



J-Squared Technologies Inc.

VxWorks 7 – Reflective Memory Driver

Install Instructions:

The driver is delivered in source form as an RPM file called rfm2_driver-1.2.0.0-vx7.noarch.rpm.

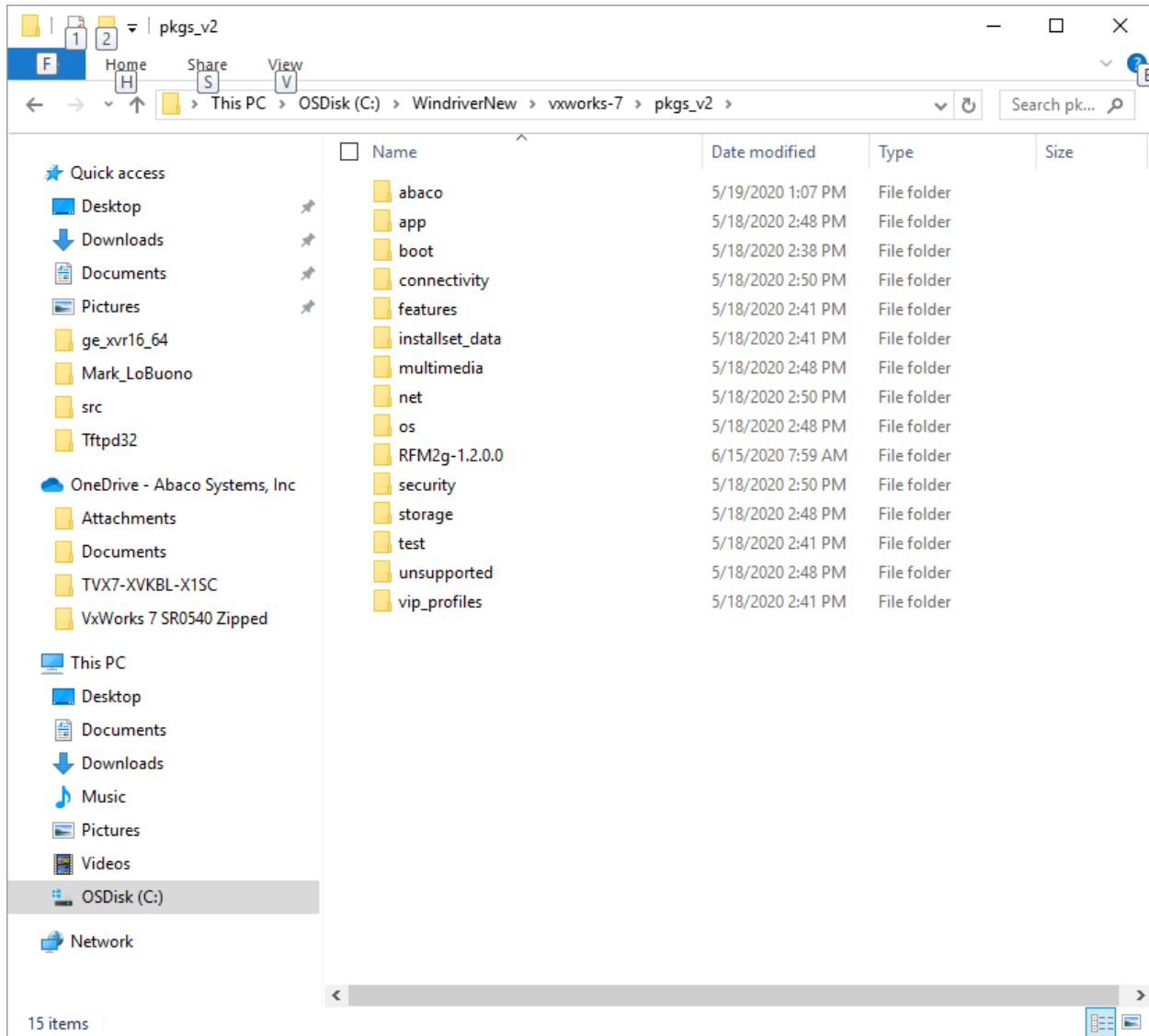
It is assumed that VxWorks 7 (SR0630) and the Abaco Systems® XVKBL BSP are already installed. If not, they must be installed first.

Then perform the following steps using C:\Windriver\workspace\ as the Windows example install directory.

- 1) Open a Windows command shell. You can use PowerShell or the old cmd program. Do not use cygwin bash.
- 2) Execute the following command to remove previously installed versions of the driver. This will remove all versions of the driver currently installed.
C:\Windriver\maintenance\wrInstaller\x86-win32\wrInstallerc.exe -y -nosplash -silent -installerUpdateURLS none -yum remove rfm2_driver.noarch
- 3) Execute the following command to install the new driver:
C:\Windriver\maintenance\wrInstaller\x86-win32\wrInstallerc.exe -y -nosplash -silent -installerUpdateURLS none -yum localinstall rfm2_driver-1.2.0.0-vx7.noarch.rpm
- 4) The driver files will be installed in C:\Windriver\vxworks-7\pkgs_v2\RFM2g-1.2.0.0 as shown here:



J-Squared Technologies Inc.



- 5) Check the Source Build Configuration for the tree item “RFM2g-1.2.0.0” and verify the “Enable RFM2g driver 1.2.0.0” has a value of “y”
- 6) Check the Image Kernel Configuration under the tree item “Hardware (default)” and verify the sub-tree item “RFM2g” is present
- 7) Select desired Components. The “RFM2g Sample Programs” contains the “RFM2g Utility Program” if that is desired in the kernel image
- 8) Set the Reflective Memory Board ID dip switch setting in the sub-tree item “RFM2G PCI VxBus driver” and sub-tree item “PCI RFM2G BUS ID” to match



J-Squared Technologies Inc.

Build Driver Source:

The RFM2 driver for VxWorks 7 is designed to be built as part of the VSB. Create a VSB (Source Build) project in Workbench, based on the `as_xvkb1_2_0_0_0` BSP. Make sure the VSB is configured for SMP mode.

Before building the VSB, expand your VSB in Workbench's project explorer and open the source build configuration. At the top level, you should see an item called `RFM2fg-1.2.0.0`. Expand this to confirm it is enabled. (The driver is enabled by default and can be disabled to exclude it from a build if desired).

Create and Build Kernel Image:

Create a VIP (Kernel Image) project in Workbench, based on the VSB built in the previous step. Select the `PROFILE_DEVELOPMENT` profile when created the project to add the VxWorks target shell to the project.

Open the Kernel Configuration for the VIP and add the component `DRV_RFM2G_PCI` to your project to include the RFM2 driver. This will also add any dependencies. Also add the component `DRV_RFM2G_IOS` which will register the driver with the I/O subsystem.



J-Squared Technologies Inc.

Add Sample Code to Kernel:

To add the sample code to the kernel, open the Kernel Configuration for the VIP and search for the configuration item called FOLDER_GEF_RFM2G. In that folder there are four components, one for each of the sample programs:

GEF_RFM2G_UTIL, GEF_RFM2G_MAP, GEF_RFM2G_SENDER, and GEF_RFM2G_RECEIVER. Select the sample program(s) to include in the kernel image, save the configuration, and build the image.

For details on what the sample programs do, see the [rfm2g_common.pdf](#).

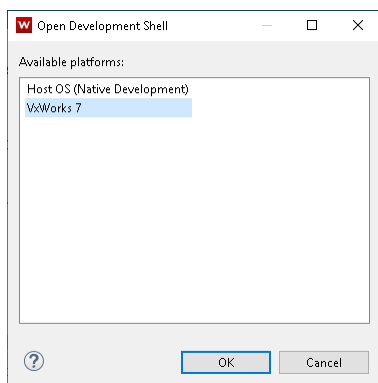


J-Squared Technologies Inc.

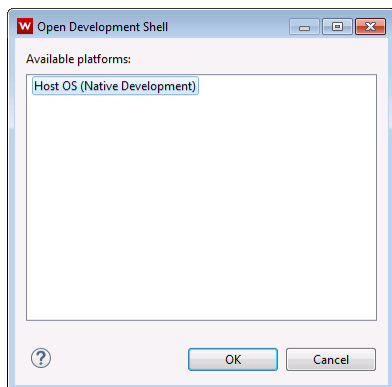
Building Sample Code VXE Files for RTPs:

The sample code can also be built as VXE files to be downloaded to the target and executed in RTPs. This build is done using a development shell in Workbench.

In Workbench, select the menu item **Project->Open Development Shell...** You should see a dialog open like the one below. Select **VxWorks 7**. Go to the section **Error! Reference source not found.** for build instructions.



If instead, you see a dialog like below without the **VxWorks 7** option, select **Host OS**. There is an additional configuration step required for this case.



In the development shell in Workbench, change directory to **C:\Windriver** and execute the following command:

wrenv.exe -p vxworks-7



J-Squared Technologies Inc.

In the development shell in Workbench, execute the following commands:

```
cd C:\Windriver\vxworks-7\pkgs_v2\RFM2g-1.2.0.0\rtp\sh build_all.sh  
<VSB_NAME>
```

<VSB_NAME> is the name of the VSB project used to build the RFM2 driver and is listed in the project explorer pane of Workbench.

The output files from the build will be in the workspace directory used by Workbench, e.g. **C:\WindRiver\workspace**. A directory is created for each of the sample programs (**rfm2g_map**, **rfm2g_receiver**, **rfm2g_sender**, and **rfm2g_util**) and the VXE file can be found in the subdirectory **SKYLAKEgnu_LP64_RTP** for each sample program. For example, for the sample program **rfm2g_util**, the VXE file can be found at **rfm2g_util/SKYLAKEgnu_LP64_RTP/rfm2g_util.vxe** in Workbench's workspace directory.

Note that these sample programs will not appear in the project explorer pane in Workbench.



J-Squared Technologies Inc.

Check for Device Presence:

Typing the command `devs` in the VxWorks target shell should show an instance of the RFM2 driver, something like below. Note that you need to add the component `DRV_RFM2G_IOS` to your VIP configuration.

-> `devs`

`drv name`

`0 /null`

`1 /tyCo/0`

`1 /tyCo/1`

`9 host:`

`10 /PCIRFM2G_0 /* the is the PCI RFM2G device */`

Status LED:

The driver initialization turns off the Red Status LED on the RFM2 board.