At the Acromag Menu, press the <3> key as shown below.

Enter choice: *3 <Enter>*

The four GPO signals should toggle high with the message below.

GPO 0-3 were toggled on.

You should be able to measure the voltages on the pins and they should be high (Approximately 3.3V) as shown below.

You can look at the source code for this at

\Acromag.Files\Acromag.Image\testMenu.c
\Acromag.Files\Acromag.Image\testGPIO.c

Code Snippet

```
user_gpo0 (1);
user_gpo1 (1);
user_gpo2 (1);
user_gpo3 (1);
```

8.6 Test GPIO Stage 2

At the Acromag Menu, press the <4> key as shown below.

Enter choice: *4 <Enter>*

The four GPO signals should toggle low with the message below.

GPO 0-3 were toggled off.

You should be able to measure the same voltages on the pins above and they should be zero volts.

You can look at the source code for this at

\Acromag.Files\Acromag.Image\testMenu.c
\Acromag.Files\Acromag.Image\testGPIO.c

Code Snippet

```
user_gpo0 (0);
user_gpo1 (0);
user_gpo2 (0);
user_gpo3 (0);
```

8.7 Test Serial Port 0

Connect a null modem serial port cable between your PC and the XCOM-6400 board Serial Port 0. Start a terminal application such as Putty on your PC and set the baud rates, data bits, parity, and stop bits to 115200,8,n,1.

At the Acromag menu, press the <5> key as shown below.

Enter choice: *5 <Enter>*

The application should respond with the message below.

```
Hello World!
"Hello World!" message sent to Com0.
```

You should see the following message on your PC terminal software.

You can look at the source code for this at

\Acromag.Files\Acromag.Image\testMenu.c
\Acromag.Files\Acromag.Image\testSerial.c

Code Snippet

```
Write_serial0("Hello World!",115200);
```

8.8 Test Serial Port 1

Connect a null modem serial port cable between your PC and the XCOM-6400 board Serial Port1. Start a terminal application such as Putty on your PC and set the baud rates, data bits, parity, and stop bits to 115200,8,n,1.
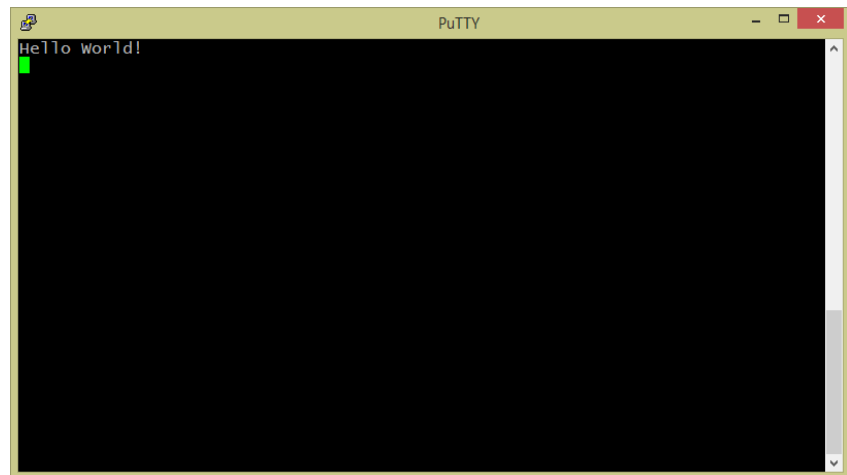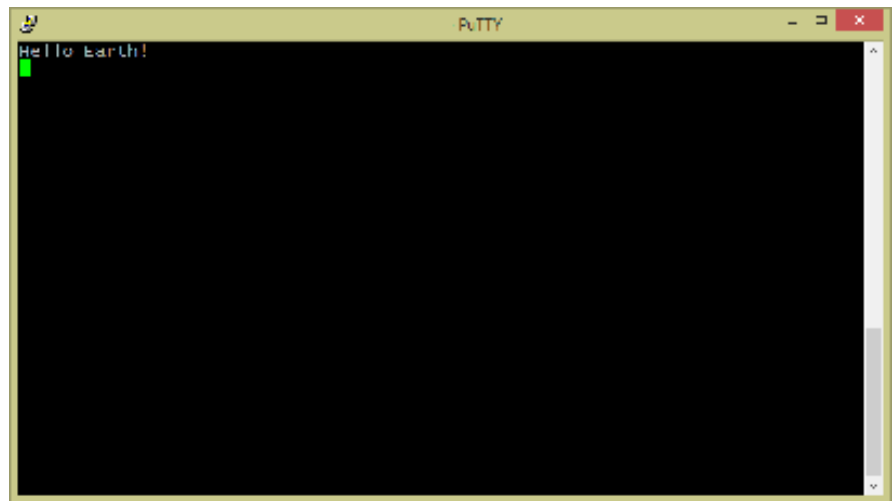
At the Acromag menu, press the <6> key as shown below.

Enter choice: *6 <Enter>*

The application should respond with the message below.

Hello Earth!
"Hello Earth!" message sent to Com1.

You should see the following message on your PC terminal software.

You can look at the source code for this at

\Acromag.Files\Acromag.Image\testMenu.c
\Acromag.Files\Acromag.Image\testSerial.c

<u>**Code Snippet**</u>

```
Write_serial1("Hello Earth!",115200);
```

8.9 Initialize Network Ports

The first network port found on the CPU is set to automatically initialize during startup. However, any of the ports can be initialized at runtime via the xcom_menu() routine.

IMPORTANT NOTE: Use caution if configuring the network ports remotely, as the new configuration could cause the connection with the target to be lost.

At the Acromag Menu (call "xcom_menu()" without the quotes from the VxWorks Shell), press the <7> key, as shown below, to initialize the network ports.

Enter choice: *7 <Enter>*

The system will then ask which port to configure, and then will prompt for an IP address for the port. The system will then configure the port according the the answers it received.

The source code for this configuration menu can be found at:

\VxW7_64bit\003_DKMIncludeFiles\acro_xcom6400_test\testMenu.c
\VxW7_64bit\003_DKMIncludeFiles\acro_xcom6400_test\testNetwork.c

8.10 Exit the Acromag Test Menu

At the Acromag Menu, press the <x> key as shown below.

Enter choice: *X <Enter>*

This option will close the menu and return you to the VxWorks prompt.

9.0 Revision History

The following table shows the revision history for this document:

Release Date	Version	EGR/DOC	Description of Revision
04 MAY 16	A	BLD/ARP	Preliminary release.
01 JUN 16	A	BLD/ARP	Production release.
27 JUN 16	B	BLD/ARP	Added symmetric multiprocessor support.
23 MAR 2018	C	JBO/ARP	Updated for VxW7, and added some explanations for ease of use.
13 NOV 2018	D	JBO/ARP	Updated to use itl_generic instead of itl_64.