# P1.1.2 TEST PLAN

*TCP IP STACK*

*VERSION 1*

SEPT 18, 2017

RICHARD CONSTANTINE

#7686561

# TABLE OF CONTENTS

# 1 – TPP VERSION HISTORY

| Version # | Implemented By | Revision Date | Approved By | Approval Date | Reason | Mark |
|---|---|---|---|---|---|---|
| 1 | Richard Constantine | 09/18/2017 | | | | |

# 2 - INTRODUCTION

## 2.1 PURPOSE OF THE TEST PLAN TEMPLATE DOCUMENT

The purpose of this test plan document is to test the TCP/IP capabilities of the MX7cK development board. This can be accomplished by connecting the board to another host computer via the Ethernet connector, editing the provided ETH795.h file with the MAC and IP addresses with the information printed on the MX7 device. Finally, the device can be tested by pinging the MX7cK board with TCP/IP packets using the *ping* command within Windows Command Prompt,

# 3 – TCP/IP STACK TESTING

## 3.1 TEST PLAN AND CASES

| Item to Test | Test Description | Test Date | Responsibility |
|---|---|---|---|
| TCP IP Stack/Ethernet Connection Test | By editing the provided ETH795.h file (located within TCP/IP Stack code provided) with the correct MAC and IP information, along with an Ethernet cable, the MX7cK board can be connected to another host and communicating via TCP/IP packets. This connection can be tested by pinging the MX7cK board with TCP/IP data packets generated and sent by the connected host. | Sept 18, 2017 | Richard Constantine |

## 3.2 TEST PROCEDURE

| Instruction | | P/F |
|---|---|---|
| 1. | Ensure the PC being used has an up-to-date version of Windows (Windows 7 or later) along with the MPLAB X IDE (currently version is 4.01), and XC32 compiler (currently version 1.44), and the PIC32 Legacy Peripheral Libraries (ensuring to install the libraries within the xc32/v1.44 folder). These tools can be found at http://www.microchip.com/mplab/ along with more detailed instructions in the reference material [1]. | |
| 2. | Connect the J3 Power Select (DBG jumper just above the Ethernet port) of the MX7cK board if it is not already connected. Then connect the MX7cK board via the J15 Debug USB connector (located just below the Ethernet connector) via the provided USB OTG cable. Power on the device using the power switch in the lower left corner – LEDs should light up which confirms that the device is receiving power. | |

| | | |
|---|---|---|
| **3.** | **IF ALREADY COMPLETE, MOVE TO STEP 4.** Go to the Google OneDrive URL sent via email. This should take you a folder called Richard Constantine - 7686561 - ECE 3740 - Assignment 1. Download and extract all files within Constantine_Richard_P1.1.zip, and place its contents in a directory called C:\Users\<YourUsername>\ECE3740\MPLABXProjects\Constantine_Richard_P1\P1.1. | |
| **4.** | Start the MPLAB X IDE by clicking on MPLAB X in the taskbar (as explained by Step 1 of [1]) or find the MPLAB X IDE by going to Windows File Explorer, browsing to the folder or drive where MPLABX is installed, using Windows search (in the top right corner of the window) for "IDE" then scrolling until mplab.exe can be located and opened. | |
| **5.** | Open the XC32-PIC32_ETH_SK_ETH795 project by selecting File->Open Project within the MPLAB X IDE (top left corner of the window), and then browsing to the newly extracted directory: C:\Users\<YourUsername>\ECE3740\MPLABXProjects\Constantine_Richard_P1\P1.1\P1.1.2\v1\TCPIP Stack\TCPIP\Demo App\ then selecting and opening the project XC32-PIC32_ETH_SK_ETH795.X. | |
| **6.** | If no files are displayed beneath the XC32-PIC32_ETH_SK_ETH795 project in the Projects tab (top left of the window), click the '+' icon to the left of the XC32-PIC32_ETH_SK_ETH795 project. This should display the files located in that folder such as Header Files, Important Files, etc.. | |
| **7.** | Click the '+' icon to left of the Header Files (located in the projects tab of the MPLABX IDE) which should also display the files such as MainDemo.h, TCPIP ETH795.h, etc., located within that folder. Open the file TCPIP ETH795.h file within the Header Files folder by right-clicking the file and selecting 'Open' – this should reveal the C code comprising that file. | |
| **8.** | Within the TCPIP ETH795.h code, locate 'Application Options' (located at line 57 of the code). Ensure that every define statement located under 'Application Options' is commented out EXCEPT lines 67, 74, and 84:<br>• #define STACK_USE_ICMP_SERVER // Ping transmission capability<br>• #define STACK_USE_DHCP_SERVER // Single host DHCP server<br>• #define STACK_USE_ANNOUNCE // Microchip Embedded Ethernet Device Discoverer | |
| **9.** | Find the MAC address printed on the back on the back your (the tester's) MX7cK board - the MAC address consists of 12 hexadecimal digits in the form UU-VV-WW-XX-YY-ZZ. Record this for the later steps. | |
| **10.** | Within the TCPIP ETH795.h code, locate 'Network Addressing Options' (at line 144). The MAC and IP information is currently configured for my (Richard Constantine's) device which has a MAC address of 00-18-3E-01-AD-37. These settings should be altered to match the tester's MAC info. First, find the host name on line 155:<br>• #define MY_DEFAULT_HOST_NAME "mx37"<br>And change it to read:<br>• #define MY_DEFAULT_HOST_NAME "mxZZ"<br>Where ZZ represents the last 2 digits of the MAC address printed on the back of your (the tester's) board. For example, using the previous MAC address of 00-18-3E-01-AD-37, we find ZZ = 37. Follow these instructions using the MAC address printed on the back of your board. | |
| **11.** | Still within the TCPIP ETH795.h code, ensure that lines 164-169 match your (the tester) device's MAC address by editing lines 164-169 to match your MAC address. Using a MAC address of UU-VV-WW-XX-YY-ZZ, these lines would look like:<br>• #define MY_DEFAULT_MAC_BYTE1 (0xUU) // MAC address printed on back of board<br>• #define MY_DEFAULT_MAC_BYTE2 (0xVV) // | |

| | | |
|---|---|---|
| | • #define MY_DEFAULT_MAC_BYTE3 (0xWW) //<br>• #define MY_DEFAULT_MAC_BYTE4 (0xXX) //<br>• #define MY_DEFAULT_MAC_BYTE5 (0xYY) //<br>• #define MY_DEFAULT_MAC_BYTE6 (0xZZ) //<br>For the previous example MAC address 00-18-3E-01-AD-37, these lines would look like:<br>• #define MY_DEFAULT_MAC_BYTE1 (0x00) // MAC address printed on back of board<br>• #define MY_DEFAULT_MAC_BYTE2 (0x18) //<br>• #define MY_DEFAULT_MAC_BYTE3 (0x3E) //<br>• #define MY_DEFAULT_MAC_BYTE4 (0x01) //<br>• #define MY_DEFAULT_MAC_BYTE5 (0xAD) //<br>• #define MY_DEFAULT_MAC_BYTE6 (0x37) //<br>Do cha | |
| **12.** | The IP address should also be updated to reflect the new MAC address. Confirm that lines 171-173 of the TCPIP ETH795.h code match the following:<br>• #define MY_DEFAULT_IP_ADDR_BYTE1 (192ul)<br>• #define MY_DEFAULT_IP_ADDR_BYTE2 (168ul)<br>• #define MY_DEFAULT_IP_ADDR_BYTE3 (1ul)<br>Find the line 174 and change it read:<br>• #define MY_DEFAULT_IP_ADDR_BYTE4 (0xZZul)<br>Where ZZ represent the last 2 digits of the your (the tester) device's MAC address – same idea as step 11. | |
| **13.** | Confirm that the default mask located on lines 176-179 of the TCPIP ETH795.h code are defined as:<br>• #define MY_DEFAULT_MASK_BYTE1 (255ul)<br>• #define MY_DEFAULT_MASK_BYTE2 (255ul)<br>• #define MY_DEFAULT_MASK_BYTE3 (255ul)<br>• #define MY_DEFAULT_MASK_BYTE4 (0ul) | |
| **14.** | Next, find the default gate and default primary DNS located on lines 181-189:<br>• #define MY_DEFAULT_GATE_BYTE1 (192ul)<br>• #define MY_DEFAULT_GATE_BYTE2 (168ul)<br>• #define MY_DEFAULT_GATE_BYTE3 (1ul)<br>• #define MY_DEFAULT_GATE_BYTE4 (55ul)<br>•<br>• #define MY_DEFAULT_PRIMARY_DNS_BYTE1 (192ul)<br>• #define MY_DEFAULT_PRIMARY_DNS_BYTE2 (168ul)<br>• #define MY_DEFAULT_PRIMARY_DNS_BYTE3 (1ul)<br>• #define MY_DEFAULT_PRIMARY_DNS_BYTE4 (55ul)<br>Using the tester's MAC address UU-VV-WW-XX-YY-ZZ, convert the last 2 digits of the MAC address (ZZ) of your (the tester's) local device from hexadecimal to decimal. Say this decimal number is represented by Φ, then the lines (181-189) should then be changed to the following:<br>• #define MY_DEFAULT_GATE_BYTE1 (192ul)<br>• #define MY_DEFAULT_GATE_BYTE2 (168ul)<br>• #define MY_DEFAULT_GATE_BYTE3 (1ul)<br>• #define MY_DEFAULT_GATE_BYTE4 (Φ ul)<br><br>• #define MY_DEFAULT_PRIMARY_DNS_BYTE1 (192ul) | |

| | | |
|---|---|---|
| | • #define MY_DEFAULT_PRIMARY_DNS_BYTE2 (168ul)<br>• #define MY_DEFAULT_PRIMARY_DNS_BYTE3 (1ul)<br>• #define MY_DEFAULT_PRIMARY_DNS_BYTE4 (Φ ul)<br>In the previous example, using a MAC address of 00-18-3E-01-AD-37: $(37)_{Hex} = (55)_{Decimal}$ Therefore Φ = 55 and the effective IP address is given by 192.168.1.55. Now, calculate Φ for your local device and update lines 184 and 189 using this value. | |
| **15.** | To test the device, one end of an Ethernet cable should be connected to the MX7cK board via the Ethernet Port (just above the J15 Debug USB connector), while the other end of the cable should be connected to the PC running Windows and the MPLAB X IDE. | |
| **16.** | Select the XC32-PIC32_ETH_SK_ETH795 project by right-clicking on the XC32-PIC32_ETH_SK_ETH795 project name in the Projects tab of the MPLAB X IDE and build the project by selecting Build. After many seconds, the message "BUILD SUCCESSFUL" should display in the Output window. | |
| **17.** | Run the project by right-clicking on the XC32-PIC32_ETH_SK_ETH795 project name in the Projects tab of the MPLAB X IDE and selecting Run. After many seconds, the message "Programming/Verify complete" should display in the output window. LD1 (on the MX7cK board) should be blinking which confirms the program is running. | |
| **18.** | Next, open Windows Command Prompt on the host PC running Windows by clicking Start and using the OS to search for 'cmd'. Select/open the Command Prompt from this search window. | |
| **19.** | Finally, in the command console, ping the MX7cK board at the configured IP address (used in step 14). For example, using the previous MAC address of 00-18-3E-01-AD-37 which results in an IP address of 192.168.1.55, the ping command would be accomplished by typing "ping 192.168.1.55" (without the quotations in the Command Prompt window) and pressing Enter. Do this for your local device by typing "ping 192.168.1.Φ" where Φ equals the decimal value of the last two digits of your local MAC address (same value calculated in step 15). If the command is successful, you will see "Pinging 192.168.1.Φ with 32 bytes of data:". If the test is successful, the device will reply with "Reply from 192.168.1.Φ: bytes=32 time<1ms TTL=100". Otherwise, replies like "Request timed out." or "Reply from 192.168.0.13: Destination host unreachable" means that communications have failed. The ping statistics will also be displayed once the ping command resolves which is another indicator showing whether the packets were successfully transmitted. | |

# 4 - TEST PLAN TEMPLATE APPROVAL

The undersigned acknowledge they have reviewed the P1.1.1 Test Plan Template document and agree with the approach it presents. Any changes to this Requirements Definition will be coordinated with and approved by the undersigned or their designated representatives.

Required Signatures:

- TA - Nahiyan Kaiser

Signature: _____ Date: _____

Print Name: _____

Title: _____

Role: _____

## REFERENCES

[1] K. Ferens, "ECE 3740 Systems Engineering Principles 1," 15 September 2001. [Online]. Available: http://ece.eng.umanitoba.ca/undergraduate/ECE3740/. [Accessed 16 September 2017].