

NET+OS 7.4.2 Updates

Last updated: 2/26/18

BSP Updates

Last updated 10/21/14

Warning: Please be aware that by downloading this patch you may need to now port your application forward to use the new BSP files. This can be done by simply creating a new project, copying your custom *.c and *.h files to the new project, and then porting any changes made in the old projects BSP folder to the new projects BSP folder. Be sure to check your Makefiles as well.

Version 1.0

Serial Driver - case 30221

Removed extra stop bit that was added between characters.

RTS was incorrectly being deasserted for 1us before transmitting each character in RS485 mode. The serial driver was incorrectly using the CPU clock instead of the PLL for baud rate calculations.

Serial Driver - case 30049

serial_channel() now returns minor as the channel number.

max_serial_ports() now uses SERIAL_MAX rather than a hard coded value.

Serial Driver - case 29973

Fixed select() implementation. We did not notify the Treck stack about event changes. Added ioctl option TCGETSIGCHANGE

USB Driver

Updated ISR to call narm_read_reg and narm_write_reg.

See latest API Reference for updates to naUsbDevQueue about zero-length packets.

Wireless Driver - case 29954

Updated Japanese channels for 802.11a

Wireless Driver

Fixed Cisco LEAP+WEP problem.

SPI MMC Driver - case 28537

Fixes to SPI SD card driver to address SD cards that do not respond for various reasons. (usually timeouts)

Version 1.1

Wireless Driver - case 30378

Fixed a problem that could cause the WIEM9210 to sleep forever when an application changes duty cycle time on the fly.

Fixed a problem with high packet loss on the WiWM9210 using AES on b-g WLAN with protection enabled. Protection mode allows a mix of B and G devices on the same channel. It sends a CTS frame (at a B rate) before every G frame. The CTS frame tells B-only devices to stay off the air until the G frame is complete. Protection mode is enabled by the access point. The problem our driver had was that the CTS and the DATA frames are not in the same FIFO so the AP thought they were invalid.

I2C – case 30378

In customizeBootloader.c, we overwrote the preconfigured GPIO line to check forced recovery status. This would cause problems with I2C when run from flash. We now restore the GPIO line after checking status.

Version 1.2

Error building NET+OS examples caused by missing file. Added usbreg_def.h

Version 1.3

Fixed a problem that would cause cout to stop working after any change to the dialog was made. Case 30472

Changed error checking code for BSP_GPIO_MUX_IOP_1_CAN_TXD and BSP_GPIO_MUX_IOP_1_CAN_TXD in gpio.h to allow the 2nd alternative path, which is a valid choice.

Added A band channels (52, 56, 60, 64, 100, 104, 108, 112,116, 120, 124, 128, 132, 136, 140) to Japanese support.

Inside the routine SetChannel() there was missing a setting of the global_channel. Also added the file mac_11j.o to Makefile.files (case 30569)

Version 1.4

Corrected bug in NAMCWait which declared the timer handle to be static, thereby making it possible for one thread to alter the variable while a second thread is using it if both threads call into the function simultaneously.

Corrected dialog behavior when APP_USE_NETWORK_TIME_PROTOCOL was TRUE and the primary NVRAM SNTP address was 0.0.0.0. In that case, the dialog never displayed the SNTP addresses and the operator would not know they can update it at that time.

Fixed incorrect size used in memcpy call in na_pton.

Version 1.5

Changed the receive gain table for airoha chips.

Version 1.6

Implemented max gain control to improve receive sensitivity.

Version 1.7

Fix to update the status when we set the tx power. `wln_get_status` was not returning the correct value for tx power.

Version 1.8

Fixed alignment issue in `MacBuffer` that could cause Wi-Fi to crash when using Green Hills Tools.

Version 1.9

Fix for Ethernet receive stall after a soft reset under heavy network load on ARM9 based chips. Case: 29694

Masked off interrupt for 9215 & 9210. This is needed since the hardware doesn't handle heavy nesting of interrupts correctly. Case: 30345

Version 2.0

Implemented automatic adjustment of the MAX AGC setting based on the receive strength of received frames.

Added firmware to enable the second DC block HP bit in the transceiver. Case 31462.

Version 2.1

Wireless change that will wait for an ack after sending a `probe_response` before setting the tx state to idle.

Moved `min` and `max` from `fwtypes.h` to `minmax.h`. `Define` was causing build errors in C++ when using STL containers; `vector`, `map` and `string`. Case 31471.

Version 2.2

Fixed problem where data overflow error was occurring when the camry serial driver is flow controlled. When the driver receives a XOFF it disables the transmitter, at this time it is unable to send XOFF to the peer if need be, resulting in data overflow. Case: 31720

Updated readme files to match FIM Application Kit H/W Reference.

Fixed Pll clk calls and updated FIM code in onewire driver.

Updated FIM1 1WIRE and FIM0 UART settings in `bsp_iop.h` for 9210 based platforms.

Fixed tx fifo underrun problem on `Connectcorewi9c_a`.

FPGA file modified to utilize the 4096 byte depth of the `recv fifo` on `Connectcorewi9c_a`.

Fixed tx fifo reset problem on piper with new `macassist` code

Changed RF register `0xC's` setting from `0xA3` to `0x2B` for `al7230`

Limited `Connectcorewi9p9215` operational max PSK/CCK power to 16dbm

Version 2.3

Moved `naValidateIpv6Information` to `nainet.c`, renamed `validateIpAddress` to `naValidateIpAddress`, Updated API Reference to add documentation to `naValidateIpv6Information` (case 31920)

Fixed a problem caused when there was a low heap situation. (case 31856)

Changed the `MAX_CODE_SIZE` for the WiEM, WiSP, WiEM9210, and CCWi9P9215 platforms to 3.5M. Note that all these platforms already have the

APP_MAX_SIZE_IN_FLASH set to 1856K. It's only natural that the MAX_CODE_SIZE (in RAM) should be about double the SIZE_IN_FLASH due to the compression mechanism. (case 31850)

Now checking the NVRAM backup copy once, every time the unit comes up, to address the possibility of a corrupted/stale backup sector. (case 31747)

Version 2.4

Updated file system interface files to use an event flag instead of polling for the result to speed up file transfer. (case 32017)

Version 2.5

Corrected a problem that could cause the I2C bus to hang if a reset is issued during active I2C communications. (case 32488)

Corrected a problem with size check when downloading backup image, and increased the backup image size in the ConnectME and ConnectME9210 to 512K. (case 32530)

Version 2.6

Set RTC device count to 0 in all ARM7, ns9750 and ns9210 based platforms. (case 32478)

Fixed a problem in the Ethernet driver. The stack can sometimes send the driver Ethernet packets that are spread out across more than one buffer. Apparently this happens a lot when using IPSEC, and often 3 buffers are sent for a single packet. The driver was not handling that well, and was getting confused.

Fixed customer reported rounding error in NATimerPeriod2Raw. Error was about (5/100) of one percent. (case 32430)

Fixed a memory leak in firmware download process. (case 32636)

Fixed problem with lock mechanism in spimaster routine.

Version 2.7

Wireless Updates

Now allow duty cycling while scanning.

Added LINKUP and LINKDOWN events. Updated mwifimac for LINKUP and LINKDOWN event and added naWlnSetEvent for wpa to setup linkup/linkdown event and startMultiScan for multiple channel mask scan.

Added naEkahauSetBatteryCallback prototype and naEkahauSetStatusCallback API.

Added naWlnIsWirelessUp function to see whether wireless is ready to transmit. We now make a call to naWlnIsWirelessUp if no LINKUP event is received from the wifi driver to verify wifi is up before sending ELP.

wlnNullFramePtrPS and wln_make_null_frame_ps are added for background scans while staying associated. The unit sends a NULL frame with the power-save bit set, so that the AP knows to buffer data for us. The unit sends a NULL frame with the power-save bit clear when the scan is done.

Multipath problem fixed by resetting the transceiver before changing channels.

Fixed problem when in A band and in ad-hoc mode that caused the unit to send out tons of RTS.

Fixed (Authentication type = WEP (802.1x))

Fixed a bug that was causing units to occasionally skip association to the nearest AP because the MacTimer() wasn't allowing enough time for a beacon to come in before it tried the next AP.

Version 2.8

Wireless Updates

FPGA file updated, for connectcorewi9c_a platform.

For AI2236 transceiver -

Config 7 = RSSI = default value

Original Setting: 0x05B688

New Recommended Setting: 0x0A8688

Note that the Airoha-recommended setting is dependent on the supply voltage: 0x05B688 for 2.8v, 0x0E4688 for 3.0v. The Piper WiEm voltage is actually around 2.88v, so the new value is consistent with the Airoha-recommended values (between the two settings). (case 32800)

Serial Updates

The multiplexor that controls the output enable for gpio[29] is being controlled the control register for gpio[28]. The work around and limitation is that gpio[28] and gpio[29] must always be configured to the same functional mode. (case 1284328)

Version 2.9

Wireless Updates

Fixed problem that caused WiEM 9210 to crash when naPdSleep() was called. (case 33148)

Fixed problem where after waking up, the device did not detect it was disassociated from its AP (case 33149)

Changed wireless driver so that when we fail EAP FAST authentication two times in a row, we will delay 1 minute before trying again. We do this because the EAP FAST negotiation is CPU intensive and interferes with other processes. (case 33239)

Version 3.0

Increased read and write delay on CS1 to support 100ns flash parts on Connectcore9p9215 and Connectcorewi9p9215 modules.

Changed the CLI startup mechanism so each service can only be loaded once. Updated to allow CLI open for serial CLI immediately after dialog, so that a wireless only driver can make use of this to configure its parameters. Telnet and SSH CLI opens take place after the stack is loaded. (case 33290)

Fixed an optimization problem on the fixes for serial port D. (case 1284328, 32984, 32985)

Now clear serial interrupt before entering sleep mode. (case 33143)

Now use the WAKE UP button for forced recovery for the cc9p9215, ccwi9p9215 and the ns9210 modules on the cc9p9215 carrier board. (case 33388)

Corrected problem that cause the Image not to execute properly when WriteToFlash is set to 'No' in bootldr.dat (case 33386)

Calling wlan_set_duty_cycle() on the fly could stall the wireless driver. (case 33142)

Version 3.1

Updated cli to describe channel index and real channel number. (case 33515)

Limited the Wi9P wireless driver to the following max rates on some A channels:

4920-4980: 18Mbps 6dBm

5040-5180: 24Mbps 6dBm

5190-5500: 48Mbps 6dBm

5520-5620: 48Mbps 9dBm
5640-5825: (All) 7dBm

Version 3.2

IamCallback now returns the gateway address based on interface. This corrects problems displaying the wireless gateway in systems with more than 1 interface. (case 34058)

Fixed compiler warnings (case 33872)

Updated I2C return codes (case 33952)

Fixed problem that would cause FIM UART to stop sending data (case 34179)

Version 3.3/3.4

Added NA_ETH_SET_PACKET_TRAILER_SIZE. This constant can be used to enable a packet trailer in Ethernet packets. It is normally set to 0 and a trailer is not used.

Updated a condition check in tx index not to cross boundary. This was causing failures when using FIM1 UART. (case 35008)

Updated the NS9215 SPI Master driver to implement additional SPI ports using the FIM processors. This was done to provide a work-around to problems in the internal NS9215 SPI port.

Added new roaming API to the Wi-Fi driver. See API Reference for details.

Updated Wireless Powersave method.

Updated MAX_CODE_SIZE from 2M to 3M on ConnectME platform.

Added Support for WPA2 Mixed Mode. (cases 33127 and 34393).

Fixed IPv6 dialog configuration issue. (case 34923)

Version 3.5

Created new error code NASTATUS_SPI_HARDWARE_ERRATA which will be returned if the user attempts to configure a mode that the NS9215's internal SPI port cannot support.

Added new APIs to CAN Driver. See API Reference for details. (Case 26317, 32022, 34921)

Version 3.6

Corrected behavior during simultaneous file access. (Case 36061, 34521)

Fixed uninitialized variable in spi_read_write. (Case 36106)

Added presence signal start and stop timing. Updated FIM to wait and detect presence signal within the start and stop times. Fixed close function to put the GPIO pin back default. (Case 35765)

Moved h/shared/nastatus.h to Common Updates Version 1.0

This update is dependent on Common Updates Version 1.0

Version 3.7

Possible serial transmit problem fixed by clearing transmit flow state on initialization during the device open() call. (Case 36318)

Implemented work-around to a bug in power save which prevented the wake on Ethernet feature from working reliably on 100 Megabit networks. The work-around is to make sure the CPU clock is no less than 25 MHz. The software will now set the CPU clock divisor before going to sleep so that it is at least 25 MHz. (Case 35867)

Several changes to the FIM SPI support

1. The MISO and CS pins. This was done to make it possible to support both SPI and serial port A on the ConnectME-9210. In this case, SPI will be run without CS, but that is okay for most devices if only one device is supported.
2. SPI clock rates are now calculated in a different way. The new method allows the clock rates to be selected with greater granularity at higher speeds at the cost of not supporting the lower speeds.
3. New functions, `NASpiWriteImmediately` and `NASpiReadImmediately`, have been implemented that allow users to break up a SPI transfer into separate write and read operations. The reason for doing this is that combining them as `NASpiLowLatencyReadWrite` does causes the calling thread to stall while the entire transfer takes place. By performing the operations separately, the calling thread can do some processing while the transfer is taking place.
4. The SPI driver will now power down the FIM port while the SPI bus is idle. This will only happen on the FIM SPI ports.
5. Added a new API, `NASpiEmptyReadFifo`, into the FIM SPI driver that removes any data pending in the read FIFO.
6. Updated the API to include a write only mode.
7. Corrected calculation of FIM SPI clock rate.

Version 3.8

Fixed type in `NASpiMasterFimAddDevice` that could prevent the second SPI port from working. (Case 36807)

Corrected platform support for CANBUS

Fixed a bug where wireless devices may not respond to RTS packets if the data rate drops to 1 Mbps and the short preamble option is enabled. This can cause the device to disassociate or fail authentication. (Case 36484)

Removed the serial select thread, this thread is not used and is left over from the Fusion stack. This code was also opening two UDP ports unnecessarily. (Case 36437)

Fixed a buffer overflow in the wireless dialog, the overflow was overwriting stack variables causing the encryption parameters to be set back to default when modifying the dialog. (Case 36485)

Added `sendBreak` process. Updated `serl_monitor` to send interrupt command to FIM to set TX low for `sendBreak` and restore TX after given duration has expired. The FIM code was also updated for `sendBreak` command. Updated actual bytes written TX instead the number of bytes sent to FIM which includes stop bits & parity. (Case 31247)

Version 3.81

Fixed the `InitTimer()` to ensure delays between radio accesses. (Case 36555)

Fixed a problem in `ParseTimElement()` which could cause us to send spurious null-data frames when in low power mode.

Added LED support to Connect Wi-EM9210 platform.

Version 3.82

Fixed mis-matched index for `tx_in` and `tx_out` when IOP is done transmitting and the serial buffer is full of data. Updated `TCSETA` in `ioctl` to flush Tx and Rx queues. Updated IOP to flush DMA buffers in ISR, and reset DMA index from the register. When setting CA bit to flush DMA

buffers, `naIopDmaFlushQueue` must wait until either all DMA buffers have been cleared in the ISR or the DMA state is idle (no interrupts pending). (Case 37914)
Swapped CS and MISO so that users can use UARTA and FIM SPI by configuring SPI without CS on ConnectME9210. (Case 37833)
Added support for the 3033 variant of the cc9p9360 module. This variant runs at 155MHz instead of 177MHz. (Case 34834)

Version 3.83

NET+OS serial driver should come up with no flow control. The serial driver was initialized to use SW flow control while loading the driver. This could halt the transmitter if it received a serial data frame with 0x13 in it.

Corrected typos and incorrect paths in IRQ1 and IRQ2 (Case 38288, 38337)

Fixed a problem when in ad-hoc mode and using "search" for channel. (Case 38469)

Fix problem in spectrum management (802.11h) and multi domain (802.11d) (Case 43043)

Fixed error in roaming parameters. (Case 43278)

Version 3.84

Changed read and write access to 32-bit words in scratchpad driver. Updated `nascratchpad` example readme file. (Case 44550)

Removed calls to `naIopPause()` which was causing the SPI FIM clock to get stuck high, and fixed `naIopPause()` and `naIopResume()` for FIM 1. (Case 43026)

Fixed problem that was causing TFTP recovery to fail due to DHCP offer handling error. (Case 44186)

Set `ENOENT` `errno` instead of `none` in file system interface when open non-existing file to read.

Version 3.85

Improved power save mode, improved beacon handling and additional communication fixes. (Case 45536, 45402, 45183)

Version 3.86

Fixed coding error to ensure the `promiscuousMode` bit is set if in monitor mode. (NET+OS-13)

Version 3.87

Fixed a problem with setting relative time alarm in RTC driver on 02/29 in a leap year. (NETOS-5)

Added code to skip empty calibration entries for the newly manufactured modules, which eliminated 4.9 GHz band. (NETOS-28)

Replaced calls to deprecated function `NATimerConfigure` with `NATimerConfig` in the `natimer` sample. (NETOS-7)

Fixed SSH security problem that allowed users to login into the module using SSH without user name/password. (NETOS-17)

Version 3.88

There was a bug such that the utility function `CoreDump` produced output on ARM7 processors, but produced nothing on ARM9 processors. It was found that the `CoreDump` code was executed, but no output was produced on the serial line through `bsp_printf`. The fix was to call function

setupSimpleSerial at the beginning of function CoreDump. This ensured that all necessary GPIOs were in a known state prior to running CoreDump. (NETOS-74)

Version 3.89

In function na9215A2dConfigure the mask used against the n value (sampling rate) was 0x3f. The problem is that the register section is 10 bits and 0x3f is only 6 bits. Mask should be 0x3ff (10 bits) (Case: 00194980)

Common Updates

Last updated 10/18/10

Version 1.0

Moved h/shared/nastatus.h to Common Updates Version 1.0
The BSP, TCPIP and SNMP updates are all dependant on this file.

Version 1.1

Modified to accommodate changes to the FIM SPI driver

Documentation Updates

Last updated 11/18/14

Version 1.0

Updated naUsbDevQueue regarding zero-length packets.

Version 1.1

Added support for IPv4 filtering.

Version 1.2

Updated USB Host error codes.
Updated getaddrinfo and ethernet multicast in bypass mode.

Version 1.3

Corrected description of serial and SPI signals.

Version 1.4

Added new API naIpsecIkeSendSaLifetimes which lets users determine if SA lifetime information should be included in the initial IKE message.

Version 1.5

Updated information exception codes.

Version 1.6

Updated information on IPv4 automatic collision detection.

Version 1.7

Added an API that allows users to install callback functions which are invoked by the kernel whenever it enters an idle state, or switches to a new thread.

Version 1.8

The following white papers have been added to the Documentation Updates for NET+OS 7.4.2. They can be found under the Documentation\white_papers directory in your NET+OS 7.4.2 installation tree.

Application Note: Ajax programming using Digi's Advanced Web Server

Application Note: Combining CLib and Native File System APIs

Application Note: Controlling Page Changes in AWS

Application Note: Creating SSL Certificates

Application Note: Javascript Forms Processing with AWS

Application Note: Using the Advanced Web Server

Application Note: Adding custom MIBs to NET+OS 7.x projects

Application Note: Real-time Device Monitoring Using AWS

Application Note: Accessing NET+OS's SNMP agent using MGSoft's SNMP management Software

Application Note: Enabling SNMPv3 Security Features in NET+OS V7.X

Application Note: Generating HTML content on the fly with AWS

Application Note: Adding DHCP options to NET+OS 7.x

Version 1.9

Updated API Reference for new link layer status callback flag for serial link down (PPP_LL_LINK_DOWN)

Added documentation to naValidateIpv6Information.

Version 2.0

Updated API Reference to clarify the usage of the NOR Flash functions. These functions were shown available for all platforms but are not available on NAND-based platforms (CC9P 9360). This is now clarified, plus some clean up indicating the topic is Flash and subtopic is either NOR or serial. Case 32268.

Version 2.1

Added LINKUP and LINKDOWN events. Updated mwifimac for LINKUP and LINKDOWN event and added naWlnSetEvent for wpa to setup linkup/linkdown event and startMultiScan for multiple channel mask scan.

Added naEkahauSetBatteryCallback prototype and naEkahauSetStatusCallback API.

Added naWlnIsWirelessUp function to see whether wireless is ready to transmit.

wlnNullFramePtrPS and wln_make_null_frame_ps are added for background scans while staying associated.

Version 2.2

Now use the WAKE UP button for forced recovery for the cc9p9215, ccwi9p9215 and the ns9210 modules on the cc9p9215 carrier board. (case 33388)

Version 2.3

Added description of channel index as it relates to the wireless channel numbers. (case 33515)
Added white paper- Application Note- Correcting Linker errors in DIGI ESP

Version 2.4

Added link to naSetNotificationUserName (case 33298)
Added white paper- Application Note- Taking Advantage of Digi's Advanced Web Server's Repeat Group Feature

Version 2.5

Added reference to NAgetGPIOpinConfig. (case 34240)
Clarified wln_scan_bss structure -> channel description. (case 33226)

Version 2.6

Added NA_ETH_SET_PACKET_TRAILER_SIZE. This constant can be used to enable a packet trailer in Ethernet packets. It is normally set to 0 and a trailer is not used.
Updated the NS9215 SPI Master driver to implement additional SPI ports using the FIM processors. This was done to provide a work-around to problems in the internal NS9215 SPI port.
Added new roaming API to the Wi-Fi driver. See API Reference for details.
Added Support for WPA2 Mixed Mode. (cases 33127 and 34393).

Version 2.7

Enhanced documentation related to the SPI CAN Driver. Added new APIs. (Case 26317, 32022, 34921)
Added white paper- Application Note- Developing CGI-based AWS Applications Using Digi's NET+OS Development Environment

Version 2.8

Added white paper- Application Note- How Changes in MS Internet Explorer Version 8 Will Affect Web Pages Containing Input type=file

Version 2.9

Added documentation for snmpTargetMIB and snmpNotificationMIB. (Case 36130)
Changed description of NASpiSlaveBufferType in ApiReference to state that the SPI receive buffer should be 32-byte aligned. (Case 36111)
Updated ConnectME ConnectWiME ConnectME9210 HW Reference to Rev J.

Version 3.0

Updated FIM SPI clock rates
Added new FIM SPI APIs
Updated NaSnmptTrap_t structure.
Updated NS9210/NS9215 Errata

Version 3.1

Added sendBreak process. Updated serl_monitor to send interrupt command to FIM to set TX low for sendBreak and restore TX after given duration has expired. The FIM code was also updated for sendBreak command. Updated actual bytes written TX instead the number of bytes sent to FIM which includes stop bits & parity. (Case 31247)

Updated autodoc comments for naIpsecPolicyContentSetRules to explain problem with using IPSEC transport mode and "don't fragment" option. Recommended use of IPSEC tunnel mode instead. Also described a work around.

Version 3.11

Updated NS9210/NS9215 Errata

Added NS9210/NS9215 FIM SPI Port Implementation Reference

Version 3.12

Corrected FIM SPI MISO and CS signals in API Reference

Version 3.13

Added APIs to allow users to limit the amount of heap the TCP/IP stack is allowed to allocate. Set the default limit to be half the heap. (Case 40091)

Improved documentation for naIsrSetFiq to describe the limitations of FIQ service routines, and to describe the other configuration settings that need to be changed to support an FIQ service routine.

Version 3.14

Fixed incorrect autodoc description of fLanguagePtr in CGI structure. (Case 41908)

Version 3.15

Application Note: Switching between DHCP and Static IP address Acquisition in NET+OS

Application Note: Handling multiple submit buttons in NET+OS development environment web-based applications

Application Note: JavaScript and pbuilder compression

Version 3.16

Updated the NS9210, NS9215, ConnectSP and Wi-SP, ConnectME Wi-ME ME9210, and Connectcore 9P and Wi-9P 9215 Hardware Reference Manuals

Version 3.17

Application Note: AWS, CSS and id selectors

Application Note: Catching Crashes in NET+OS using ESP

Version 3.18

Updated the Connectcore 9P and Wi-9P 9215 Hardware Reference Manual

Version 3.19

Removed erroneous note from FCConnect.

Version 3.20

Removed Constants for SSL Version which became obsolete upon mitigation of the POODLE SSL attack. (NETOS-105)

FileSystem Updates

Last updated 08/16/12

Version 1.0

Fixed a crash when issuing FTP get command twice for a non-existent file name on a NAND flash file system. Case 31491.

Version 1.1

Fixed NULL pointers. Case 31558.

Version 1.2

There was a weakness in the file system such that when the file size exceeded the size limitation of the block size, a hang could occur. Added check for file size relative to block size (512 blocks limit file size to 8M)

Version 1.21

There is a section of each NAND block which is used to store information about that block such as whether the block is good or bad, the ECC for the block, and some other data. There is no standard for the format of this field. NET+OS YAFFS formats the field one way, and other file systems differently. The problem we ran into is that U-Boot expects to see good blocks marked with two bytes of 0xff at the beginning of this field, but NET+OS YAFFS was only writing one 0xff. This caused U-Boot to think that some NET+OS blocks were bad. This was only a problem on NAND parts that support 2K blocks. (Case 37837)

Version 1.22

Added conditional build based on BSP_YAFFS_DRIVER_ENABLE, and corrected unconditional inclusion of yaffs in libfilesystem.a (Case 41204)

Flash Updates

Last updated 02/26/18

Version 1.0

Added support for M29W640GH flash parts.

Version 1.1

Added support for SPANSION S29AL032D90TFI040 4M Flash type for ConnectCore9p9215 and ConnectCorewi9p9215 platforms

Version 1.2

Added support for Fujitsu Micro MBM29LV650UE Flash type for ConnectCore7u_a platform

Version 1.21

Added support for STMicro NAND512W3A2D NAND flash which is used in the latest versions of the cc9p9360. (Case 37388)

Version 1.22

Added support for Flash types Spansion S29GL032N and Eon EN29LV320B for connectcorewi9p9215_a and onnectcore9p9215_a

Version 1.23

Added a new Flash function called quick_identify_flash() to work around the problem of resetting while Flash is in the autoselect state. (Case 45001)

Added support for SPANSION S29GL064N Flash type for ConnectCore7u_a platform

Version 1.24

Removed support for Macronix MX28F4000 Flash for the ConnectME and the ConnectME9210. This Flash type does not have an exit auto select mode command which caused the quick_identify_flash() routine to crash. This Flash type is not on the approved list for either product.

Version 1.25

Fixed the BSP size problem with the connectcore9p9215_a by removing support for AMD AM29LV160T Flash and removing some unused code in naflash.c

Version 1.26

Added support for SPANSION S29PL032J Flash part.

The new Flash part has been added for following platforms:

- connectcore7u_a
- connectcore9c_a
- connectcore9p9215_a
- connectcore9p9360_a
- connectcorewi9c_a
- connectcorewi9p9215_a
- connectem
- connectme
- connectme9210
- connectsp
- connectwiem
- connectwiem9210
- connectwime
- connectwisp
- ns9360_a
- ns9750_a
- ns9210_a
- ns7520_a

FTP Updates

Last updated 12/08/11

Version 1.0

Fixed a problem where we returned the wrong error when select times out due to no data on the data socket. (case 31281)

Version 1.1

Fixed problem that caused FTP to fail when the accepted socket = 0. (case 33412)

Version 1.11

Fixed incorrect response to EPSV command. (case 1326897)

GHS Updates

Last updated 11/12/12

Version 1.0

Resolved time zone issues. (case 1345341)

PPP Updates

Last updated 11/02/09

Version 1.0

Changed PPP read thread to wait on serial select() for incoming read data instead of polling the serial driver for read data. (case 29900)

Fixed error where fd=0 was not allowed, causing an error whenever we wrapped around after 128 sockets were used. (case 30832)

Version 1.1

Fixed a race condition in PPPCloseInterface().

Version 1.2

Added new link layer status callback flag for serial link down (PPP_LL_LINK_DOWN)

Passed the new status to application level so that it can close the session gracefully, and updated the application files to handle this condition.

Updated API Reference (case 31873)

Version 1.3

Reworked a condition check in PPPCloseInterface() that hits when close is called before open completes. This check was incorrectly determining the interface state based on the current PPP link level state.

Realport Updates

Last updated 05/26/09

Version 1.0

A pointer to the realport configuration structure is passed as a parameter to a threads operation function which uses that pointer to create another thread. If the configuration structure is a local variable on the stack, we have a problem after the first call returns. The solution is to save a copy of the configuration structure when RealPortStartServer is called. Case 30485

SNMP Updates

Last updated 04/14/14

Version 1.0

Calls to function naSnmpGetSysName() fail consistently with a return code of NA_SNMP_NO_MEMORY due to a typo in the source code. The function now returns NA_SUCCESS on success and NA_SNMP_NO_MEMORY on failure. (case 30083)
Added fflush() after fputs() to flush out the STDOUT writes in the SNMPv3 example apps CLI.

Version 1.1

Applied fixes to handle large sub ids while processing get requests, and for handling requests with bad lengths.

Applied changes for conceptual tables.

Version 1.2

Applied fixes to prevent failures while adding V3 Users. (case 30940)

Added namib3 example application.

Fixed write access in nasnmpv3 example application.

Version 1.3

Added fix from Treck to return the correct security level in a NotInTimeWindow SNMP report. (case 29818)

Version 1.4

Fixed viewing USM table and several other SNMPv3 tables. (case 31929)

Fixed adding users before SNMP agent has started. (case 31941)

Version 1.5

Fix for crash caused by the mismatch of TCPIP and SNMP libraries with and without IPSEC. (case 33312)

Version 1.6

We get the following linker error: undefined reference to 'tfProcessUnsignedInt' when unsigned ints are included in a MIB. Enabled TM_USE_PROCESS_UINT in SNMP agent to resolve issue. (case 33658)

If only one SNMPv3 user exists and it does not have an authentication protocol, that user will be denied access to the SNMP agent. Fixed access for user with no authentication protocol. (case 33851)

Version 1.7

SMIDUMP.EXE

Fixed invalid indices generated by the MIB compiler. (case 34053)

MAKEINS.EXE

Fixed a problem with the instructions filename which could cause runtime errors with some MIBs. (case 34054)

Version 1.8

Added Error code for naSnmpDeleteTrap, when deleting 1st entry. (Case 35853)

Added snmpTargetMIB and snmpNotificationMIB API and callbacks.

This update is dependent on Common Updates Version 1.0

Version 1.9

Changed the nasnmpd example to use enterprise OID array of NaSnmpTrap_t structure. Changed autodoc for NaSnmpTrap_t structure. (Case 36249)

Version 1.91

Updated MIB-II extensions for IP, TCP and UDP to bring them in sync with Treck Agent.

Version 1.92

Updated MIB-II extensions for IP, INET-ADDRESS and UDP to bring them in sync with Treck Agent.

Updated the SNMP agent to use the short format for ASN1 item lengths when possible.

Previously the old code always encoded the length as 3 bytes. Now it will be encoded as 1 byte if possible.

Version 1.93

Fix for snmp replying to a different IP address, than in received request (IPv6) (Case 43127)

Version 1.94

Fixed MIB compiler hanging problem, when compiling ITU-MIB.mib (rfc 3877) (Case 44380)

Report correct Ethernet link speed in SNMP ifSpeed (Case 41612) – **Requires TCPIP fix v1.47.**

Version 1.95

Replaced direct malloc() and free calls() with tm_snmp_malloc and tm_snmp_free in NET+OS snmpv3 api. (NETOS-39)

SNTP Updates

Last updated 04/26/12

Version 1.0

Fixed memory leak when calling tzset. Now properly freeing previously allocated memory. (case 34601)

Version 1.01

Back ported UDPsntp.c from netos 7.5 to 7.4. The 7.5 UDPsntp.c has correct checking on na_pton error return status, and select with binary back off algorithm timeout is called after sendto regardless of the error return status. (Case 39128)

SSH Updates

Last updated 01/27/10

Version 1.0

This change fixes a memory leak in ssh_connection_receive_channel_request().

Version 1.1

This change sets the TCP_NODELAY flag on all sockets that it opens. The nassh example application has also been updated with this change. (case 32879)

SSL Updates

Last updated 11/18/14

Version 1.0

This change fixes a memory leak when using naSSLNgConnect. About 1500 bytes of memory would be lost every time we made a client connection. (case 30656)

Version 1.01

Updated https_daemon function to start tls_server_http with correct variable and thread. After the maximum tls_server_http threads had been started, it sometimes reset incorrect connection info and restarted the tls_server_http thread. Case 31362.

Version 1.02

Fixed memory leak in tls_handshake_protocol_receive_certificate_request. The function calls d2i_X509_NAME to extract data from the input stream. d2i_X509_NAME returns the data in an allocated buffer which tls_handshake_protocol_receive_certificate_request was not freeing. Fixed it by having the function free the buffer when it finishes with it. Case 34780.

Version 1.03

Removed erroneous comment from nasslclient example root.c file.

Version 1.04

HTTPS transactions were found to be quite slow. A recv call in function recv_with_timeout was performing non-blocking recvs without a select and just performing a one second sleep between recv calls. A select call was added and this reduced the amount of time sleeping. This greatly reduced the time to complete an SSL handshake transaction.

Version 1.05

Mitigates the Poodle SSL attack by blocking any SSLv3 connection attempts. (NETOS-105)

TCPIP Updates

Last updated 04/14/14

Version 1.0

Added support for IPv4 filtering.

Disabled interrupts around recvEventEntryPtr->eventCount in notifyInterfaceIsr. This fixed synchronization of receive event count for Ethernet.

Added code to clean up after an Ethernet or wireless driver could not send it.

In Ethernet code just added statistics counters that are normally #ifdefed out.

In TCP benchmark, moved WSACleanup to the end and added a second delay to let the TCP connection to close gracefully before WSACleanup to avoid connection resets with slow connections. Case 30288.

Fixed PPP issue that allowed packets to be routed after interface was closed. (case 30789)

Version 1.1

Added support for the AI_PASSIVE flag in getaddrinfo. This applies to GNU only.

Aligned Ethernet receive DMA buffer descriptors on 32 byte boundary. (case 30848)

Version 1.2

Fix for naIamRelease not restarting DHCP. (case 30553)

Added new API naIpsecIkeSendSaLifetimes which lets users determine if SA lifetime information should be included in the initial IKE message. (case 32302)

Corrected problems in naipsec_ike example application that prevented it from setting its network address correctly, and made it easier for users to set the VPN peer IDs of both the local device and the remote device. (case 32303)

Version 1.3

Include fix from Treck to address IPv6 Ready Logo failure for Test v6LC.1.2.10: "Unrecognized Routing Type - Intermediate Node" (case 32490)

Version 1.4

Fixed the delay between DHCP retransmissions to be compliant with RFC 2131, the delay now starts at four seconds and increases exponentially until 64 seconds. (Case 35938)

Moved h/shared/nastatus.h to Common Updates Version 1.0
This update is dependent on Common Updates Version 1.0

Version 1.41

Updated tfCheckSetNetmask to allow subnet masks in which some of the first 8 bits are not set. Previous versions required all of the first 8 bits to be set (Case 37043)
This update is dependent on Common Updates Version 1.0

Version 1.42

Implemented fix provided by Treck that allows the stack to handle Classless Inter-Domain Routing (CIDR) correctly. Previous versions would not be able to route between machines at 25.0.0.50 and 24.0.0.50 with a subnet of 248.0.0.0. (Case 37043)
Fixed problem where we were freeing an invalid buffer pointer when we received a DHCP NAK response. We were freeing the wrong buffer pointer. (Case 37023)
This update is dependent on Common Updates Version 1.0

Version 1.43

Ported fix for limited UDP broadcasts. We were unable to transmit UDP broadcasts to 255.255.255.255.
This update is dependent on Common Updates Version 1.0

Version 1.44

While plugging/unplugging the Ethernet cable quickly, IAM could get locked up. Here, we needed stronger handling of disconnecting -- to release whatever had been acquired or to end the process of acquiring. (Case 37693)

Version 1.45

Added APIs to allow users to limit the amount of heap the TCP/IP stack is allowed to allocate. Set the default limit to be half the heap. (Case 40091)
Fixed lockup condition in Ethernet bypass transmit. (Case 37952)

Version 1.46

Fixed a crash when passing a long hex string to getaddrinfo() or zbsGetAddr() (Case 41637)
Allow DNS servers that are received in DHCPv6. Deprecated DNSAddServer (Case 41347, 41348, 41349)
Use DNS server information for static configuration. (Case 39300)

Version 1.47

Report correct Ethernet link speed in SNMP ifSpeed (Case 41612) – **Requires SNMP fix v1.94.**

Version 1.48

Fixed a apparent memory loss problem, caused by incorrect accounting in treck_current_memory_usage, which was losing 4 bytes on every actual free(). Moved it to tfKernelMalloc and tfKernelFree to be both on the same and account for every actual malloc() and free() by Treck. (NETOS-39)

Telnet Updates

Last updated 09/30/10

Version 1.0

Corrected unstable behavior in telnet CLI related to continuous login and logouts. (case 35687, 34480, and 34521)

Threadx Updates

Last updated 09/10/10

Version 1.0

Added an API that allows users to install callback functions which are invoked by the kernel whenever it enters an idle state, or switches to a new thread.

Version 1.1

Corrected wrong usage of sprintf that was causing the cli threads display to end early if thread names lengths were greater than 23 characters. (case 33821)

Version 1.2

Changed C library Mutexes to inherit thread priority. (Case 33485)

USB Host Updates

Last updated 08/26/09

Version 1.0

Added the ability to send data on an interrupt endpoint from a USB host to a USB device.

Utility Updates

Last updated 05/11/10

Version 1.0

NET+OS Programmer

The default password has changed - support the new password, as well as the former for backwards compatibility

A failure was reported when configuring a static IP address. (case 29906)

Version 1.1

SMIDUMP.EXE

Fix for MIB compiler generating code with the wrong index. (case 27535)

Version 1.2

Updated NetosProg_ReadMe.html.

Version 1.3

Replaced corrupted netosprog.exe file. Could cause a crash when run.

Version 1.4

Moved SMIDUMP.EXE to the SNMP Package

WebServer and Email Updates

Last updated 11/29/10

Version 1.0

Email

Added authentication failure string, and made MCGetSmtpAuthentication call for supported authentication.

Advanced Web Server

Added code to ensure that files are closed successfully before clearing the file handle. (case 30048)

Version 1.1

Advanced Web Server

Fixed error in process that would cause a POST request to be rejected if Content-Length was 0. (case 31003)

Version 1.2

Corrected return values when calling ipaddrcmp and sysaccess_handle.

Made RpHandleUserExit call in RpFindUrl instead of calling RpExiternalCgi and RpHandleCgiResponse to make sure the connection state gets updated. (case 32258)

Version 1.3

Changed NAFS_ZERO_OFFSET_ERASE to NAFS_RANDOM_WRITE to correct a problem when using NAND flash. (case 32376)

Version 1.4

Set and save DHCP to TRUE when DHCP is selected and set and save DHCP to FALSE when static is selected. (case 32436)

Version 1.5

Fixed the problem where SMTP authentication fails on CRAM MD5 if the password contains the number 6 or \. (case 33111)

Version 1.6

Cross-site scripting is a methodology where a hacker "piggy-backs" malicious javascript or other language code in a web request or web page and induces a web server to run that code. This

vulnerability has been remedied. Now return bad-request if item is not found on the form. (case 33871)

Version 1.7

Added fOtherMimeType in rpCgiPtr structure for customized content-type. (case 33984)

Version 1.8

Updated web server to return must-revalidate on cache-control header for Content-Disposition: attachment.(case 34406)

Removed itemValue for invalid form entries to prevent cross-site scripting. (case 34389)

Version 1.9

Resolved an issue that prevented the AWS from sending a page when a socket with an FD of 0 was used. (Case 35169)

Version 2.0

Fixed RpWriteIpAddressInDotForm() to use IPv6 for IPv6 family (IPv4 was always used) Also corrected size when copying local address in StcpActiveConnectionStatus(). (Case 35889)

Version 2.1

Skip multipart/form-data item for empty data content. (Case 36298)

Version 2.2

Updated naCgiCheckAccess to execute callback from naSysAccessSetAuthHanlder. (Case 36553)

Version 2.3

This release contains security updates that remediate CVE-2014-9222 and CVE-2014-9223. (NETOS-205)