Net Release Notes

MPLAB Harmony Integrated Software Framework

Release Information for Net

Release Notes

This topic provides the release notes for this version of MPLAB Harmony Net.

Description

MPLAB Harmony Net Version: 3.1.0 **Release Date:** December 2018

Software Requirements

Before using MPLAB Harmony Net, ensure that the following are installed:

- MPLAB X IDE 5.10 or later
- MPLAB XC32 C/C++ Compiler V2.15 or later
- MPLAB Harmony Configurator 3.1.0
- Harmony net repository, 3.1.0
- Harmony bsp repository, 3.1.0
- Harmony csp repository, 3.10.0
- Harmony core repository, 3.1.0
- Harmony dev_packs repository, 3.1.0
- Harmony mhc repository, 3.1.0
- Harmony mplabx_plugin repository, 3.1.0
- CMSIS-FreeRTOS repository, 10.0.1 for building the FreeRTOS projects (from www.github.com/arm-software/cmsis-freertos)

What is New and Known Issues

The following tables list the features that have been changed or added and any known issues that have been identified. Any known issues that have yet to be resolved were retained from the previous release.

MPLAB Harmony:

Feature	Additions and Updates	Known Issues
General	Nothing to Report.	• MPLAB Harmony has not been tested with C++; therefore this programming language is not supported. The "`-O1`" optimization level (or greater) is recommended when building any projects that include the MPLAB Harmony prebuilt binary (.a file) peripheral library. This is necessary so that the linker will remove code from unused sections (for peripheral library features that are not used). Alternately, you may select "Remove Unused Sections" in the General options for the `xc32-ld` (linker) properties dialog box. • EDBG is not supported in this release

Middleware and Libraries:

Feature	Additions and Updates	Known Issues

TCP/IP Library	**Additions:**
	Nothing to Report.
	
	Updates/Bug Fixes:
	• UDP:
	  • Fixed comments in the UDP header for - TCPIP_UDP_OptionSet() not providing 100% compatibility with the BSD setsockopt().
	  • Provided better explanation for some of the UDP_MULTICAST_FLAGS flags.
	• HTTP_NET:
	  • Added HTTP_NET connection events: open, close, timeout.
	  • Updated the HTTP app with the new SMTPC module changes. Old SMTP module is obsolete now.
	  • Added case-insensitive comparison for the HTTP headers. Added stricmp() helper.
	  • Added file open/close events.
	  • Fixed the legacy mechanism of using callback position. Cleared the callbackPos counter only when the dynamic variable print is called first time
	• Telnet:
	  • Better processing of the telnet options.
	  • Improved the telnet usage of the NET_PRES layer.
	• TCP:
	  • Updated the comments for TCPIP_TCP_IsConnected().
	  • Fixed bug in the calculation of the network shift mask.
	• tcpip_manager:
	  • Updated the processing of the network interface flags to avoid the MAC Initialize flag to remain set if the network interface is stopped before completing initialization.
	  • Fixed the selection of the DNS client as default only if the module is enabled.
	  • Fixed the tcpip_heap_external semaphore delete not checking that it was successfully created.
	• mDNS:
	  • Fixed the use of the wrong field to detect an authoritative reply. The field "qr" was used instead of "aa" when deciding for an authoritative answer.
	• IPv6:
	  • Fixed the routine converting a string to an IPv6 address.
	• tcpip_commands:
	  • Added missing Close() when MIIM operation failed.	

Board Support Packages (BSP):

Target Board	Additions and Updates	Known Issues
SAM E70 Xplained Ultra	New to Harmony 3	None
SAM V71 Xplained Ultra	New to Harmony 3	None

Drivers:

Feature	Additions and Updates	Known Issues
MAC Driver	**Additions:** • Added GMAC Ethernet Driver	None

Applications:

Feature	Additions and Updates	Known Issues
TCP/IP Demonstrations	**Added the following demonstrations:** TCP/IP demonstrations for SAM E70: • berkeley_tcp_client • berkeley_tcp_server • berkeley_udp_client • berkeley_udp_relay • berkeley_udp_server • snmpv3_nvm_mpfs • snmpv3_sdcard_fatfs • tcpip_client_server • tcpip_tcp_client • tcpip_tcp_client_server • tcpip_tcp_server • tcpip_udp_client • tcpip_udp_client_server • tcpip_udp_server • web_net_server_nvm_mpfs • web_server_nvm_mpfs • web_server_sdcard_fatfs TCP/IP demonstrations for SAM V71: • berkeley_tcp_client • berkeley_tcp_server • berkeley_udp_client • berkeley_udp_relay • berkeley_udp_server • snmpv3_nvm_mpfs • snmpv3_sdcard_fatfs • tcpip_tcp_client • tcpip_tcp_client_server • tcpip_tcp_server • tcpip_udp_client • tcpip_udp_client_server • tcpip_udp_server • web_server_nvm_mpfs • web_server_sdcard_fatfs	• The web_net_server_nvm_mpfs demo does not currently support encrypted connections. The wolfSSL third party package cannot be added to the application as an encryption provider. This will be added in the next release. • The tcpip_client_server application does not currently have documentation. The commands to use this demonstration are provided in the run time help available as console commands. This will be added in a future release. • When the File System service is used in a demonstration application, make sure to set ‘Maximum Simultaneous File Access’ to a number above 10. Especially the HTTP server needs to open multiple files simultaneously. • When the RTOS is enabled, make sure to include RTOS system API’s (like Time Delay functions) in application task to give other tasks the chance to execute. This can be achieved by enabling the ‘Use Task Delay’ from Application Configuration -> RTOS Configuration.

Development User Interface:

Feature	Additions and Updates	Known Issues
MPLAB Harmony 3 Framework Downloader	**New to Harmony 3**, in support of installing individual Harmony 3 components	

MPLAB Harmony 3 Configurator	**New:** 1. Project Graph added to user interface 2. Available Project Components and Active Project Components panels 3. Configuration Options Panel replaces MPLAB Harmony and Application Configuration Menu and provides options for the selected component	• Adding TCP/IP components to the project must be done using the TCP/IP Configurators in order to load the dependencies properly. Adding the TCP/IP components from the 'Available Components', without using configurator, might cause build issues. • Similarly, remove the TCP/IP components by using the TCP/IP Configurator for the respective layer • Do not add the TCP/IP components to the project by using both the TCP/IP Configurator and manually adding from 'Available Components'. • Do not delete the TCP/IP Configurators from project graph. • When multiple modules depending on the 'netPres' component are added to the project (like Berkeley, HTTP_NET, TELNET etc.), multiple 'netPres' nodes will be shown on TCP/IP STACK group box. But internally all these TCP/IP modules are using a single instance of the NET_PRES service. So only one instance of NET_PRES is needed.
MPLAB Harmony 3 Interactive Help		Interactive help using the "Show User Manual Entry" in the Right-click menu for configuration options provided by this module is not yet available from within the MPLAB Harmony Configurator (MHC). Please see the the "Configuring the Library" section in the help documentation in the doc folder for this module instead. Help is available in both CHM and PDF formats.

Release Contents

This topic lists the contents of this release and identifies each module.

Description

This table lists the contents of this release, including a brief description, and the release type (Alpha, Beta, Production, or Vendor).

Folder	Description	Release Type
\net\apps\tcpip\tcpip_udp_client	UDP client basic application.	Beta
\net\apps\tcpip\tcpip_udp_client_server	UDP client + server basic application.	Beta
\net\apps\tcpip\tcpip_udp_server	UDP server basic application.	Beta
\net\apps\tcpip\web_net_server_nvm_mpfs	HTTP_NET web server with NVM MPFS file system.	Beta
\net\apps\tcpip\web_server_nvm_mpfs	HTTP web server with NVM MPFS file system.	Beta
\net\apps\tcpip\web_server_sdcard_fatfs	HTTP web server with FATFS file system on as SD Card.	Beta
\net\apps\tcpip\wolfssl_tcp_client	wolfSSL TCP client basic application.	Beta
\net\apps\tcpip\wolfssl_tcp_server	wolfSSL TCP server basic application.	Beta
\net\apps\tcpip\berkeley_tcp_client	BSD TCP client basic application.	Beta
\net\apps\tcpip\berkeley_tcp_server	BSD TCP server basic application.	Beta
\net\apps\tcpip\berkeley_udp_client	BSD UDP client basic application.	Beta
\net\apps\tcpip\berkeley_udp_relay	BSD UDP relay basic application.	Beta
\net\apps\tcpip\berkeley_udp_server	BSD UDP server basic application.	Beta
\net\apps\tcpip\snmpv3_nvm_mpfs	SNMPv3 with NVM MPFS file system application.	Beta
\net\apps\tcpip\snmpv3_sdcard_fatfs	SNMPv3 with FATFS file system on as SD Card application	Beta

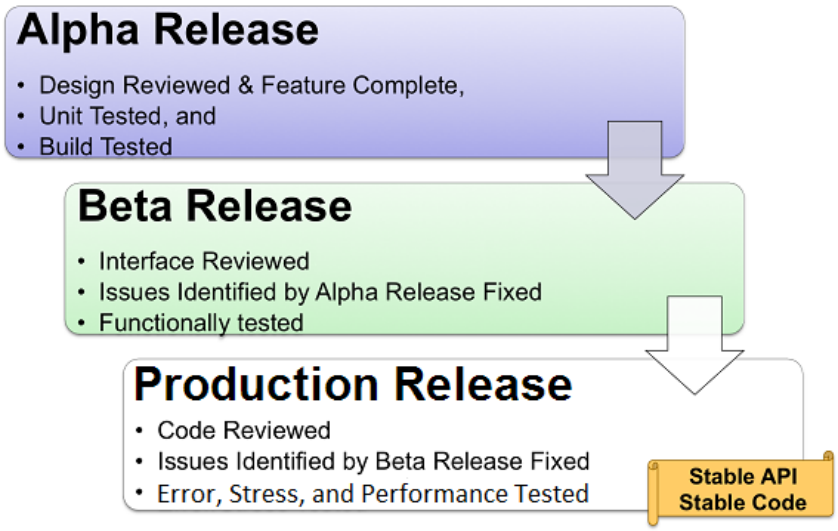
\net\apps\tcpip\tcpip_tcp_client	TCP client basic application.	Beta
\net\apps\tcpip\tcpip_tcp_client_server	TCP client + server basic application.	Beta
\net\apps\tcpip\tcpip_tcp_server	TCP server basic application.	Beta
\net\apps\rtos\freertos\tcpip_client_server	TCP + UDP client + server multi-threaded application.	Beta

Release Types

This section describes the release types and their meaning.

Description

MPLAB Harmony module releases can be one of three different types, as shown in the following illustration.



Alpha Release

An alpha release version of a module is usually an initial release. Alpha releases will have complete implementations of their basic feature set, they are functionally unit tested and will build correctly. An alpha release is a great "preview" of what a new development Microchip is working on and it can be very helpful for exploring new features. However, it has not gone through the complete formal test process and it is almost certain that some of its interface will change before the production version is released, and therefore, is not recommended for production use.

Beta Release

A beta release version of a module has gone through the internal interface review process and has had formal testing of its functionality. Also, issues reported from the alpha release will have been fixed or documented. When a module is in a beta version, you can expect it to function correctly in normal circumstances and you can expect that its interface is very close to the final form (although changes can still be made if required). However, it has not had stress or performance testing and it may not fail gracefully if used incorrectly.

Production Release

By the time a module is released in a production form, it is feature complete, fully tested, and its interface is "frozen". All known issues from previous releases will have been fixed or documented. The existing interface will not change in future releases. It may be expanded with additional features and additional interface functions, but existing interface functions will not change. This is stable code with a stable Application Program Interface (API) that you can rely on for production purposes.

Version Numbers

This section describes the meaning of MPLAB Harmony version numbers.

Description

MPLAB Harmony Version Numbering Scheme

MPLAB Harmony uses the following version numbering scheme:

```
<major>.<minor>[.<dot>][<release type>]
```

Where:

`<major>` = Major revision (significant change that affects many or all modules)

`<minor>` = Minor revision (new features, regular releases)

`[.<dot>]` = Dot release (error corrections, unscheduled releases)

`[<release type>]` = Release Type (a for alpha and b for beta, if applicable). Production release versions do not include a release type letter.

Version String

The SYS_VersionStrGet function will return a string in the format:

"`<major>.<minor>[.<patch>][<type>]`"

Where:

`<major>` is the module's major version number

`<minor>` is the module's minor version number

`<patch>` is an optional "patch" or "dot" release number (which is not included in the string if it equals "00")

`<type>` is an optional release type of "a" for alpha and "b" for beta. This type is not included if the release is a production version (i.e., not an alpha or a beta)

Note: The version string will not contain any spaces.

Example:

```
"0.03a"
"1.00"
```

Version Number

The version number returned from the SYS_VersionGet function is an unsigned integer in the following decimal format (not in a BCD format).

`<major> * 10000 + <minor> * 100 + <patch>`

Where the numbers are represented in decimal and the meaning is the same as described in Version String.

Note: There is no numerical representation of the release type.

Example:

For version "0.03a", the value returned is equal to: 0 * 10000 + 3 * 100 + 0.

For version "1.00", the value returned is equal to: 1 * 100000 + 0 * 100 + 0.

Index

R

Release Contents 7
Release Information for Net 2
Release Notes 2
Release Types 8

V

Version Numbers 8