Be Boulder.

PROJECT #3 GUIDE

UCB ECEN 5803 FALL 2021 PROJECT #3: TI AM3358 CORTEX A8 WITH WINCE AND LINUX

TABLE OF CONTENTS

I. MODULE 1 BUILD THE WINDOWS EMBEDDED COMPACT 7 TOOLCHAIN

1. Platform Builder for Windows Embedded Compact 7 is a plug-in for Visual Studio 2008 and requires the following to function: Visual Studio 2008, Visual Studio 2008 Service Pack 1, and NET Compact Framework 3.5. If you find that a WEC 7 evaluation key is not available, try the one in WInProductKey.docx on Canvas.
2. Download:

   .NET Compact Framework 3.5 - https://www.microsoft.com/en-us/download/details.aspx?id=21

   Visual Studio 2008: (3.30 GB) http://download.microsoft.com/download/8/1/d/81d3f35e-fa03-485b-953b-ff952e402520/VS2008ProEdition90dayTrialENUX1435622.iso – for Visual Studio 2008, https://www.microsoft.com/en-us/download/details.aspx?id=38794 for Windows Embedded Compact 7, 180 day evaluation. See also https://docs.microsoft.com/en-us/previous-versions/windows/embedded/jj200367%28v%3dwinembedded.70%29

   You need Visual Studio 2008 Service Pack 1 which is available for download from the following URL: http://www.microsoft.com/downloads/en/details.aspx?FamilyID=27673C47-B3B5-4C67-BD99-84E525B5CE61
3. Install .NET compact Framework 3.5. Do this before installing Visual Studio.
4. Follow the instructions in Windows Embedded Compact 7 Evaluation Edition.docx on Canvas to complete the installation.
5. Start the Platform Builder, and capture a screen shot.

II. MODULE 2 BOOT WINDOWS AND TEST

1. To test Windows CE 7, we will use a pre-built image. The process for doing this is very similar to what you have done with the Linux images. Begin by downloading WEC7 BBB 01.17.00.zip from Canvas. This may take some time.

2. Follow the instructions in BBB_WinCE_Platform Support.docx to create the image, but be careful to use the HP image creator in WEC7 BBB/Tools/Drivekey directory, and not the TI image writer. After completing the directions to create the image, insert the SD card, press the user button and power up. Boot from the SD Card, DO NOT MOVE THE IMAGE TO THE EMMC. Ignore the eMMC portion of the document.
3. Upon booting up, the serial port should be sending out debug messages. Open a terminal window to capture them. What do you see? Be aware that the port used to transmit the information may be different from what you expect and may require an RS232 to USB adapter.
4. How much memory is used by the code? (What is the image size?)
5. Capture a screen shot of the terminal window.
6. The Ethernet and HDMI ports should also be active. Connect the HDMI output to a monitor using an HDMI Cable and adapter if necessary. Reboot the system – what do you see?
7. Write C code for a G.711 coder/decoder. See http://www.opensource.apple.com/source/tcl/tcl-20/tcl_ext/snack/snack/generic/g711.c for an example. Use this decoder to decode a file given to you by your instructor. You will need to use Visual Studio 8 to compile code for this application.
8. Record your observations. How is the behavior of Windows CE 7 different from Linux?

   Now, evaluate Windows Embedded Compact 2013
9. Begin by downloading WEC2013 BBB 03.02.zip from Canvas. This may take some time.
10. Follow the instructions in BBB_WinCE_Platform Support.docx to create the image, but be careful to use the HP image creator in WEC7 BBB/Tools/Drivekey directory or Win32 Disk Imager from Source Forge, and not the TI image writer. After completing the directions to create the image, insert the SD card, press the user button and power up. Boot from the SD Card, DO NOT MOVE THE IMAGE TO THE EMMC. Ignore the eMMC portion of the document.
11. Upon booting up, the serial port should be sending out debug messages. Open a terminal window to capture them. What do you see?
12. How much memory is used by the code? (What is the image size?)
13. Capture a screen shot of the terminal window.
14. The Ethernet and HDMI ports should also be active. Connect the HDMI output to a monitor using an HDMI Cable and adapter if necessary. Reboot the system – what do you see? How does this differ from Win CE 7? Note: Code development for Win CE 2013 will require Visual Studio 2012.

III. MODULE 3 SCRIPTING WITH LINUX

Aim: Creation and running an executable script in Linux to print useful information about the Linux Kernel.

Scope: The function implemented by the script is under the discretion of the group. The complexity and number of functions implemented is also under the discretion of the group.

Procedure:
Creating an executable script:
i) Use a suitable scripting language for writing a script that uses commands in the terminal for access data relevant to the terminal. Scripts can be written in bash or python or something similar. Information for bash can be

obtained from http://tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html and http://www.freeos.com/guides/lsst/. An introduction to python can be found here: https://docs.python.org/2/tutorial/ and https://www.linuxjournal.com/article/1121 .

ii) Convert the script to an executable file.

iii) An additional bonus points will be awarded if the script is executed at boot time.

Example of what the executable script may perform
- display the current running processes.
- display the current data and time, kernel version
- display the kernel dump.

The points on this exercise will vary on two factors:

A. Number of functions implemented by the script. (Minimum: 3)
B. Complexity of the functions implemented by the script. (A function is not complex if there is 1 exclusive command that performs that function).

IV. MODULE 4 BUILD YOUR OWN PBX WITH ASTERISK

1. For development of code in Linux, GCC and GDB are the preferred compiler and debugger to use. Go to www.asterisk.org and download Asterisk. Look at https://wiki.asterisk.org/wiki/display/AST/Beginning+Asterisk and read the sections on Beginning Asterisk, Installing Asterisk, and the Hello World Project. You may be able to use either the Debian or Angstrom package repository to install the asterisk package directly after connecting the BeagleBone to the internet under Linux. Carefully read the documentation and online guides and incorporate Asterisk, the open source PBX into one of your Linux SD cards. How much memory is used by the code? (What is the image size?)
2. Using either a SIP phone plugged into the same LAN as the BeagleBone Black (you may need an Ethernet switch to create the network), or with a PC running a softphone application connected to the Beaglebone, configure Asterisk to provide a voicemail message at extension 100. Configure your SIP phone or softphone and register with Asterisk.
3. Make a call to extension 100 and record what you hear.
4. [Optional, for 5 extra credit points] Add another SIP phone or softphone to the network, and make a phone to phone call.

V. MODULE 5 BUILD A WINDOWS CE TELECOM OR IOT APPLICATION (OPTIONAL, FOR 10 POINTS EXTRA CREDIT)

1. Using either Windows Embedded Compact 7 or 2013 on the BeagleBone, or on a VM on your laptop, create an application related to some telecom or IoT function. You can start with one of the test programs included in the BSP on Codeplex. Some possible applications:

    Any Mobile Phone apps, like SMS messaging, voicemail, or contact manager;

A Morse Code Receiver that decodes and displays incoming Morse Code generated by a switch on a small microcontroller board;

An IoT sensor aggregation application that uses SPI, I2C, and/or GPIO sensors and packetizes the information for Ethernet Access;

A Web server that provides a home webpage for your Beagle Bone Black.

2. Build your application, and any associated test hardware or interfaces, and test your new application.
3. Document your new application by providing a zip file of the code and screenshots showing the application in operation. Be prepared to demo this application to the TAs.

VI. MODULE 6 BUILD YOUR OWN WINDOWS CE (OPTIONAL, FOR 10 POINTS EXTRA CREDIT)

1. Carefully read the documentation and online guides and use Platform Builder to create a custom Windows Embedded Compact 7 OS for a virtual machine. Virtual PC, Virtual Box, or VMware are typical choices. Boot Windows in the virtual machine and capture a screenshot. Again you may need to do this in Windows 8.1 or 7 compatibility mode.
2. For even more extra credit, create a custom Windows Embedded Compact 7 or 2013 OS for the BeagleBone Black. Boot the BeagleBone, and record either the serial port terminal window or the HDMI output.

VII. MODULE 7 MEASURE ARM CORTEX-A8 DMIPS UNDER WINCE AND LINUX (OPTIONAL, FOR 5 POINTS EXTRA CREDIT)

1. Port your DMIPs benchmark code from Project 2 and compile and run it under Linux on the BeagleBone Black.
2. Port your DMIPs benchmark code from Project 2 and compile and run it under WinCE on the BeagleBone Black. Does it match your expectations? How does it compare to the Linux implementation – do you get the same performance numbers?