uEZ v2.14.000 – 03/06/2023

HIGHLIGHTS

- Updated projects to support Rowley Crossworks v4.10.8.
- Updated projects to support IAR EWARM v9.32.
- Updated lwIP integration with FreeRTOS per-thread semaphore and more complete core locking and checking functions, allowing for large performance and reliability boost with full duplex mode and multiple thread applications.
  - lwIP updated to v2.2 and core locking application usage now required.
- Included an Amazon Web Services (AWS) MQTT Shadow topic/subscribe demo using PKCS11 certificate handling with TLS 1.2 support
  - Added a demo GUI to show information sent/received to/from AWS.
- Added FreeRTOS_Plus libraries for networking and command line including several Amazon AWS libraries. (see details section)
- Added MbedTLS v3.4.1 into uEZ 2.14 https://github.com/Mbed-TLS/mbedtls
- Updated LPC4357 Driver with
  - Support for 8,128-byte jumbo frames on Ethernet enabling a large performance boost.
  - SD card support for re-initialization to allow for hot plugging or error recovery. Full power cycling is supported in case of card lock-up error. Automatic switching to 25MHz supported for problematic SD cards.
- Added the ability to play a startup video on LPC4357 with 536x328 resolution at 15FPS.
- Improved the video player reliability for cases where the SD card is formatted incorrectly, is physically removed, is not fully compatible.
- Updated FreeRTOS to v10.6.1 which enabled more features for timers, static allocation, per-thread semaphore, etc.
- Updated FATFS to v1.15 Patch 3
- Replaced CyaSSL with wolfSSL v5.6.3 https://github.com/wolfSSL/wolfssl
- Added wolfMQTT v1.16 https://github.com/wolfSSL/wolfMQTT

FOR ALL EXISTING Projects:

- Generate a new project maker project and add your existing application files to the newly generated project. Make sure that the projects Config_Build.h file don't include COMPILER_TYPE defines. Switching to Heap 4 is also recommended.

DETAILS Specific for Networking and Network Demos

- Added the following FreeRTOS+ MIT Licensed software libraries:
  - CoreMQTT Release v2.1.1 https://github.com/FreeRTOS/coreMQTT
  - CoreMQTT-Agent Release v1.2.0 https://github.com/FreeRTOS/coreMQTT-Agent
  - CorePKCS11 Release v3.5.0 https://github.com/FreeRTOS/corePKCS11
    - Compatible with mbedTLS 3.4.1
  - CoreJSON Release v3.2.0 https://github.com/FreeRTOS/coreJSON

- BackoffAlgorithm v1.3.0 https://github.com/FreeRTOS/backoffAlgorithm
- Cellular-Interface Release v1.3.0 https://github.com/FreeRTOS/FreeRTOS-Cellular-Interface
- CoreHTTP Release v3.0.0 https://github.com/FreeRTOS/coreHTTP
- CoreSNTP Release v1.2.0 https://github.com/FreeRTOS/coreSNTP
- FreeRTOS+TCP Release v4.0.0 https://github.com/FreeRTOS/FreeRTOS-Plus-TCP
  - Note that as of today FreeRTOS+TCP is not integrated into the uEZ network stack/API. The AWS software demo will run MQTT and MbedTLS on top of lwIP instead.
  - Note that lwIP with tuned packet pools (this exists today as the default in uEZ, partially by customer request) may end up with better performance than FreeRTOS+TCP once integration is completed. FreeRTOS+TCP currently only supports 2 buffer mechanisms: either HEAP4/5 or (large) fixed sized buffers.
- FreeRTOS Release pack v202212.01 included:
  - network_transport (compatible with mbedTLS 3.4.1)
  - Cellular Modules
  - Utilities (logging setup for RTT)
  - Reliance Edge
  - FreeRTOS+CLI
- Added the latest AWS release packages:
  - Device Defender Release v1.3.0 https://github.com/aws/Device-Defender-for-AWS-IoT-embedded-sdk
  - Device Shadow Release v1.3.0 https://github.com/aws/Device-Shadow-for-AWS-IoT-embedded-sdk
  - Fleet Provisioning Release v1.1.0 https://github.com/aws/Fleet-Provisioning-for-AWS-IoT-embedded-sdk
  - Jobs Release v1.4.0 https://github.com/aws/Jobs-for-AWS-IoT-embedded-sdk
  - Ota Release v3.4.0 https://github.com/aws/ota-for-aws-iot-embedded-sdk
  - SigV4 Release v1.2.0 https://github.com/aws/SigV4-for-AWS-IoT-embedded-sdk

- FDI adapted the Amazon provided example transport_mbedtls_pkcs11.c file into tls_transport_mbedtls_pkcs11_FreeRTOS.c. This file handles the TLS/PKCS11 certificate setup and integration, but like the provided example doesn't include the platform specific socket connections unless using FreeRTOS+TCP stack (with full platform integration). FDI provides the UEZNetwork integration for connect, send, and receive in mbedtls_bio_tcp_sockets_wrapper_uEZ.c, following the mbedtls_bio_tcp_sockets_wrapper.h example recommendation. Future FreeRTOS+TCP integration or TLS direct to a Wi-Fi module could be handled in that sockets wrapper or new appropriately named sockets wrapper with build time or run-time switching between interfaces and stacks.

- Note: The older version of MbedTLS is compatible with the released version of the AWS components alongside lwIP. The latest MbedTLS requires updating transport and PKCS11 components. The accepted pull request for those components will be part of the next AWS software releases. Other higher layer AWS software, such as the Device Shadow and CoreMQTT could theoretically be updated independently without conflict.
- Setup MbedTLS to use the newly included Simplerandom library. Hardware sensor data is mixed into the generation routines. Once initialization is performed using CRYPTO_InitHardware() the GetRandomNumber() routine can be used in various applications. See uEZRandom.c and uez_mbedtls.c.
- Implemented the jumbo frame settings, per-MCU, in the HTTP server demo and lwipopts.h. The NXP MCU EMAC DMA scheme limits the max frame sizes.
- Updated the lwIP TCP window size and send queue lengths to support the AWS TLS MQTT demo.
- Added several missing MBOX defines in lwipopts.h that were preventing initialization from completing successfully.
- Partially fixed lwIP SACK as the flag was getting cleared before it could be used. Note that there may be other areas where enabling the setting (later in initialization of a new connection) will be needed for some applications.
- Updated the lwIP DHCP task to run forever. This allows the code to restart when the Ethernet cable is unplugged and plugged back in resulting in a new IP address and continued sending of data.
- Note: While not yet part of any demo, uEZNetwork API or GUI, the DHCP client could be run time disabled by pausing the task then calling dhcp_stop(), allowing an application to set a static IP address. By default, the DHCP client is now enabled in the demo projects.
- Added lwIP RTT diagnostics as a faster alternative to using the UART.
- Added SNMP to lwIP options/system port and started enablement of IPV6:
  - Note: Fully enabling IPV6 will require an MCU specific hash table driver update to support multicast and there will be a performance degradation of all networking software with IPV6 support enabled, even when using only IPV4 addresses.
- uEZDemos/Source/App/uEZDemoCommon/NetworkDemos.c was added to launch lwIP related demos and will be kept separate from the Network Startup C files.
  - SNMP V3 was tested with snmpwalk, See NetworkDemos.c for details.
  - SNTP client example was included and is run also with the AWS demo to get network time from an automatic DHCP server or named server.
  - lwiperf was successfully tested using iperf from a PC.
  - tftp server was tested and allows for easily sending a file to the uEZGUI, which will be written to the SD card. This mechanism could be used to speed up software development by allowing for quick updates to the files on an SD card without having to remove the card from the uEZGUI.

- Improvements were made to the to the lwIP configuration file to enable more features by default, separate debug vs release build settings, etc.
- Updated the lwIP packet pools configuration to fix SNMP build issues, and increased the pools to allow for jumbo frames and larger frames that can be fragmented when jumbo frames are disabled.
- Updated the lwIP packet pools configuration to include smaller size pools, achieving a noticeable performance improvement in the HTTP server demo for smaller packets such as ACK packets. Larger pools can now be used automatically if all smaller pools are being used.
- Updated some uEZ demos and APIs to use the now-required core locking functions in lwIP. These were also added to the lwIP examples.
- Updated the Ethernet PHY initialization routine to allow passing in a Boolean parameter to allow for power on initialization or later re-initialization in case the PHY clock stops.
    - Note that once a full driver is developed, a newly supported Wi-Fi or cellular module could be initialized as a second network interface using a similar initialization function routine, with lwIP supporting some level of automatic handover between multiple interfaces.
- Updated LWIP to use proper intrinsic CMSIS byteswap functions in both Crossworks projects and IAR projects
- The BasicWEB demo has a new HTML printout format and uses the updated FreeRTOS statistics functions to pull information. Now, the priority numbers, stack remaining, runtime stats, and heap free can be viewed for up to 15 tasks.
- Updated the HTTP server demo to read larger blocks from the SD card matching the size of the SD card blocks or larger, using better memory alignment to help avoid unaligned access requiring additional copies for the DMA. The demo can now adjust to the jumbo frame size automatically enabling a large performance boost on LPC4357 applications.
- Both the SD card HTTP server demo and the BasicWeb demo can be included in a demo project and run simultaneously on separate RTOS tasks and separate TCP ports. The webpage showing the current running tasks can be periodically accessed from a web browser, while a large file is separately downloaded from the SD card. Using the updated lwIP RTOS integration, the file will download with the correct checksum.
- Added an mflash example to allow running the AWS demo. Note that the mflash implementation attempts to program the flash in the wrong order (first block last), and this may currently fail with certain size pieces of data as the routine hasn't been fully debugged yet.
- Added the ability to detect if a misbehaving LAN8720 PHY stops outputting a clock. The PHY can now be software reset, even if no reset output GPIO is available. In most cases, it is now possible to re-initialize the Ethernet without needing to reboot the entire system. But doing this reliably will require good application-level support in case the network stack or application protocol stack

gets stuck somewhere. (These may need to be re-initialized under certain conditions.)
- Fixed ping checksum in lwIP along with all options that specifically depended on LWIP_IPV4 define being present.

DETAILS

- SEGGER RTT and SystemView updated to v3.52.
- Added a workaround for __GNUC__ being the wrong version in Crossworks causing the wrong intrinsic functions to be included in CMSIS and MbedTLS.
- Added FPU_PRESENT define to Cortex-M4 projects as there is a double check in the software for generating some FPU code in CMSIS.
- Added a HEAP4 output section option in the drop-down menu for all Rowley Crossworks library projects, demo projects, and project maker projects. The HEAP4 output section has not been added to the IAR projects, at this time.
  - Note: HEAP4 or HEAP5 is required to run some Amazon software.
- uEZ Project Maker Rowley Crossworks projects now include a link to the library project, and the M4 project can now build the M0 project automatically. Note that the first build of the M0 project will trigger a library build which may take some time. That mechanism starts only from command line build (setup using the prebuild command) with matching library configuration names. As soon as the library solution is removed from the M0 project, it will stop building it automatically.
- Fixed the SEGGER Systemview port stack growth setting since the required header can't be included into the Systemview file. This fixes a stack display issue.
- Projects no longer need or should have COMPILER_TYPE define in Config_Build.h. This will be figured out automatically.
- Fixed COMPILER_TYPE issue with IAR assembler.
- The SD card HTTP server demo now allows downloading to external flash (NOR or QSPI) as a file, showing an improved download speed compared to reading files from an SD card. The LPC4357, using Jumbo Frames, can dump the 16MB QSPI straight to a file on a developer's PC in 8 seconds, no debugger required.
- Updated FreeRTOS Config to enable more features such as per-thread semaphores and timers. Various missing and/or newer define settings are now included to allow running more FreeRTOS+ software and demos.
- Removed FPU support from Cortex-M3 and Cortex-M0 builds in FreeRTOS config which will save memory space.
- Added resistive touch screen platform defines.
- Cleaned up the microcontroller specific configPRIO_BITS definitions to get them from common header files where possible (everywhere except IAR assembler) and move the additional hardcoded defines out of FreeRTOSConfig.h into the common header uEZProcessor.h.

- Added FATFS buffer file table example into video player and related audio player.
- Hardcoded the FATFS default formatting size to au_size 4096, matching PC formatting and improving video player reliability.
- Updated UEZTickCounterGetDelta to handle tick rollover.
- Removed various includes, such as lwipopts.h and uEZMemory.h, in platform files and certain low-level headers to avoid circular include.
- Various threads now include a global thread handler so that the thread can be controlled from application software. These will default to NULL before init.
- Updated the GUIDemo project's emWin memory. The full demo running on 800x480 LCD used 4.96MB of the emWin memory area.
- Cleaned up various include paths and no-longer supported tools/features:
  - Removed KEIL MDK related defines.
  - Removed CODE_RED related defines
    - Note: All future MCUXPRESSO projects will use standard GCC macros/defines/checks matching the Rowley Crossworks projects where possible.
  - Added missing IAP files to library projects.
  - Cleaned up UtilityFuncs to allow easier header including in RTOS files.
  - Fixed many IAR specific warnings.
  - Removed old library build scripts.
  - Replaced many #if defines with #if (DEFINE == VALUE) checks.
- Added missing R4 define in CARRIER uEZPlatform fixing most issues with its project maker project.
- Added platform defines for GPIO_PSU_3VERR GPIO. Removed the consoled command from uEZGUIs that don't have GPIO.
- Removed FreeRTOS Debug-Trace selections from Project Maker and Demo projects as they were not included, supported, or tested. This feature requires an annually renewed, per-user, Percepio license subscription and possibly patches or specific configuration changes in FreeRTOS. The folder structure was kept in place to allow drop-in of the correct Tracealyzer version matching FreeRTOS V10.6.
- Removed Release build selection from IAR Demo projects since they normally aren't tested and require extra time to keep per-file options correct.
  - Note: IAR library projects and project maker projects still have Release build selections that are maintained.
- Addd UEZGetRunTimeStatsCounter and UEZConfigureTimerForRunTimeStats to uEZRTOS.h to allow usage of the FreeRTOS Run Time Statistics feature. An unused timer peripheral will need to be selected in the platform file to use the feature.
  - This was tested on LPC4357 M4 with timer3 setup to generate the 20khz timer callback. The timer needs to be initialized before the RTOS is started.

- Added InterruptSetPriority() to Interrupt.h API. Peripheral interrupt priorities can now be changed using the same definition levels/names as in the LPC drivers.
- The LCD backlight power-on delay on select uEZGUIs no longer uses the Timer 0 peripheral by default, instead the RTOS timer is used. This frees up Timer0 for application use. The backlight delay no longer blocks.
- uEZBSP delay functions were moved to the LPC specific UtilityFuncs.c files. This partially cleans up the platform files of what was duplicated code and allows IAR library projects to use a defined per-file optimization level on the file inside the library, so timing is consistent between optimized and non-optimized builds. These can be overridden in an application project if the MCU frequency is changed.
- Added SEGGER_RTT_printf_0 (along with 1 and 2) to allow a direct printf function replacement. Logging software can easily include the RTT version with a fixed RTT channel parameter.
- Created “SEGGER_ENABLE_EMWINSPY_RTT” to allow setting up an unused RTT channel for SEGGER emWinSPY.
- Added UEZPlatform_5V_Monitor_Get_Raw_Reading() to uEZPlatformAPI.h allowing calls from library functions and demos.
- Removed empty or unused process stacks in Crossworks projects. The empty or unused process stacks should no longer show up in the output graph of any project build. By defining USE_PROCESS_STACK and setting the desired number in a project, the feature could be used with a FreeRTOS MPU port. Currently the MPU ports aren’t included in uEZ but could be setup on the Cortex-M4 LPCs.
- Updated the video player in uEZDemos to always center the video if no information text is displayed below and make sure the arrows in the video selector screen don’t go off the screen on certain LCD resolutions.
- In uEZDemos added emWin window manager pieces to allow running various demo screens such as the AWS demo and allowing for restarting emWin when switching between SEGGER demos.
- Improved emWin performance when calling GUI_X_GetTaskId() by avoiding a search for a uEZ Task handle. The FreeRTOS task handle is returned which is used for task protection if multiple tasks are calling emWin.
- Updated the Cortex-M4 project maker and LPC4357 demo projects to allow for easier text and data subsection placement overrides in Crossworks projects. This allows for easier placement of large demo projects into QSPI memory. A fully enabled demo project with emWin demos, TLS networking, and USB fit into the split internal flash banks and QSPI even if the total size is greater than 2MB. No per-file override configuration or library project modifications are needed.
- Removed the SEGGER AppWizard installer from the uEZ package as it is not suitable for the micro-controllers or color formats used on uEZGUIs. The previous uEZ releases still contain the latest NXP released version of the tool. It is recommended to use SEGGER’s GUIBuilder.exe instead.

LPC43XX Specific

- LPC4357 uEZ Project Maker and GUI demo projects now start FlashB usage at 64KB offset to accommodate the AWS demo certificate/key storage. This offset can be removed by changing ”FLASHB_START=0x1B010000” to ”FLASHB_START=0x1B000300”
- Changed LPC4357 I2C interrupts to high priority.
- Updated Flash_NXP_LPC43xx_GetBlockInfo() to return the correct block size for the first 64KB area.
- The SD card HAL was updated to include the “IsCardInserted” routine to use when a card detect GPIO is present on an LPC4357 uEGZUIs.
- Added CMD6 to the LPC4357 SD card driver and improved specification timing including timeout detection and post-initialization clocking.
- Updated uEZGUI-4357 platform files to remove the SD card power sequencing routine as it is now included in the LPC4357 drivers.
- LPC4357 SDCard drivers now include full re-initialization support to allow for hot plug including full power off/on sequencing and complete peripheral reset which can clear out all known error conditions.
- On LPC4357 uEZGUIs the SD card error detection callback in combination with card detect pin can be used to detect errors if a card fails to initialize at high speed mode BUT remains connected so the driver can attempt initialization at a slower frequency. Tested or recommended SD V2.00+ cards:
  - SanDisk industrial cards: wear leveling, industrial temperature range, current specification rating.
  - Samsung Evo U1 cards passed testing but may lack wear-leveling/temperature range support.
  - Kingston non-industrial U1 rated SDHC micro-SD cards have compatibility issues
  - Kingston industrial cards not yet tested
- uEZGUI-4357-50WVN projects now include a Config_Build.h switch that needs to be changed to support Rev 1.X PCBs. In addition, the memory size needs to be lowered to 16MB in the placement configuration of the project. Going forward all projects will be targeted to the Revision 2 boards that have the current LCD and 32MB SDRAM.

uEZ v2.13.100 – 09/15/2023

HIGHLIGHTS

- Updated projects to support Rowley Crossworks v4.10.4.
- Video player performance improvements. Frame skip to timer now working.
- PC video converter application updated to allow higher resolution video, such as full screen 800x480 on LPC4357 units.

- Release build now disables more asserts for measurable performance improvements.
- LPC4357 Driver Updates
  - SDRAM drivers updated with improved latency timings to better match the loaded SDRAMs
  - All projects updated to use 32MB SDRAM which is loaded on the latest uEZGUIs.
  - SD card now runs at 51/48 MHz and includes performance optimizations.
  - Video player can now run 800x480 video full-screen at 8 FPS
  - SD card slideshow speed massively improved.
  - M0 library build output added to Crossworks library project. Crossworks Project Maker now generates an M0 project.
  - All major drivers now build on LPC4357_M0 library project.

FOR ALL EXISTING Crossworks Projects:

- Generate a new project maker project and add your existing application files to the newly generated project.

DETAILS

- On uEZGUI-4357-50WVN Projects now include a Config_Build.h switch that needs to be changed to support Rev 1.X PCBs. In addition, the memory size needs to be lowered to 16MB in the placement configuration of the project.
- Added missing CRP regions to IAR LPC4088/LPC1788 project icf files.
- Crossworks project maker projects now have correct Release build settings. Previously a duplicate setting caused “NDEBUG” to not be applied, at least in some cases.
- QSPI drivers added for possible alternative ISSI QSPI parts. (faster, large size parts)
  - Note: These QSPI updates can apply to both LPC4088 and LPC4357.
  - Note: The new ISSI drivers remain untested and not included in the driver by default. Including additional drivers will increase the size of any bootloader.
  - LPC4357 can memory map read up to 128MB sized QSPI.
- In HTTP Server demo add missing error handling
- In Project Maker test project storage test add ability to see when an SD card or USB Flash drive is initialized.
- Increase emWin demo RAM for 800x480 uEZGUIs.
- Fix the stdinout redirect code in platform.c files since the calling order matters.
- Cleanup and better document all uEZPlatform.h audio volume definitions. Some uEZGIUs had volume that was too low or high that are now fixed.
- In lwipopts add the ability to change the priority per MCU. This will prevent the more serious crashes on 4357 where the LCD goes out.

- Moved config trace facility to the SEGGER_RTT_SYSVIEW_Config.h so that both library and app project can get the definition correctly. This will allow the TASK command to show up again.
- Cleanup LWIP assert to use less characters and put in a define to attempt to use only the filename and not the full path name. (The end result is still toolchain dependent.)
- Fix sys_thread_new so that if we call to create a thread with 0 stack size or prio it will at least give us at least double the minimum stack size and at a high priority.
- In HTTP server fix bug where it wasn't flushing.
- In HTTP server add code to allow changing parameters per MCU so we can have higher stack for CM4s vs CM3s and have different priority for LPC4357.
- In HTTP server increase the input buffer size since we need a larger amount to handle the buzzer setting input message from modern PCs. We can now use the buzzer feature again just fine from newer web browsers.
- Fixed the simpleUI SUIShowPage0Fancy to not go negative on framebuffer. This was causing crashes on all LPC4357 projects using SWIM software and could also cause issues on LPC1788/LPC4088 projects trying to access non-existent memory.
- The PC application VideoTGAtoBIN converter now supports multiple color formats selectable at run-time using a new command line switch. The BeginConversion.bat script example was updated to use it for uEZGUIs with the correct color format.
- S25FL256L_64K QSPI driver added as possible load option for LPC4357 to improve QSPI performance.
- LPC4357 example program script updated to reset type 2 to insure more reliable programming.
- LPC4357 Drivers:
  - SDRAM CAS 2 support added
  - Cleanup of several LPC4357 drivers
  - On LPC4357 move the IRTC require to always be after main task start which improves bootup time slightly. It was removed from the standard platform require function call.
  - Removed the un-needed RTC errata fix since all LPC4357 uEZGUIs were built using production revision chips.
  - Lowered RTC delay interval to 1100 ticks during the RTC set routine.
  - CREG0 32K output (to other peripherals) was disabled to avoid tripping the alarm feature at bootup that we aren't using. This alarm was occasionally triggering output pins from the LPC4357 due to be enabled but not initialized. Note: some output pins could only be measured on older uEZGUI hardware.
  - Note that 1Hz output is still enabled to RTC and the 32K oscillator is still enabled.

  - Add several new macros for the high frequency pin mode for use with specific GPIOs. These are now used during various peripheral initialization.
  - Added new test code to SD card driver to be able to count errors.
  - Fixed SD card driver clock divider logic and fix 400KHz startup routine.
  - Fixed the SD card pins to disable the filter on clock line for >30MHz. From the datasheet they recommend setting EHS slow on CMD/DAT pins, so keep it slow but set clock pin to fast.
  - Added missing SDDELAY register setting that is required per datasheet. The chosen values were tested across many units. In combination with GPIO changes, this allows the SD card to run at 51MHz with no known issues.
  - LCD drivers updated to run at 60Hz refresh based on the actual pixel clock timing.
  - Add the ability to switch to 102/120/160/204 source clocks for LCD. (only tested 102 and 120 so far)
  - Fixed LCD pins to use the high frequency macro and in some cases use the high current macro and label those high current pins. (different type of registers for different pins)
  - Added LCD power-off ability to LPC4357 platform files.
  - Updated USB Driver to prevent trying to use unsupported USB 2.0 rates.
  - Removed printf debug code from LPC4357 USB driver that could slow/crash it.
  - Updated USB power on delay for off-board USB port in platform files.
  - Removed LPC1788 power control register settings from various LPC4357 drivers.
- On LPC4357 Project Maker the M0 support must be enabled in Config_Build.h.
  - After changing the Config_Build.h, first open the separate M0 project ("_M0.hzp") and build the matching build (debug or release).
  - Build the M4 project.
  - In the M4 project connect the debugger in JTAG mode, then click the "run" button. It will automatically program both projects.
  - Note: The M0 project cannot program any LPC4357 internal code flash from a SEGGER J-Link.
  - Start the execution of the M4 project. Once the main task has been initialized, the M0 will be started.
  - The M0 project can now attach the debugger (only in JTAG mode) and use up to 2 breakpoints at a time.
  - Note: The M0 will perform no PLL, SDRAM, or QSPI initialization. It will be started at 204MHz with 1000 Hz RITIMER RTOS tick rate. The M0 can read all memory at any time. The M0 will run uEZBSP init and be able to run uEZ API functions. The example project maker includes the Heartbeat LED example using uEZGUI function calls. Optionally the uEZBSP can be disabled in the platfom file to run a simple while loop.
  - Untested interrupts are included in the uEZGUI platform file for core-to-core communication. See LPC34357 User's manual for details.

- Note: When the M0 project has been rebuilt, an M4 project, script, or J-Flash must be used to program it after stopping both cores.

ERATTA

- LPC4357 RTC driver will hang the core for over 4 seconds during cold-power-on when battery backup has been drained. Possibly it can happen when battery isn't drained also. This can lead to possibly 8-9 second boot-up times.
- The emWin demo in the uEZGUI demo project still has differing background colors between 480x272 LCD units and 800x480 LCD units.
- See 2.13.000 Errata section for previous errata.

uEZ v2.13.000 – 06/30/2023

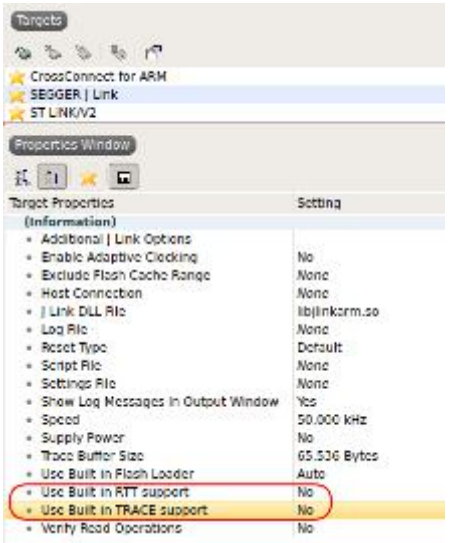
HIGHLIGHTS

- emWin updated to v6.32bc
- SEGGER RTT and SystemView updated to SystemView version 3.50.a.
- Project Maker projects now include a uEZ Resource example.
- Updated Project Maker Crossworks projects to be automatically boot loadable with the latest uEZ Bootloader release.
  - Bootloader build target included with automatic starting address switch and memory fill where needed.
- Updated projects to support IAR EWARM v9.32.
- Updated projects to support Rowley Crossworks v4.10.3.
- IAR library projects have a new optimized build-target for size reduced bootloader projects (high optimization enabled for select files).
- Added a generic 4.3" 480x272 LCD driver for uEZGUI-4088-43WQN that supports all LCDs used over the lifetime of the product.
- Better CRP support and automatic project address switchover in Rowley Crossworks projects. See "WARNING: Code Read Protection (CRP) in LPC Devices for details.
- Major code improvements to LPC4357 projects.

FOR EXISTING PROJECTS:

- When updating Crossworks projects, make sure the following is performed:
  - Update all developer computers to Crossworks v4.10.3. Reason for this is covered in the errata of previous releases.
  - Add the file uEZ/Source/Library/StreamIO/StdInOut/CrossWorks/StdInOut_CrossWorks.c to all application projects. The file can no longer be used in the library projects.
  - In all Crossworks application projects add the following settings to "Common"
    - Debug I/O Implementation: None
    - Default debugIO implementation: None

- Remove StdInOut_CrossWorks.c from all Rowley Crossworks library projects and instead add it to all application/demo projects.
- Update all Crossworks projects to use the new threads_CM3.js or threads_CM4F.js (scripts). This fixes a bug when viewing running tasks in the Crossworks IDE.
- In Rowley Crossworks make sure the following settings are set to "No" in the SEGGER J-Link settings menu or the MCU will not function correctly after a debug session has been started due to pin conflicts.



DETAILS

- Changed how LPC4357 uEZGUI projects blink the heartbeat LED. (This used to be with uEZ function calls, now it is with bare LPC commands.)
- Improved the memory placement setup in LPC4357 Rowley Crossworks projects. More variables have been added to allow changing the start address for boot loadable project builds. Standard builds start at address 0x1A000000, while bootloader started projects start at address 0xxxxxxx. This automatically includes or excludes the CRP region as needed. Bank B fill is included to ensure that data starts at address 0x1B000000. This prevents issues with the hex-to-bin conversions that have the potential to place code into the CRP region thus inadvertently blocking full access to the MCU for debugging or programming.
- Added missing autoblock settings to USB clock setup on LPC4357 drivers.
- Updated LPC4357 bootup routine to no longer enable clock output (for testing).
- Improvements and fixes were made to the LPC4357 timer driver. It can now accept normal timeout ranges used throughout the uEZ software. (Large values such as 1ms will now work correctly.)
  - Fixed crashing due to passing in a NULL.
- Updated LPC4357 SD card pin setup routine for powering off SD card due to one of the pins not being a normal GPIO pin.
- Updated LPC4357 touchscreen IRQ pin setup.

- Updated the LPC4357 IAP driver. Block select handling and bad range checking updated. Added more error handling and fixed semaphore crashes.
- Updated LPC4357 Interrupt names such as USART2, USART3
- Updated LPC4357 UART2 handler for GPIO5
- Added alternate PWR COM console selection ability to LPC4357 platform files.
- Added non-console UART require for the alternate PWR COM on LPC4357 platform files.
- Improved power control for LPC4357 QSPI driver enable.
- Updated the LPC4088 projects to use the newer QSPI driver used by the LPC4357.
- Added CRC calculation software to the LPC4357 library projects. These use a software CRC even though the part has built-in CRC checking.
- In the LPC4357 projects, disabled the unused OE pin going to the SDRAM.
- For existing LPC4357 projects, update the SDRAM refresh cycle count to 4096. This will better match the SDRAM used in these uEZGUI's. In addition, adjust timing so that the (untested) IS42S32800D-7TL could be supported as a load option. Cleanups were made to the SDRAM driver and as a result some timings are now improved.
- Updated the LPC4357 DAC driver to avoid sending the DAC output to the wrong pins. This part can output the same DAC signals to multiple pins at the same time including the onboard amplifier, expansion connector, and the LCD pins. Added separate require routine for offboard DAC. Note that the DAC pin going to the audio AMP can't be disabled so to use offboard audio the onboard amplifier should be muted on that channel. The DAC output is available at the ALT PWR COM connector for easy use with a video player or wav player.
- Updated LPC4357 bootup routine to not enable clock testing output.
- Updated to the LPC4357 timer driver to accept normal timeout ranges used throughout uEZ software. ("Large" values, such as 1ms, will now work correctly.)
  - Fix NULL handle checking crash
- For existing LPC1788/LPC4088 projects, update the SDRAM refresh cycle count to 4096. This will better match the SDRAM used in the uEZGUI's.
- Updated uEZBSPDelay() timing in Crossworks builds using manual optimization override. For IAR the corresponding platform.c file will need to be built (in application project) with optimization enabled for correct delay timing. A warning will appear when the function is not optimized.
- In each Crossworks library, changed the CMSIS/system_LPC43xx.c, LPC4300_Startup.s, thumb_crt0.s to be excluded, as we will now include those only in application projects. The same exclusion applies to LPC4088 and LPC1788 also. (This will avoid duplicate definition errors.)
- Cleaned up StdoutGet() and StdinRedirect() related code in the library and platform files to avoid having duplicate global variables.
- Updated the SD card bootloader demos for all production uEZGUIs.

- Updated the Project Maker projects to avoid activating button presses on 2 screens when a single screen press occurs. These changes were applied to the back/reset button:
  - Enable BUTTON_REACT_ON_LEVEL emWin setting on project maker projects. This is a new default recommended by SEGGER.
  - Add timer enable button ignore example code to prevent button from being activated during the first 2 seconds of switching to a GUI window.
- Updated the library build to add new #defines that are used when the library is to be used by the bootloader. For example, file system FORMAT command was removed. This example could be used to remove other software from an application project.
- Updated the library build to print out a note corresponding to which FreeRTOS heap is used. #defines are now included allowing the heap to be easily changed. By creating new “Debug HEAP4” and “Release HEAP4” builds for example, the heap can be switched to heap 4 without having to add or remove files from the library project. Current projects, however, are missing some of the additions needed for other heaps. This may come in a future project maker release, or an example could be provided on request.
- Updated the library build to print which heap is selected and if any FATFS features are disabled.
- Enabled warnings in IAR library projects.
  - Removed unused suppressions: Pa082,Pa039,Pe061
  - Added new suppressions: Pe193,Pe826,Pe068
- Updated GAPI library color defines and color size includes.
- Updated the library projects to add resource cache so you no longer need to manually add the cache to application projects.
- Added low-level PWM driver for select uEZGUIs for code size reduction of application projects. (No longer using uEZPWM/LPC_PWM APIs when that option is enabled.)
- Corrected audio enable require routines for some uEZGUIs to avoid a popping sound at boot-up.
- Updated several demos, such as temperature graph, with the latest uEZ release and emWin versions.
- Updated uEZ Demo TimeDate menu to allow pre-2000 dates.
- Defaulted uEZGUI-4088-43WQN project to use the new generic LCD driver (in Config_Build.h).
- Updated several 480x272 LCD drivers to use the same timings as the generic driver.
- Updated some GPIO loopback tests for factory testing.
- Updated the uEZGUI-4088-43WQN Crossworks uEZ Demo project to correct the internal SRAM size in the xml map file and add missing QSPI peripheral register set view.
- Added reset/exit buttons to the settings menu of the uEZ GUI Demo application. These buttons are used in the bootloader demo to exit or just to reset the unit.

- Added UART output to the uEZ GUI Demo application. The SD card is mounted at boot up, if an SD Card is found and a FATFS file system is discovered, information will be printed out to the UART (SD Card size, File size, number of blocks, etc…). This now forces SD card initialization at bootup.
- Updated the project maker:
  - projects now include a back/reset button.
  - LPC1788 templates now include a NOR Flash region.
  - IAR projects now include correct NDEBUG setting in the Release builds.
- Updated GCC optimization macros for newer toolchains.
- Added defines to allow FreeRTOS HEAP selection without needing to remove/add files from library projects.
- Updated select LCD drivers to add the ability to remove timer API calls for code size savings.
- Cleaned up SimpleUI_Types.h include for T_pixelColor. Corrected various SWIM includes. This fixes linkage order problems.
- Updated some platform files to prevent network stack code from being included in the build when disabled.
- Updated CMSIS files so that weak definitions can be used. Previously this was only usable in specific toolchains.
- Updated SUIHidePage0 being able to write out of bounds in specific cases or thinking that pixel color size is 32 bits instead of 16 bits.
- Updated placement of pack macros for IAR to remove warnings.
- Added missing Rowley Crossworks inline macro (to be able to use INLINE).
- Fix GAPI library build issues related to color defines/color size includes.
- Replaced the FreeRTOS vtask delay (no longer supported) with INCLUDE_xTaskDelayUntil.
- Updated FreeRTOSConfig to remove settings for older versions of FreeRTOS and uEZ.
- Improved QSPI driver error handling.
- Increased the maximum frequencies used by QSPI chips. Note that while frequencies are listed in the NXP provided QSPI library, there is no driver mechanism to change the frequency. We must limit the frequency in the PLL setup routine then avoid using the standard “READ” command for a typical chip. Note that the READ command is not implemented in the driver, and that page program commands can handle a higher frequency in specification for multiple brands of parts.
- Added Infineon QSPI support for relevant chips that might be used as alternate parts in the future. All alternate parts have similar specifications to the Macronix QSPI.
- Removed unused QSPI parts from the library to save code space.
- Added One Time Programming (OTP) support to the Macronix QSPI driver.
  - Various QSPI/Octal SPI parts have 512KB or more usable OTP space which may be useful for permanent serial number, key storage, or other identification information.

- Added missing UEZPlatform_TouchscreenRT_Require() to relevant platforms.
- Cleaned up StdoutGet() and StdinRedirect() related code in library and platform files to avoid duplicate global variables.Recent toolchain updates required updating some library projects to remove duplicate included files. (These duplicates now led to duplicate definition build errors.)
- Updated several library projects to include missing resource cache library software. It no longer needs to be manually added to application projects.
- Updated code that was causing warnings such as function definitions missing (void) or (void * workspace).
- Documented available LCD clock frequencies in the LCD drivers. The drivers require the programmer to specify a frequency larger than each "step" that the driver supports. Note that LPC4357 can switch clock source to the same frequencies as LPC1788, to be able to run LCDs the same way.
    - As an example, we use 9.24 MHz clock selection to achieve 9.23Mhz LCD clock (120MHz main clock with a 13 divider achieves 9.23Mhz). 9MHz is not possible, selecting 9MHz would result in a clock that is too slow.
- Add example to uEZGUI-4088-43WQN to allow disabling HSYNC/VSYNC signals and running pure DE mode. (the signals are set GPIO output low per spec) This can't actually be used on all revisions of the uEZGUI as we believe that earlier LCD models don't support that mode. We may use this mode on a future product or LCD change revision of a newer uEZGUI to maximize driver compatibility. Most of the non-DE supporting (internal to the LCD) chips are no longer produced, so it is a safe bet that future LCDs will support DE mode.
- Added UEZBSP_EXTERNAL_FLASH_BASE_ADDRESS to uEZPlatform.h so application code can reference the start of the external NOR or QSPI flash. Note that the LPC4088 and LPC4357 have QSPI flash at different addresses. By using this define, the same application code can be used on both uEZGUIs.
- Updated the software to check if the RTOS is running. This is required because the current functionality cannot be used until main() is reached and some variables are initialized.
- Updated the FATFS Error handling to translate all possible error returns.
- Corrected UART2 require in the 43WQR platform file.
- In targeted uEZGUIs, a low-level code path was included in the platform files to allow for size reduction. Notable examples are low level backlight PWM in uEZGUI-1788-70WVN and uEZGUI-4357-xxWVN. In addition, more uEZGUI GPIO routines can be excluded in the LPC4357 such as the reset pin usage.
- NOTE: The Project Maker resource example now uses the HIMG software from the uEZ library.
- NOTE: uEZ bootloader software should be placed next to the uEZ folder using the folder name "uEZBL" for script access.

WARNING: Code Read Protection (CRP) in LPC Devices

NXP LPC MCUs can use CRP to protect your Intellectual Property (IP) when it is in the MCUs memory. If you wish to use CRP, please refer to NXP's application notes, User's Manuals, or the community forums.

CRP is controlled by values written into specific memory locations of the MCU. When CRP is not used the CRP memory locations are all 0xFF's (not protected). All FDI projects are written utilizing no CRP.

Memory section .crp1 and when needed .crp2 are each setup with 4 bytes, initialized to 0xFFFFFFFF (no CRP). We do not recommend changing the memory sections in our product demos as they are set up for the available memory of the uEZGUI. As an example, please review the Memory Usage Map and Memory placement section in uEZGUI-4357-70WVN.c.

Important Note – You should take particular care when modifying the value placed in the CRP word, as some CRP settings can disable some or all means of access to your MCU (including debug). Before making use of CRP, you are strongly advised to read NXP's documentation on this functionality. You can access the documentation for your MCU at Microcontrollers and Processors |NXP

See "LPC Code Read Protection Readme.pdf" for important details and examples of memory placement in LPC devices.

ERATTA

- SystemView HEAP viewer is not fully setup or working. It seems that additional pieces would need to be added and projects may need to be switched to a different heap for it to function. Without the heap viewer, the updated SystemView doesn't really offer any new functionality over the previous releases.
- The optimization macros only function in IAR toolset to lower the optimization, not raise the optimization. This means for uEZGUIs such as uEZGUI-4357-70WVN the uEZBSPDelay() timing is not accurate in IAR unless the project's optimization or the uEZGUI-4357-70WVN.c's optimization is set to a high enough level. To optimize just that function it would need to be split into a separate file or converted into an assembly routine. Some software may not function correctly if uEZBSPDelay() takes twice as long to execute as is generally expected.
- The QSPI driver cannot change delay cycles to allow for higher frequency operation.
- The QSPI driver cannot dynamically change frequencies due to being hardcoded in the PLL setup.

- In the latest Rowley Crossworks, you may need to manually run indexing and possibly close/re-open the projects for the IDE to be able to "find" the included header files.
- LPC_EMC->DynamicRefresh might calculate a lower number than it needs to be for the SDRAM. But, using a number that is too high would be bad, so a possibly lower number is acceptable.

uEZ v2.12.00 – 10/06/2022

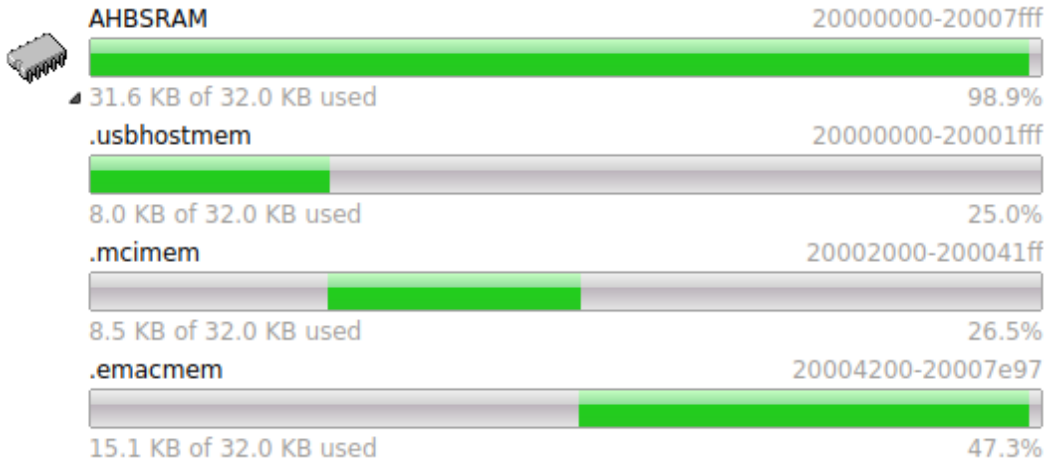
HIGHLIGHTS

- Added support for uEZGUI-4357-70WVN
- FreeRTOS Updated to v10.4
- emWin updated to v6.16c
  - NXP MCU licensed SEGGER AppWizard now included
- lwIP updated to v2.1.3
- SEGGER RTT and SystemView updated to SystemView version 3.30.
- FATFS updated to R0.14b
  - Format command now supported with UEZFileMKFS function.
  - New information query commands for the attached storage device.
  - Software has been restructured (towards this release) to make it easier to drop in the next few FATFS updates.
  - Compatibility improved to better support larger file systems and better ability to support devices formatted from a PC.
- Added uEZGUI_noSDRAM projects as an example for running the demo projects using only internal RAM and on-chip flash memory.
  - These projects run most of the command line commands and peripherals (audio, storage, sensors, etc), but the LCD and Ethernet are not started.
  - The external memory will still be initialized, so for example a memory test could be run across the entire SDRAM.
- Added "Test" project to project maker including:
  - emWin storage test (used to check reliability and also device compatibility)
  - emWin touchscreen test (used to check for phantom touch issue)
  - emWin QSPI slideshow example (only uEZGUIs with QSPI included)
- Updated projects to support IAR EWARM v9.30.

DETAILS

- Updated storage related functions (see project maker for example):
  - UEZFileMKFS(const char aDriveNum)
    - Formats the attached device using default recommended settings for FAT32.
    - Note: Formatting on the uEZGUI may improve storage performance.
  - UEZFileSystemGetStorageInfo (const char * const aDrivePath, T_msSizeInfo *aInfo)

        - Returns the number of sectors, sector size, and block size of the device.
        - Display total size of device.
        - Attempts to auto mount the device first.
    - UEZFileSystemGetVolumeInfo (const char * const aDrivePath, T_uezFileSystemVolumeInfo *aInfo)
        - Returns the bytes per sector, total number of clusters, and number of free clusters of the attached device.
        - If this function returns with no errors then you have a useable device attached, and the file system is now mounted and ready to be used.
    - Using these new functions an application can detect if the attached device is present and has a valid filesystem (ready for writing). If a device is attached but the file system cannot be read, the uEZGUI can now easily format the device to get it back into a useable state.
- Added a new LCD driver for 4.3" NHD43480272EF-ASXP sunlight readable display.
- Added a new LCD driver for 4.3" DLC0430BCP12RF sunlight readable display.
- Added a new LCD driver for 4.3" NHD43480272EF-ASXV.
- Added a new LCD driver for 7" NHD70800480EF.
- In the latest demo and project maker projects, the LPC1788/4088 AHB memory placement configuration has been updated to use a combined AHB. The tested configuration is to place USB memory in the beginning (region 1), MCI memory (for SD card) going across the AHB boundary, then Ethernet memory is placed in the second half of AHB (region 2). For each application the AHB sizes may need to be adjusted (for example allocating more USB Host memory and less MCI memory). When adjusting the memory sizes, it is important to keep the USB and Ethernet memories entirely in a single region as we did not test having them across both regions. But the SD card driver doesn't have any issues.
    - The LPC1788 and LPC4088 projects will now use #defines to ensure the following 3 sections are always included, even if USB or MCI is turned off.
    - This is the tested and recommended memory configuration. Note that each region starts aligned.



- In some cases, the LPC1788/4088 MCI driver was hitting a DMA underrun condition after heavy file writing activity. Updated the driver to use the receiver direction as DMA flow controller and added an automatic retry to handle the cases where we need to wait on the SD card to be ready to process the next read or write command under heavy load.
  - For flow control, when sending data to the SD card, set the MCI as the flow controller. When receiving data, the DMA is the flow controller.
- It is highly recommended to use the UEZFileSystemSync("1:") command after writing a file, as we ship the default configuration of FATFS where the entire contents of the file/header may not be written until the file is closed or file system is unmounted.
- uEZ library projects now have only one Config_Build.h file that is shared for each toolchain project.
- Updated and improved LPC4357 software support to be more in-line compared to LPC4088 on both IAR and Rowley Crossworks uEZGUI projects:
  - USB Host now runs at the correct speed.
  - USB Host Mass Storage driver is stable and passes reliability test on both USB ports.
  - Hot plug detect works for both USB ports.
  - SD card passes reliability test
  - Code Read Protection is properly implemented.
  - Updated examples showing proper usage of GPIO.
- The memory placement was cleaned up in all demo projects to use the ".frames" section for LCD buffer and the ".emWin" section for emWin memory. You can easily turn the LCD off and not use any memory as shown in the noSDRAM projects. In addition, the LCD related memory can be relocated in any project.
- All uEZGUI demo projects are updated to use the full SDRAM range in the placement setup. However, for project maker the user will still need to manually update the following projects to 16MB SDRAM after generation (if needed).
  - uEZGUI-1788-70WVT
  - uEZGUI-1788-70WVM

- The LPC1788/LPC4088 .emacmem size and start address have been corrected and will now show up correctly in the map file or graphical memory view.

LPC17xx_40xx_EMAC.c

```
/*-------------------------------------------------------------------------*
 * Constants:
 *-------------------------------------------------------------------------*/
/* EMAC Memory Buffer configuration for 15.1K Ethernet RAM. */
#define NUM_RX_FRAG         8           /* Num.of RX Fragments X*1536= Y kB */
#define NUM_TX_FRAG         2           /* Num.of TX Fragments X*1536= Y kB */
#define ETH_FRAG_SIZE       1536        /* Packet Fragment size 1536 Bytes  */
#define ETH_MAX_FLEN        1536        /* Max. Ethernet Frame Size         */

/* EMAC variables located in 16K Ethernet SRAM */
#define RX_DESC_BASE        ((TUInt32)&G_emacMemory) //.emacmem RAM to be placed
#define RX_STAT_BASE        (RX_DESC_BASE + NUM_RX_FRAG*8)
#define TX_DESC_BASE        (RX_STAT_BASE + NUM_RX_FRAG*8)
#define TX_STAT_BASE        (TX_DESC_BASE + NUM_TX_FRAG*8)
#define RX_BUF_BASE         (TX_STAT_BASE + NUM_TX_FRAG*4)
#define TX_BUF_BASE         (RX_BUF_BASE  + NUM_RX_FRAG*ETH_FRAG_SIZE)
#define TX_BUF_END          (TX_BUF_BASE  + NUM_TX_FRAG*ETH_FRAG_SIZE)
#define EMAC_MEMORY_SIZE    (NUM_RX_FRAG*8 + NUM_RX_FRAG*8 + NUM_TX_FRAG*8 + \
                             NUM_TX_FRAG*4 + NUM_RX_FRAG*ETH_FRAG_SIZE + \
                             NUM_TX_FRAG*ETH_FRAG_SIZE)
```

- Updated memory tests and GPIO loopback tests are included from factory test improvements.
- DKTS CARRIER Projects from Project Maker must be edited to update the CARRIER revision, LCD driver, and touchscreen driver setup to match the target hardware.
- In lwIP, a new lwippools.h was added that includes 1536 bytes for pbuf.
- lwIP memory pool Packet buffers are now placed in the SDRAM ".network" section.
- Additional lwIP FreeRTOS functions were added which should improve lwIP reliability with thread lock handling.
- More projects now include the "DEBUG" and "NDEBUG" defines and more features now use those flags to enable or disable test code (FreeRTOS and lwIP).
- lwIP assert is now functional and included by default in debug builds with the "DEBUG" define included in the library and application projects. Projects now include build system file path and file line information in the executable output, unless disabled. This can print to the serial console the exact lwIP ".c" file and line number when a failure is detected, such as a null pointer, missing buffer, bad calling parameter, etc.
- In debug builds, task information can now be retrieved on the console using "TASK" command, if configUSE_TRACE_FACILITY is set to 1 in the project.
- uEZ Project maker no longer requires installation and by default will now automatically have the correct uEZ folder path set to the working directory, since the executable is run out of that directory now.

- New SDRAM tests are included in the noSDRAM projects. This project was designed to perform a comprehensive test of the installed SDRAM, but not to put any files or data into the SDRAM.
- LPC1788 and LPC4088 uEZGUIs can now use the UART even before the RTOS/uEZ system is initialized (setup), using raw register setup code. This allows error messages to be printed during bootup or if a failure condition occurs.

ERATTA

- Do not use USB MSC with the LPCxx88 as it is not supported. (If this was supported it would allow you to plug in an SD Card into the SD Card connector of the uEZGUI, then connect USB device form the uEZGUI to a host PC. Finally, in file explorer of the host PC, you would be able to access the files on the SD Card.)
- When performing extensive file access of an SD card in the uEZGUI-4357-xxWVN, the functions may time out and return a failure condition. To avoid failing, it is recommended to automatically recover using error handling when this situation occurs. (At FDI we apply a brief delay, then repeat the SD card access.)
- lwIP 2.xx has not been extensively tested at FDI. If issues are found, please contact technical support at **support@teamFDI.com**.
- At the time of this release, uEZGUI-4357-xxWVN does not have the ability to output information over the UART early in the boot cycle. (This is enabled on the rest of the uEZGUI devices.)

uEZ v2.11 – 06/05/2020

HIGHLIGHTS

- SEGGER emWin updated to version 5.48 from NXP for LPC MCUs:
  - Added full set of NXP licensed emWin development tools from NXP.
  - Added correct matching version of emWin manual.
  - Updated emWin GUIDEMO to show correct colors with new version of emWin.
  - See project maker for new generation example.
  - See emWinColorUpdate.pdf.
- Added Ozone projects to all uEZGUI demos and project maker projects.
  - RTOS aware debugging now possible with FreeRTOS.
- Updated projects to support Crossworks v4.7.0.
- Updated projects to support IAR EWARM v8.50.
  - Added EWD files to IAR projects to retain JLink debugger settings.
- Added DHCP support.
- SEGGER RTT updated to version 6.70.
  - Assembly version added that may bring more performance.
  - Several bugs were fixed from SEGGER where signed integers were replaced with unsigned integers.

- SEGGER SystemView updated to version 3.10.
- The NXP TDA8551 audio amp is discontinued and will be phased out of all uEZGUIs over time.
  - Updated the following 3 units' platform files to allow autodetecting when LM4811 I2C audio amp is present using I2C ack, then use new amp.
    - uEZGUI-1788-43WQR
    - uEZGUI-1788-70WVT
    - CARRIER_LPC1788
- Added example JLink programming scripts for different LPCs:
  - This includes setting ECRP correctly on LPC54608 when SDRAM is used.

DETAILS

- uEZ library projects now have only one Config_Build.h file that is shared for each toolchain project.
  - Fixed inconsistencies between different copies.
- Added a document explaining the procedure to update existing emWin GUIs to be 5.48 compatible.
- uEZ audio mixer and drivers now set the correct volume on both TDA8551 and LM4811 audio amps from 0 to 255.
- Implemented platform dependent maximum on-board speaker volume control. This ensures a maximum of 0.5W is delivered to the BeStar Speaker.
- Changed most int types to int32_t so that they won't change size when using a 16-bit/64-bit MCU.
- Used <stdint.h> so that we use common integer types in uEZTypes.h across all compilers.
- Fixed the integer size of time seconds counter to eliminate bugs that only occur on specific MCUs.
- Converted HTTP server to socket API which results in performance improvements.
- HTTP server now supports using "ENABLE_HTML_PARSED_VARIABLES" to enable or disable the live variable update in the HTML file.
- Using "ENABLE_HTML_QUESTION_IN_URL" question marks are now allowed in the HTTP server URL.
- Added an untested option to the HTTP server demo to use a virtual file system.
- Fixed LPC4357 GCC printf by making sure we don't override __getchar and __putchar functions with empty ones in thumb_crt0.s.
- Fixed having duplicate copes of Global.h.
  - Added PRT_ADDR type to Global.h.
- Fixed SystemView and RTT Buffer section placement.
  - Added .rtt and .rttbuf sections as standard.
  - Normally .rtt will be in internal RAM to be easy to find, and .rttbuf will also be placed in internal RAM to avoid problems running memory tests.
  - Placed .rttbuf in SDRAM when using SystemView with heavy trace usage.
- Added real names to SystemView that will show up on a PC when used.

- Fixed an issue with UEZTimeDateAddDuration reported by a customer. Confirmed that various dates can be correctly added.
- Updated SplitLPC1788HexIntoBins for LPC546XX.
  - Fixed missing printf arguments in this application.
  - Project now compiles in VS2017.
  - Increased max QSPI size to 32MB.
  - Changed name to SplitLpcHexIntoBins.
- Added UEZBSPDelay functions to uEZPlatform API.h They should be used sparingly.
- Added UEZTaskDelay(1); to USBMSDrive.c to prevent slow bootup when SD card and USB device MSC are used.
- Used the correct identifier for IAR in library Config_Build.h, which is __IAR_SYSTEMS_ICC__ for the compiler.
- Used the correct identifier for GCC in library Config_Build.h, which is __GNUC__ for the compiler.
- Inverted the uEZ memory defines order (variable, placement) to (placement, variable) so that we can set multiline variables using them.
  - This is used in the project maker projects for emWin memory placement. The memory will grow based on number of frames or widgets defined in Config_App.h. The memory will automatically place itself correctly to avoid wasting RAM.
- Fixed stack overflow on Chksum task on LPC43XX.
- Fixed missing QSPI on LPC4088 Crossworks memory map.
  - Used linker_memory_map_macros to change SDRAM/QSPI size.
- Cleaned up crossworks memory map for LPC43XX.
- Removed various instances of hardcode GPIO port/pin in application code. Used platform definitions instead.
- Removed various instances of hardcode SDRAM size in application code. Used platform definitions instead.
- Added factory speaker test into uEZGUI-1788-70WVT project.
- Added touchscreen tests and accelerometer tests to MainMenu.c that can be turned on using a define.
- Cleaned up accelerometer drivers.
- Added floating point output support to LIS3DH driver.
- Fixed/Updated the memory placement/mapping on all uEZGUI demos and project maker projects.
  - Added variables to Crossworks projects to allow for easily changing the start address for use with a bootloader.

uEZ v2.10 – 02/06/2019

HIGHLIGHTS

- KEIL support removed

- Renesas RX support removed
- Added SEGGER RTT
- Added SEGGER SystemView
- FreeRTOS update to v9.0.0
  - SystemView added into all Cortex-M files
  - SystemView tested only on CM3/CM4
- Tracealyzer updated to work with FreeRTOS version 9
- Free Modbus updated to v1.5.0
- Add LWIP version 2.0.X
- Add new general-purpose section placement macros that automatically translate into compiler specific section placement for IAR, and GCC (Crossworks)
- Updated the project maker example projects to include Micrium µC/Probe example projects to use with the demo. These were tested against µC/Probe v4.5.0
- Updated projects to support Crossworks v4.3.2
  - Breakpoints now work in debug builds in Crossworks 3.X/4.X
- Updated projects to support IAR EWARM v8.30.1
  - Includes fix for future IAR releases where a header file will be removed
- Updated all IAR projects to C99/C11 mode.
  - IAR no longer has C99 only mode, as their C99 has basically been C11 compatible for multiple years already.
- Updated all Crossworks projects to C11/C+11 mode to be in line with IAR.
- Added serial flash drivers for Micron M25P80 and Macronix MX25L8006E.
- Added GainSpan roaming mode support.
- Fixed SD card related issues on LPC43xx. The previous error lead to corruption or initialization failures.
- Added new platform specific functions to platform files:
  - UEZPlatform_LED_Require and UEZPlatform_Set_LED for platforms such as uEZ-GW1.
  - UEZPlatform_System_Reset for common reset API that will use a hardware GPIO triggered full system POR where available or NVIC_SystemReset when not available.
- Added support for updated NewHaven NHD50800480TF LCDs using NewHaven_NHD50800480TF_Rev2.c (new driver chip required new timing).
- Updated all LPC projects to use the same CMSIS files for all toolchains, simplifying the code and fixing various random per-toolchain regressions.

DETAILS

- Added new library directories for SEGGER's RTT and SystemView as separate components.
  - SystemView requires RTT and FreeRTOS
  - RTT can be used by itself.
- For all LPC1788/4088/4357 IAR and Crossworks projects update the project placement files (usually called uEZGUI-1788.icf, etc) to include placement information for SEGGER's RTT main structure, SEGGERs RTT buffer structure, and frames placement. This allows for the following:
  - Proper example of using section placement macros for RAM, including release builds with optimization turned on
  - RTT section is always in SRAM, so the SEGGER RTT viewer and Crossworks RTT viewer (using SEGGER) can automatically find the structure when using RTT or SystemView.
  - RTT buffer section is always in SDRAM, so that large allocation can be done for the RTT consoles or SystemView trace buffer
  - LCD Frames section now properly implemented so that if the number of buffered frames is changed the placement will automatically be changed and reflected in the IDE's RAM usage. (For example, the Crossworks SDRAM % graph now updates correctly on code change.)
    - This change freed up RAM in some projects, making it easier to use the RAM for other purposes, such as networking or SystemView.
    - This change only applies to project maker projects.
- Updated all Crossworks projects to (where possible) include the settings in the project's common section, not the solution section.
- Updated all Crossworks projects to reorganize the settings into common debug/release/debug trace builds.
  - This fixes mismatched settings between projects.
- Updated all Crossworks projects to including missing settings such as:
  - GCC mode
  - C++ preprocessor definitions
  - Compiler mode selection
  - Debug settings that new versions of Crossworks expect.
    - This fixes issues setting breakpoints in newer Crossworks releases.
- Bug Fixes for LPC43XXX:
  - Fixed USB Host, Ethernet, various issues on Crossworks projects.
  - Fixed Crossworks memory mapping and interrupts.
- Fixed serial driver names in LPC code to fix a name regression.
- Crossworks CMSIS updates
- Fixed Crossworks Cortex-M4 issues.
- Update NXP UART code to allow enabling HW flow control pins
  - Limited testing was done on LPC43XX, but not on other platforms
  - Placeholder for required registers is now included for other LPCs
- Added missing FreeRTOS files for CM0, CM7, and Cortex-M MPU ports

    - These are currently unused in existing demos.
- LPC43XXX: Fixed I2C pin assignment issue.
- FreeModbus: Allow server to keep running even if socket error occurs.
- Added GainSpan NCM Auto Config command and roaming mode example.
    - Note: Requires new custom built GainSpan firmware.
    - Contact FDI or Telit for new firmware builds due to website changes.
- Fixes to GainSpan scan code regression.
- Fixe uEZDACWAVFile.c file name.
- Fixe file names of CMSIS LPC header files to be consistent.
- Removed duplicate Global.h files.
- Added preview DHCP support into LWIP 2, but currently no demo project exists.
    - User cannot switch yet between DHCP and static IP at runtime.
- Fixes to FreeRTOS priority settings in uEZ:
    - ARMCM0 section added
    - Fixed LPC 1788/4088/43XXX priority levels.
- Fix #include paths to specific versions of LWIP so that library/application code will compile against newer versions of LWIP.
- Removed cmsis_iar.h include and replace it with intrinsics.h per IAR.
    - Changed the __disable_irq to __disable_interrupt which is in the new intrinsics.h. We confirmed that as of IAR 8.22.2 these use the same internal function call to __iar_builtin_disable_interrupt.
    - This change has no effect on the behavior, but it is necessary to compile with a future release of IAR. In IAR 8.2X+ you would get lots of warnings without this change.
- Added proper __putchar and update putchar override for Crossworks version 4.2
    - Due to the changes in Crossworks 4.2 stdio.h, we must define the prototype for the int __putchar(int, __printf_tag_ptr); function.
    - https://www.rowleydownload.co.uk/arm/documentation/index.htm?https://www.rowleydownload.co.uk/arm/documentation/arm_release_notes.htm
    - https://rowley.zendesk.com/hc/en-us/articles/210033043--undefined-reference-to-putchar-error-message
- Added preliminary support for Orient AFY480272B0-4.3N12NTM.
- Fixed the LPC43XX platform files so that SD card power sequencing is correct. This fixes the issue where initialization would fail on certain cards.
- Fixed the LPC43XX SD card driver to insure byte alignment for the DMA controller. This fixes the issue where bytes were dropped.
- Minor updates to touch detection threshold on specific Focaltech touchscreens used on Newhaven displays.
- FPU support fixed on LPC408X and LPC43XX, but not tested.
- Fixed various touchscreen/calibration screen related bugs.
- Fixed various startup bugs.

- Fixed LPC43xx RTC support with proper initialization and saving routine.

ERATTA

- In IAR version 8 when using emWin the following warning "Warning[Lt009]: Inconsistent wchar_t size" will show up due to the existing library from NXP being compiled with IAR version 7. uEZ software does not use wchar_t and does not pass wchar_t types to emWin functions, so this warning is not an issue. emWin will need to be updated to a new version built off of a new version of IAR to fix this.
- When using the LPCXX88 uEZGUI as a USB MSC to read an SD card, the files read may be corrupted.
- On some uEZGUIs the accelerometer demo does not function.
- GPIO drivers are not complete on LPC43XX.
- DK-57VTS-LPC1788 CAN loopback portion of built-in FCT no longer works. This will be fixed in a future release.
- Some compilers now output a binary file using the new DWARF4 file format. Currently this format is not compatible with □6c/Probe. The workaround is to disable the DWARF4 format in the compiler until µc/Probe is updated to support it. See the Project Maker guide for more details.
- On uEZGUI-4357-50WVN with uEZGUI-EXP1 the USB host must use a GPIO controlled power pin which does not always work due to timing. The unit may need to be powered off, then on for the USB host to show up.
- Using RTT may generate a verify warning in Crossworks where the memory will be shown initialized on debug start, but not in executable.
- The Crossworks project for the uEZGUI-4357-50WVN has the following issues that will be fixed in a future release:
    - USB host driver works intermittently.
    - Slow video playback.
- The USB device mass storage driver for the uEZGUI-4357-50WVN does not work.
- uEZ-GW1 not included in this release as the demo was not yet updated nor is hardware available in distributor stock. A future release will include an updated UbiquiOS demo with multiple interface support.
- LPC43xx RTC code currently takes a long time to save a new date and time, sometimes more than 10 seconds.
- LPC43xx Demos may crash if the user doesn't wait for the RTC to save before switching to another screen.
- LPC43xx uEZTaskDelay timing may be much less accurate than on other LPCs.

uEZ v2.09 – 09/08/2017

HIGHLIGHTS

- For all LPC1788 IAR and KEIL projects update the project placement.icf file (usually called uEZGUI-1788.icf) to ensure that LPC17xx_40xx_PLL.o is

placed into SRAM. This is accomplished by renaming all references to LPC1788_PLL.o to the correct file name. This update also applies to the Project Maker and Start Here projects. It is highly recommended to apply the ICF file update to all existing IAR and KEIL LPC1788 projects using uEZ 2.06 to uEZ 2.08 particularly if software reboot is desired. This change does not apply to library projects. This fixes a regression introduced in uEZ 2.06.

- Updated the project maker example projects to include Micrium µC/Probe example projects to use with the demo. These were tested against µC/Probe v4.
- Updated projects to support Crossworks v4.0.4
- Updated projects to support IAR EWARM v8.11
- Updated projects to support KEIL 5.24.2.0
- Updated all KEIL projects to C99 mode to be in line with IAR and Crossworks.
- Updated FATFS to V0.12B.
- Added preview support for CLANG compiler in Crossworks v4 (Use the build configuration drop-down menu to select CLANG or set compiler type to clang.)
- Added preview support for NXP LPC54608 (under LPC546xx directory) for IAR
- Added preview project to support uEZGUI-35CP
- Added support for Tianma TM035NBH02_09 3.5" LCD
- Added preview support for uEZ-GW1
- Added preview software support for UbiquiOS Wireless Stack
- Added LAIRD Wi-Fi/BT/BTLE module support through Ubiquios library
- Added support for DIGI brand XBEE module cellular modem (P/N: XBC-V1-UT-001) for use on the Verizon network, support added through Ubiquios library
- Added Bluegiga BTLE support (BLE121)
- Updated LPC1788 and LPC4088 KEIL projects to work in KEIL v5 using legacy support mode.
- Updated to allow LPC4357 to run the SD card at 51MHz clock speed.
- Removed the BSOD (Blue Screen of Death) as it does not work as intended in modern uEZ releases.
- Added support for Renesas R1EX24512ASAS0A I2C EEPROM.
- Added Touchscreen noise detection application window to uEZDemoCommon. This can be used to detect phantom touches on a noisy PCAP touchscreen.
- Added SD card reliability test application window to uEZDemoCommon. This demo reads and writes files to SD cards using periodic unit reboots. This demo includes an example of using the watchdog timer.
- Removed debug delay code from NetworkStartup.c which improved boot-up time when networking is enabled.

DETAILS

- Fixed PWM example in doxygen code
- Removed printf debug statements from USB code

- Updated BMPSave code to work with the current version of uEZ.
- Removed unused emWin files from older versions.
- Updated LCD driver timer variable to volatile to fix issues with KEIL release builds.
- Fixed LPC1788 IAR library project #define to be set to LPC1788 instead of LPC4088.
- Changed DK-TS to enable the 4 button set by default, so the button board #define can be used for a plug-in button board for testing GUIs without touch screens.
- Changed LOAD_SPACE in various demos to the new LCD_FRAMES defines instead. (LCD_FRAMES_START, LCD_FRAMES_END, LCD_FRAME_BUFFER, LCD_FRAME_SIZE, LCD_DISPLAY_BASE_ADDRESS, etc)
- Removed LOAD_SPACE from LPC based demo projects, but it is kept in RX based projects.
- Updated and improved the bowling demo
- Added bowling demo support for the LPC4357
- Added option to disable audio support without needing to modify existing code.
- Fixed JTAG mode support on uEZGUI-4357-50WVN IAR project.
- Updated watchdog doxygen example
- Increased maximum number of characters for the FDI Command console to 256 from 78 to support Ubiquios key provisioning.
- Added ethernet support to uEZGUI-4357-50WVN-BA project.

ERATTA

- DK-57VTS-LPC1788 CAN loopback portion of built-in FCT no longer works. This will be fixed in a future release.
- Some compilers now output a binary file using the new DWARF4 file format. Currently this format is not compatible with µc/Probe. The workaround is to disable the DWARF4 format in the compiler until µc/Probe is updated to support it. See the Project Maker guide for more details.
- The Crossworks project for the uEZGUI-4357-50WVN has the following issues that will be fixed in the next release:
    - USB host drivers work intermittently
    - Ethernet does not work
    - Screen saver does not work
- The USB device mass storage driver for the uEZGUI-4357050WVN does not work. This will be fixed in the next release.
- The SD card driver for the uEZGUI-4357050WVN has an issue where it drops some bytes on a file read. This is most noticeable using the slideshow demo. This will be fixed in the next release.

uEZ v2.08.002 – 09/30/2016

HIGHLIGHTS

- Update Start Here and Start Here End main.c files for the Start Here guide for uEZGUI-4088-43WQH-BA

uEZ v2.08.001 – 08/31/2016

HIGHLIGHTS

- Update Start Here files for v1.11 Start Here guides for all production LPC based uEZGUIs (uEZGUI-4088-43WQN-BA, uEZGUI-1788-43WQR-BA, uEZGUI-1788-56VI-BA, uEZGUI-1788-70WVT-BA, uEZGUI-1788-70WVM-BA, uEZGUI-4357-50WVN-BA)

uEZ v2.08 – 08/05/2016

HIGHLIGHTS

- Added support for NXP LPC4357
- Added project to support uEZGUI-4357-50WVN-BA
- CyaSSL Encryption Source code
- Support for Modbus RTU in FreeModbus
- LPCUSBLib for LPC4357
- Mini-XML, ANSI- C Library to read and write XML and XML like files.
- Updated compiler version for all IAR and Crossworks projects
- Added USBTMC driver to library
- Added VNC Client driver to library
- Added HTTP Client driver to library
- USB host support for USB to RS232/TTL serial FTDI ICs.
- Update GainSpan Wi-Fi driver to use threading and interrupt driven I/O
- Update emWin to 5.30 version.

DETAILS

- Fix some demo projects and project maker projects not properly set to Cortex-M4 mode for LPC4088
- Hard float support for LPC4088 and LPC4357 enabled in all IAR/Crossworks projects using emWin hard float libraries.
- Removed version numbers from all Crossworks and IAR projects. To update existing projects to uEZ 2.08, the file path of the library file must be changed to match.
- Updated NetworkStartup to not start HTTP server if network not active
- Update IRD LPC1768 project to latest Crossworks release. Most features tested and working, except for USB and Ethernet which will be added in a future release.
- CRC command line test added
- Added external Interrupt HAL drivers to LPC libraries
- Change GainSpan Wi-Fi flow control setting to fix issues with GS2011M
- GainSpan Wi-Fi driver now checks for presence of module before configuration

- Reliability updates to HTTP Server
- Updated LPC17XX/40XX SSP driver to toggle chip select between bytes.
- uEZGUI-4088-43WQN-BA now uses M555 emWin color driver
- Added ST Accelerometer to uEZGUI projects for uEZGUIs that will include it in upcoming builds. (uEZGUI-1788-43WQR-BA, uEZGUI-1788-70WVM-BA, uEZGUI-1788-70WVT-BA, uEZGUI-4088-43WQx-BA, uEZGUI-4357-50WVN-BA) Auto-detection code figures out which accelerometer is loaded on the uEZGUI. See PCN for your specific uEZ GUI P/N for complete details.
- Add drivers for Newhaven 5.0" WVGA display
- Update some uEZGUI platform files to fix UART name issues.
- Updated Doxygen files.
- Fixed missing function in LPC1788 PWM driver
- Updated all LCD drivers to use passed in address, which is needed for LPC43XX and future micro-controller support.
- Update other hardcoded memory addresses in uEZ to be compatible with new MCUs.
- BSOD removed due to changes in FreeRTOS to catch critical nesting issues. The BSOD code is handled in an exception and uses uEZ API drivers and require RTOS features. This change disables the BSOD.
- Update SplitLPC1788HexIntoBins for LPC4357 support. (also supports LPC40XX)

uEZ v2.07 – 06/26/2015

HIGHLIGHTS

- FreeModbus with TCP/IP support added.
- emWin library files updated to version 522.
- New emWin temperature graph example released
- Add CAN drivers to LPC1788/LPC4088
- FATFS updated to version R0.11
  - Includes optional long file name support
- FreeRTOS updated to v8.1.2 with Tracealyzer v2.6.0 and Cortex-M4F support.
- Floating point mode now supported on the LPC4088 builds using the SoftABI mode.
  - New LPC4088 Cortex-M4F CMSIS files and other low level code.
  - See here for details: http://www.lpcware.com/content/faq/lpcxpresso/cm4-floating-point
- Boot-up speed has been improved on some uEZ GUIs.
- USB-HID Host Mode device driver now included with mouse/keyboard example code for LPC1788/LPC4088.

DETAILS

- Updated LPC1788/4088 I2C registers to meet the latest recommendations from NXP for the duty cycle requirements of I2C Fast Mode Plus (1MHz). Reference NXP I^2C-bus specification and user manual (UM10204).
- Improved the following LCD drivers with a new common driver structure:

  - Newhaven 4.3” display used on uEZGUI-4088-43WQN-BA
  - Microtips 7.0” display used on uEZGUI-1788-70WVM-BA
  - Sharp 4.3” display used on uEZGUI-1788-43WQR-BA
  - Tianma 7.0” display used on uEZGUI-1788-70WVT-BA
  - Inteltronic 5.6” display used on uEZGUI-1788-56VI
  - Okaya 5.7” display used on DK VTS kits
- Simplified lwIP options file to use a single version in uEZ with separate LPC/RX configurations.
- Updated copyright information at top of most files.
- Improved the reset pulse width and timing during initialization of the FT5306DE4 I2C capacitive touch controller on the uEZGUI-4088-43WQN-BA.
- Update uEZGUI platform files to be more consistent with respect to features such as UART routines, console options, GainSpan programming modes, etc.
  - Fixed some require routines, such as I2C2 on the uEZGUI-4088-43WQN-BA which has different expansion pins than other uEZGUIs.
  - Added RS485 Half Duplex to uEZGUI platform files.
- Update GainSpan Wi-Fi driver to handle BSSIDs.
- Update GainSpan Wi-Fi driver to handle Wi-Fi module not being present. (run-time detection)
- Added support for PMOD use of GS1011/GS2011 GainSpan-WAB Wi-Fi modules on the uEZGUI-1788-70WVM-BA and uEZGUI-4088-43WQN-BA.
- Update GainSpan programming routine for both program and run mode support for programming external SPI flash through the GainSpan module through the serial port connected to a PC.
- New Loopback expansion board explicit compile-time switches to replace detection code.
- New 72MHz compile option for LPC1788/4088 platforms.
- Updated uEZ DAC WAVE file playback and SWIM GUI demos to avoid memory leaks.
- Updated case and path names to fix compile issues on Linux, tested using standard Crossworks 3.2 builds.
- Remove LPC2478 Flash driver and miscellaneous old references to LPC2478.
- Remove RedPine Wi-Fi drivers that are no longer supported in hardware or software on uEZGUIs.
- Update uEZ PINLOCK BSOD in release mode to print out which pin is causing the crash.
- Update Sitronix ST1232 touchscreen driver for increased y-axis accuracy on the uEZGUI-1788-70WVM-BA
- Add timer create routines to platform files.
- Fixed bugs in the LPC1788/LPC4088 timer driver.
- Updated GPIO test command for 3 pin loopback test when odd number of pins to be tested.
- Checksum added to SWIM demo title bar

- Update default FRAMES define in CrossWorks builds to have more SDRAM available for other uses.
- Updated uEZ DAC WAVE file playback to avoid static noise when playback starts/stops on certain wave files.

uEZ v2.06c – 05/08/2015

HIGHLIGHTS

- Updated LPC1788/4088 I2C registers to meet the latest recommendations from NXP for the duty cycle requirements of I2C Fast Mode (400KHz). Reference NXP I^2C-bus specification and user manual (UM10204).
- Improved the following LCD drivers to include more accurate initialization timing:
    - Newhaven 4.3" display used on uEZGUI-4088-43WQN-BA
    - Microtips 7.0" display used on uEZGUI-1788-70WVM-BA
    - Sharp 4.3" display used on uEZGUI-1788-43WQR-BA
    - Tianma 7.0" display used on uEZGUI-1788-70WVT-BA
    - Inteltronic 5.6" display used on uEZGUI-1788-56VI
    - Okaya 5.7" display used on DK VTS kits
- Removed hard-coded delays in the LPC1788/4088 LCD driver. All delays/timing are now set only inside of the LCD specific driver.
- Application Demo change
    - uEZ SWIM Demo - Startup routine simplified
- Added support for new STM LIS3DH accelerometer used on the uEZGUI-1788-70WVM-BA hardware revision 2.0.
    - The require routine dynamically checks which accelerometer is present on the PCB, so no specific configuration is necessary as long as the standard platform file is used.
- Modified the UEZBSPDelay1MS() function defined in the platform files of LPC1788/4088 uEZGUIs to improve the accuracy of the delay.
- To reduce the risk of issues such as locking-up the capacitive touch controller on the uEZGUI-4088-43WQN-BA and uEZGUI-1788-70WVM-BA units, we now turn on the LCD power early in the initialization sequence and no-longer allow the LCD power to be turned off during normal operation.
    - The new #define option applies to the uEZGUI-4088-43WQN-BA revision 1.X and uEZGUI-1788-70WVM-BA revision 1.X/2.X only.
    - Future revisions of these PCBs may no longer require this fix.
- Updated NOR Flash timing to match the latest worst case timings published by NXP for 90nS devices.
- Increased the LPC4088 SPIFI clock to run at half the CPU clock for improved performance.
- Improved the reset pulse width during initialization of the ST1232 I2C capacitive touch controller on the uEZGUI-1788-70WVM-BA.

uEZ v2.06b – 09/15/2014

HIGHLIGHTS

- Added support for Crossworks 3.2 on LPCXX88 products that are in production
- Updated documentation package, users manuals, and quick start guides

uEZ v2.06 – 1/3/2014

HIGHLIGHTS

- Added support for LPC4088 MCU
- Expansion board files for uEZGUI-EXP-DK
- Processor name change for LPC1788 to support both LPC1788 and LPC4088
- Removed support for LPC2478 and LPC2478 based products
- Added new platform for uEZGUI-4088-43WQN

DETAILS

- uEZ General Changes
  - emWin updated to v5.22
  - FreeRTOS updated to v7.5.3
  - Updated IAR to v6.50
- Application Demo changes
  - uEZ SWIM Demo
    - Startup routine simplified
    - FreeRTOS Trace Plus updated
    - uEZGUI-4088-43WQN project added
- Build Changes
  - Added IAR 6.50, KEIL4, and CrossWorks 2.3 projects for LPC4088
- HAL/Device Driver Changes
  - Audio Amp TI LM48100
    - Timeout decreased on I2C writes
  - EEPROM NXP LPC1788
    - Name changed to LPC17xx_40xx
  - Flash
    - LPC1788 IAP changed to LPC17xx_40xx
    - Added LPC SPIFI M4 Library
    - Spansion S29GL added retry IWaitStatusDoneOrError
  - LCD
    - Added support for Inteltronic LMTDA043ZHN3 4.3” display
    - Removed unused code from Newhaven LCD driver
  - Tone generator
    - Changed PWM driver to set the pin to output mode to reduce noise when audio is not playing
  - Touchscreen
    - Added test code to touch screen drivers
    - Added support for Himax PCAP controller
    - Cleaned up Sitronix and Newhaven drivers to be cleaner and more responsive
  - USB Device

    - Changed name from LPC1788 to LPC17xx_40xx
- Library Changes
    - Audio
        - Added support to DAC audio driver to mute and unmute the audio when not playing to eliminate noise from the speaker
    - GUI
        - Added emWin support for v5.22 and M4 libraries
- Platform changes
    - Removed LPC2478 based platforms
    - Added uEZGUI-4088-43WQN platform
    - Updated LPC1788 based platforms for processor name change
    - Set uEZGUI platforms to mute audio
- Processor Changes
    - Removed LPC2478 support
    - Changed LPC1788 to LPC17xx_40xx and added LPC4088 support
- RTOS Changes
    - Updated FreeRTOS to v7.5.3
    - Updated FreeRTOS Plus Trace to v2.2.3
- uEZ Changes
    - Added UEZDeviceTableIsRegistered
    - Moved vTraceInitTraceData call to uEZInit, for Trace builds
    - Increased size of TSMon stack to support PCAP I2C controllers

uEZ v2.05 – 6/20/2013

HIGHLIGHTS

- Audio Mixer system added providing unified audio control across all hardware configurations. Audio Amp levels are normalized to be 0-255 instead of custom values per amp.
- Input Events are now used by both keypads and touchscreens in a single unified structure.
- Improved FreeRTOS+Trace support with named queues and semaphores. Many queues and semaphores are now named when UEZ_REGISTER is enabled.
- FreeRTOS+Trace added to all platforms (where possible)
- uEZ Button Board support added allowing keypad to control displays that do not have an integrated touchscreen.
- Better handling of USB Host port and detecting if located on uEZ GUI or on Expansion Board.
- Improved RESET timing when connected to LPC1788 processors with a debugger using IAR, Crossworks, or Keil.
- Improved Network Startup code using Bring Up/Take Down commands. Unified for both wired and wireless configurations.

DETAILS

- uEZ General Changes
  - emWin updated to v5.20
- Documentation
- Application/Demo Changes
  - uEZGUI Demos & DK-TS Kit Demos
    - uEZ Button Board support added.
    - HTTPServer instead of BasicWeb is default now.
    - Video Player added (for platforms with MCI based high speed SDCard)
- HAL and Device Driver Changes
  - Audio Amp
    - NXP TDA8551 – Changed to support 0 – 255 input level.
    - TI LM48100 Audio Amp added.
    - Wolfson WM8731 Audio Amp added.
  - Expansion board
    - Added uEZGUI-EXP-DK expansion board files.
  - GPIO
    - PCF8574T port expander code added as a HAL_GPIO driver.
    - PCF8575 port expander code added as a HAL_GPIO driver.
  - I2C
    - Extended I2C API
      - Is Hung
      - Reset bus – not implemented
  - LCD
    - MicroTips UMSH 8065M 21T corrected with proper signal timings and reset pin control.
    - Okaya RH320240T – added control pins for SPI chip select, shutdown, and enable.
  - Keypad
    - Generic GPIO based Keypad driver added (one button per GPIO based on table). Generates input events based on regular polling. All pins must share same GPIO Port.
  - Network
    - Fixed a possible null pointer assignment when disabling Nagle algorithm.
  - RS485
    - Fixed header that was blocking use of RS485 Timer based version with standard RS485 non-Timer based driver.
  - SDCard for MCI
    - Added code to return when already initialized.
  - Tone Generator
    - Frequencies are now generated based on frequency reported by UEZPlatform_ProcessorGetFrequency().
  - Touchscreen
    - Touchscreen now uses input events instead of touchscreen readings when reporting touches.
    - MSTAR MSG2032 Touchscreen driver added.

        - Sitronix ST1232 touchscreen driver no longer uses hard coded GPIO values but passed in GPIO values.
    - USB Device
        - Changed clear and set feature request to return true, to support Macintosh computers.
- Library Changes
    - General
        - emWin updated to v5.20
    - FDICmd
        - Test commands added for audio playback over DAC and I2S
    - SWIM / SimpleUI
        - Keypad Input Event support added and logic added to select buttons in the choices code based on geometries.
    - HTTP Server
        - Settings for source drive and port are now run time configurable instead of compile time configurable.
    - emWin
        - Updated to emWin v5.20 distribution
        - Improved emWin performance by implementing better semaphore blocking. The CPU can now go fully idle when no events are occurring instead of polling.
- Platform Changes
    - General
        - Better handling of USB Host port and detecting if located on uEZ GUI or on Expansion Board.
        - Audio Mixer added on multiple configurations as appropriate. Audio levels now go from 0 to 255.
    - LPC1788 CARRIER
        - Fixed initialization order of LCDs (LCD must be configured, then LCD Controller turned on)
    - LPC2478 CARRIER
        - Fixed initialization order of LCDs (LCD must be configured, then LCD Controller turned on)
    - uEZGUI-1788-43WQR
        - UART4 full duplex require routines added.
        - MCI driver added and attached to MS1 when used as SDCard drive. Needed only for Expansion Boards.
        - Fixed registering of I2S device as I2C device.
        - Button Board create routine added.
    - UEZGUI-1788-43WQS
        - Added UART0 Full Duplex require routine
    - uEZGUI-1788-56VI
        - Added MCI require routine for SDCard support
    - uEZGUI-1788-70WVE
        - LCD Controller no longer uses P2_3 and P2_5 for unneeded frame porch and line porch pins.

        - Added MCI require routine for high speed SDCard support
    - uEZGUI-1788-70WVM
        - Added
    - uEZGUI-1788-70WVT
        - 72 MHz configuration added as compile option
        - Added MCI require routine for high speed SDCard support
        - uEZPlatform_Full_Require added.
    - uEZGUI-2478-70WVE
        - 72 MHz configuration added as compile option
    - CARRIER_RX62N
        - SDRAM configuration changed to use full 16MB
    - CARRIER_RX63N
        - SDRAM configuration changed to use full 16MB
- Processor Changes
    - NXP
        - General (NXP and Renesas processors)
            - I2C API extened
        - LPC1788
            - EMAC driver now has all its memory blocks in one .emacmem block to be put in the AHB peripheral memory space.
            - I2C pins are now properly configured to turn off hysteresis.
            - MCI driver now has all its memory blocks in one .mcimem block to be put in the AHB peripheral memory space. Size was made smaller so it works better with other peripherals.
            - PWM now properly calculates timing based on current PCLK divider.
            - UART4 added.
            - SSP timing calculation properly allows for highest speed now (it would do half).
            - USBHost driver now has all its memory blocks in one .usbhostmem block to be put in the AHB peripheral memory space.
- RTOS Changes
    - General
        - Naming of Queues and Semaphores added (FreeRTOS and SafeRTOS included)
        - UEZDeviceTableRegisterAlias added to allow generic naming of specific devies.
    - FreeRTOS
        - Added ability to see task list in Rowley Crossworks with threads.js script
- uEZ Changes
    - General
        - UEZDeviceTableRegisterAlias now allows devices with the same name to be created.
    - AudioMixer

    - High level Audio Mixer system added for controlling audio output volumes.
- GPIO
    - UEZGPIOReadPort() added for reading complete ports easily by their port identifier.
- Keypad
    - UEZKeypad now requires a queue to receive input events from the keypad instead of polling from UEZKeypadRead (removed).

uEZ v2.04 – 4/2/2013

HIGHLIGHTS

- Added support for 5 different PCAP Touch Screen LCD's in sizes of 3.5", 4.3", 5.0", 7.0" and 10.4"
- Added support for FreeRTOS+Trace for expanded debug and development capability.
- FreeRTOS support upgraded to FreeRTOS v7.3.0
- Segger JTAG SWO Viewer support added where applicable for expanded debug and development capability
- New BMP drawing routines used with new title screen.
- Screen saver features added to dim, animate, and turn off the LCD.
- Networking API improvements to separate infrastructure functions (configure, bring up, take down) from standard use. Also NetStart task created to bring up network.
- Added support for video playback from high speed SDCard at 15fps on a uEZGUI-1788-43WQR with a uEZGUI-EXP-DK Expansion Board

DETAILS

- uEZ General Changes
  - FreeRTOS Trace added as a Debug option to all LPC1788 builds
  - emWin v5.18 upgrade (from emWin v5.14)
  - FreeRTOS upgraded to FreeRTOS v7.3.0 (from v7.0.1)
- Documentation
  - Improved SPI & Timer documentation with examples.
- Application/Demo Changes
  - uEZGUI Demos & DK-TS Kit Demos
    - uEZ GUI Demos now compile with FreeRTOS+Trace enabled
    - New Title Screen using smaller BMP graphic
    - All Expansion Board dependencies in demos have been removed.
    - Draw App now uses dynamic memory instead of LOAD_SPACE to simplify memory map.
    - uEZGUI-RX62N-35QT Functional Test App no longer tries to test non-existent NOR Flash
  - Video Playback Demo added for uEZGUI-1788-43WQR
  - Screen saver enable option added to Brightness app
- Build Changes
  - Added Builds for FreeRTOS+Trace
- HAL and Device Driver Changes
  - Audio Amp
    - AudioAmp TDA8551T properly locks pins dedicated to controlling amp.
  - Audio Codec
    - Wolfson Audio Codec WM8731 is now initialized to use both I2S and DAC audio inputs and play out the headphone jack.
  - GPIO
    - Get Mux routines added
  - GPMDA

        - Burst/Single/Last requests can now be triggered through software
    - I2C
        - Added PCA9645 Mux code. Creates new I2C busses for each mux output and automatically changes lanes as needed.
    - LCD
        - LCD now has vertical sync detection and control (UEZLCDWaitForVerticalSync and callbacks). All LCD drivers have been updated.
        - LCD drivers now take parameters for SPI and GPIO pins in their Create routines instead of being hardcoded.
        - MicroTips AWT 800480T50P03 LCD Driver added.
        - MicroTips UMSH 8253MD 3T LCD Driver added.
        - MicroTips UMSH 8596MD 20T LCD Driver added.
        - MicroTips UMSH 8065MD 21T LCD Driver added.
        - Sharp LQ104V1DG28 LCD Driver added.
    - Mass Storage
        - New High Speed 4-bit parallel SDCard MCI driver added.
    - Network
        - Network API changed to allow infrastructure to be split from opening/closing the network. Infrastructure calls now configure, bring up, and bring down any network.
        - Infrastructure changes made to lwIP, GainSpan, and Redpine network drivers.
        - Option for DHCP client/server added (on networks that support it)
        - Passwords for networks now only in one place.
    - Serial RS485
        - Stability and semaphore fixes.
    - Stream
        - JTAG SWO support added for multiple compilers. Works best with Segger JTAG SWO Viewer.
    - Touch Screen
        - EXC7200 Touch Screen Driver added.
        - Direct Drive Touchscreen now has improved touch detect.
        - MSG2032 Touch Screen Driver added.
        - Sitronix ST1232 Touch Screen Driver added.
- Platform Changes
    - General
        - Removed all expansion board detection calls.
        - Proper detection of USB Host on min-AB connectors on all platforms that support it.
        - All platforms no longer have G_network or G_networkStatus as part of improved networking API.
    - LPC1788 CARRIER
        - GainSpan SPI added as feature
    - LPC2478 CARRIER
        - Fixed pin initialization problem between USB Device and LCD.

- RX62N and RX63N CARRIER
  - Direct Drive code now properly handles UEZLCDSetBaseAddr().
- uEZGUI-1788-70WVT
  - No longer incorrectly enables LCD_FP and LCD_LP pins.

- Processor Changes
  - NXP
    - General (NXP and Renesas processors)
      - GPIO GetMux function added to all processors.
      - Blue Screen of Death moved out of processor files.
    - LPC1788
      - EMAC now uses bit banging to detect presence of EMAC instead of blindly locking up the processor.
      - Blue Screen of Death pulled out of LPC1788 processor code.
      - GPDMA improved for supporting DMA/Peripheral transfers.
      - LCD Controller support for vertical sync interrupt and callback
      - MCI driver added for high speed SDCard support
      - Disabled initial power on of devices. Peripherals are now turned on ONLY by their respective drivers.
      - SSP Driver now properly disableds interrupts when doing polled transactions.
    - LPC2478
      - EMAC now uses bit banging to detect presence of EMAC instead of blindly locking up the processor.
      - LCD Controller support for vertical sync interrupt
  - Renesas
    - RX62N
      - LCD Controller support for vertical sync interrupt
    - RX63N
      - LCD Controller support for vertical sync interrupt

- Library Changes
  - BasicWeb
    - Now uses UEZGetTaskList() instead of FreeRTOS call.
    - Uses new Network API with passed in network handle.
  - BMP Drawing
    - Fixed 1 off width calculation
  - DAC Wave Player
    - Improved support for 8-Bit and 16-Bit playback.
    - Reports location within playback and completion to synchronize with video.
    - Improved stability and buffering.
  - FDI Commands
    - File test routines added
    - I2C commands added (probe, read, write).
  - File System
    - File system volume information now can be read to get used and free size.
    - Sync command added to ensure all data is flushed to media.

- GainSpan AtCmdLib
  - Program and Mode Pins are now locked when used for GainSpan
  - Improved handling of multiple network connections
  - Fixed problem with escape code encoding
- HTTPServer
  - Uses new Network API with passed in network handle.
- lwIP
  - lwIP now properly stops if error from EMAC driver.
  - MAC address now is set through settings passed in when bringing up infrastructure instead of callback.
- Screen Saver
  - New screen saver feature added with callbacks for drawing bouncing logo and dimming of screen. Turns off display after longer period of time.
- uEZFailure
  - New uEZFailure library created to handle Blue Screen of Death.

- RTOS Changes
  - Task priority can now be modified with UEZTaskPriorityGet and Set.
  - UEZGetTaskList added to OSAL for a list of tasks using RTOS independent methods.
  - FreeRTOS+Trace
    - Tasks are now registered with trace (via UEZTaskRegister) automatically.

uEZ v2.03 – 11/29/2012

- uEZ General Changes
  - Improved lwIP / SNMP operability (disabled by default).
  - RX63N CARRIER added with Direct Drive support
- Documentation
  - uEZCRC documented with doxygen.
- Application/Demo Changes
  - uEZGUI Demos
    - emWin initializes only once now in demos.
    - GainSpan logo added to icons.
    - Slideshows with audio only start next slide if audio is complete.
    - Only enable USB Host and Ethernet in demos if enabled in target's Config_Build.h file.
    - Added GainSpan test and program code.
- Build Changes
  - Added batch build files for several more combinations (including RX63N).
- HAL and Device Driver Changes
  - LCD
    - Added Sharp LQ042T5DZ11 LCD Driver
  - Generic RTC
    - Fixed a case in the Generic RTC where Hardware RTCs without a validate command would lock up in a semaphore dead lock state.
- Platform Changes
  - General
    - Added FDI CARRIER_RX63N with Direct Drive support.
  - LPC2478 DK-TS-Kits
    - Fixed Console to properly use UART2.
    - Fixed CAN test in Functional test software
    - Adjusted platform file to reserve upper 8K for bootloader.
  - LPC1788 DK-TS-Kits
    - Fixed CAN test in Functional test software
  - uEZ GUI
    - IAR builds now use a NORFlash_region at link time.
    - POE Expansion board detection no longer disables UART0 TX.
    - New EMC timing parameters and enable page mode.
    - CRC0 driver properly created on all uEZGUI platforms.
  - UEZGUI_1788_56VI
    - Added GainSpan require routine.
    - Added SSP2 require routine.
- Processor Changes
  - NXP LPC1788
    - MATRIXARB added to LPC1788 register list.
    - EMAC PHY address detection no longer requires DEBUG_LPC1788_EMAC_STARTUP fixing expansion boards with EMAC PHYs with non-zero addresses.

        - External Memory controller to NOR flash now has page mode enabled by default increasing the speed of NOR flash accesses. Parameters to EMC driver has changed and are now range checked.
        - Fixed Serial UART code to initialize PCLK dividers and power bits for all UARTs.
    - Renesas RX63N
        - Direct Drive added.
        - RTC driver added.
        - SPI transfer in/out routine fixed. Was not properly placing data into output buffer.

- Library Changes
    - uEZ Network
        - Fixed compile problem caused by disabled DNS option in lwIP.
        - Enabling LWIP_SNMP now works in uEZ Library. LWIP_SNMP_NUM_VARS allocates more memory. Private MIB also supported.
    - uEZ File System
        - File Find sessions now have a NO_DYNAMIC_MEMORY_ALLOC option that avoids requiring UEZMemAlloc.

uEZ v2.02 – 10/16/2012

- uEZ General Changes
  - LPC2478 THUMB Mode now available as compile option for smaller builds.
  - Updated projects to IAR 6.30.x.
  - Updated projects to Rowley Crossworks 2.2.
- Documentation
  - More headers are documented.
  - SWIM library documented.
- Application Changes
  - x
- Build Changes
  - Fixed problem with IAR structures not properly being packed when UEZPacked.h used.
- HAL and Device Driver Changes
  - General
    - More require routines for following drivers
      - PCA9555 (Button and LED driver)
      - Lumex SO2004DSR Character Display
  - Accelerometer
    - Fixed reading range reported by Analog Devices ADXL345
  - CRC
    - CRC Device Driver added. Supports Generic version that communicates with hardware acceleration as well as a Software version.
    - Increased number of CRC supported types in API (currently implementations usually only support a few).
  - GPIO
    - Added Enable/Disable open drain as control option
  - LCD
    - Added KOE TX13D06VM2BAA LCD display
  - Network
    - GainSpan driver added.
  - Serial
    - Using control STREAM_CONTROL_GET_READ_NUM_WAITING now reports bytes in queue waiting to be received.
  - SPI
    - API now sports a non-blocking version of SPI transfers via TransferNoBlock and IsBusy commands at all levels.
    - Device driver supports callback routines when SPI transfers complete.
  - StdInOut
    - Support added RX63N processor and RX Standard compiler.
  - USB Device
    - Support added for LPC1756.
- Platform Changes
  - General

        - Remvoed UEZPlatform_WiredNetwork0_Connect from platforms. These are now up to application to create.
        - IAR ICF files now list FRAME memory, EMWIN memory, and RAM memory in sperate blocks.
        - Added UART0-3 require routines where appropriate.
        - uEZGUI_1788_43WQR Added.
    - FDI CARRIER_RX62N
        - Properly configured for BGA176 RX62N.
    - FDI uEZGUI-1788-43WQS
        - Split USBHost to PortA and PortB require routines.
    - FDI uEZGUI-1788-56VI
        - Split USBHost to PortA and PortB require routines.
    - FDI uEZGUI-1788-70WVT
        - Split USBHost to PortA and PortB require routines.
    - FDI uEZGUI-2478-43WQS
        - Configured to support THUMB mode.
        - Added CRC0 and Watchdog support.
        - Added UEZBSP_VectorTableInit for vector table copying and remapping.
    - NXP IRD 1.0
        - Added back in for IAR compiler.
    - Renesas RDK RX63N
        - Added
- Processor Changes
    - NXP
        - LPC1756
            - Added the LPC1756 with require routines with proper pin mappings.
            - FreeRTOS and SafeRTOS builds for IAR compiler. *(SafeRTOS must be purchased to use build)*
            - Added DAC, BatteryRAM, GPDMA, I2S, and PLL driver
        - LPC1768
            - Added back in with require routines with proper pin mappings.
            - FreeRTOS and SafeRTOS builds for IAR compiler. *(SafeRTOS must be purchased to use build)*
            - Added DAC, BatteryRAM, GPDMA, I2S, and PLL driver
        - LPC1788
            - SafeRTOS build for IAR compiler added. *(SafeRTOS must be purchased to use build)*
            - Improved CRC driver (more features).
            - Turn off debug EMAC output by default.
        - LPC2478
            - Added BSOD.
            - Added Watchdog driver.
    - Renesas
        - RX62N

        - Split out BGA176 require routines from 100 pin routines (ADCBank, EMAC, I2C, and Serial).
        - Added RTC driver.
    - RX63N
        - Added the Renesas RX63N processor
- Library Changes
    - uEZ System
        - Added UEZCRC routines for doing CRC operations (hardware or software based).
        - Added UEZEEPROM for easy access to EEPROM devices.
        - UEZNetwork now has missing UEZNetworkJoin and UEZNetworkLeave commands for associating/disassociating from a network.
        - UEZSPITransferNoBlock and UEZSPIIsBusy added to support newer SPI API.
    - emWin
        - emWin now uses routines UEZEmWinGetRAMAddr() and UEZEmWinGetRAMSize() to setup memory.
    - FAT FS
        - Errors getting the current date will use a date of 1/1/2000 12:00 on files.
    - HTTPServer
        - Now defaults to drive 0: instead of drive 1: for network config.ini settings
    - SimpleUI
        - Initializes now requires using SUIInitialize in addition to SUISetSettings. This configures the SimpleUI state used when a Blue Screen of Death (BSOD) occurs.
        - SUIInitialize sets the screen orientation (flip x, flip y) and passes down to the SWIM library.
    - SWIM
        - Modified the SWIM library to use a new SWIM driver system for handling low level raster drawing.
        - SWIM now supports flipping along the x and/or y axis at runtime.
- RTOS Changes
    - FreeRTOS
        - Added LPC2478 Thumb mode support.
        - Added RX600 support.
- Templates:
    - RX600 template added for both RX62N and RX63N.
- Utilities:
    - Build files
        - Several batch files for the different uEZ libraries have been added.

uEZ v2.01 – 7/16/2012

- Documentation
  - Updated Doxygen configuration file from version 1.7.4 to 1.8.1.1
  - Added extensive number of examples to doxygen output in uEZSystem files.
  - Continued to add additional comments and doxygen tags.
- Application Changes
  - uEZDemos directory now contains all FDI demo code.
- Build Changes
  - Added macro VARIABLE_NOT_USED() similar to PARAM_NOT_USED().
  - Rowley Crossworks now compiles uEZ in EABI format for improved performance.
- HAL and Device Driver Changes
  - LCD
    - Added NewHaven 4.3" NHD43480272 LCD display support.
    - Added default touchscreen coordinates for each LCD in Config_LCD.h. Calibration is still recommended, but these values are approximate.
  - Serial
    - Fixed Half Duplex driver issue with drive enable polarity.
    - Fixed RS485 driver issue with drive enable polarity.
- Platform Changes
  - General
    - All IAR projects now have PLL configuration results stored in internal RAM instead of trying to store in external RAM that has not yet been initialized.
    - Cleaned up the macro NOP() in all platforms so they properly perform one cycle of no operation.
    - Improved .h files for many platforms to more closely match the actual platform configuration.
  - uEZGUI-1788-43WQS
    - Fixed problem with USBDeviceController always being included instead of referenced.
  - uEZGUI-1788-56VI
    - Added support for boot time configuration of USB Host or USB Device depending on actual USB connection since the new rev hardware now supports switching on the fly.
  - uEZGUI-1788-70WVE
    - Fixed LCD Controller not correctly configuring power pin timing.
    - Added UART0 to UART3 requirement and configuration routines.
  - uEZGUI-1788-70WVT
    - Added IAP driver
  - NXP IRD 1.0
    - *Removed. Please use uEZ v1.12 or earlier with this platform.*
  - DK-TS-RX62N Kits
    - Demos updated and now working with HEW
- Processor Changes
  - NXP

  - LPC1768
    - Added support back in for LPC1768.
  - LPC1788
    - Fixed issue configuring A/D pins. A/Ds would always read zero.
    - Runtime serial configuration of parity, stop bit, and parity added instead of compile time.
    - USB Device Interrupts no longer attempt to process if the USB Device driver is not initialized.
    - Exception handler information is now sent to internal RAM instead of possibly external RAM.
    - Watchdog trip now only works if watchdog has been initialized.
  - LPC2478
    - CPUEnableInterrupts and CPUDisableInterrupts moved to processor files.
    - LCD Controller now uses a start-up time of 10 ms instead of an unnecessarily long 500 ms.
    - Fixed error in configuring PLL that caused errant race condition.
    - Better definition of NOP() macro.
    - Added IN_INTERNAL_RAM tag for variables required to be in internal RAM.

- Library Changes
  - SWIM
    - Fixed early return case in swim_put_triangle causing it not to draw.
    - Added function swim_put_text_horizontal_centered for drawing centered text.
  - SimpleUI
    - Removed requirement of ScreenShot() routine to call audio routines.
  - Segger emWin
    - Updated to the latest public EABI releases of library from NXP.
  - Web
    - HTTPServer now looks for on drive 0:
- Utilities:
  - Removed Pins1788ToH utility because the new uEZ system of required routines eliminates the need for this utility.

uEZ v2.00 – 3/27/2012

- uEZ General Changes
  - Extensive changes to the directory structure of the source files, thus the step up to version 2.00.
  - Platform configuration no longer use the multi-tier #define system before. Instead, 'Require' routines are called to add the features needed by the application as provided by the platform system. UEZ_ENABLE_x defines are no longer used.
  - uEZ is now designed to be built as a library and included with an application instead of an all in one project basis. Projects for the libraries are now stored in /Build/Generic/<Company>/<Processor>/<RTOS>/<Compiler>. The libraries are now built and distributed with the source in a Debug and a Release sub-directory to the above.
  - SSP driver type has been removed as a hardware and device driver. The NXP processors that have SSP now have only a SPI interface to them.
  - BSP folder has been pulled into the uEZSystem directory.
  - Many FDI Demos have been folded into the same common source code and projects.
  - uEZGPIO system added to make manipulating IO pins as easy as a uEZGPIOSet(pin) function call. No more HAL_GPIOPort ** references required. NOTE: Not all drivers use this system yet, but will.
  - uEZGPIO Lock system added for noting which pins have been configured and should not be changed without first unlocking.
- Application Changes
  - Slideshows now use an .INI file setting to find slideshows based on the development kit.
  - Renesas RX62N RDK's audio is now working (and with MIDI too).
- Build Changes
  - uEZ is now designed to be built as a library and included with an application instead of an all in one project basis. Projects for the libraries are now stored in /Build/Generic/<Company>/<Processor>/<RTOS>/<Compiler>. The libraries are now built and distributed with the source in a Debug and a Release sub-directory to the above.
- HAL and Device Driver Changes
  - General/API
    - Create and Require routines written using the existing HAL and Device tables and configure routines. NOTE: Configure routines will be merged into Create routines in the near future.
    - Several NXP specific device drivers have been removed and properly replaced with the Generic version.
    - Timer interface added for Device driver.
  - FATFS FileSystem driver
    - Added MakeDirectory command
  - GPIO

        - Configure, Activate, and Deactivate functions are now no longer supported as their functionally are now better served by the Create and Require functions.
    - lwIP Network device driver
        - ResolveAddress command added for DNS lookup.
        - Added Leave command to disassociate from a network.
    - Freescale MMA7455 Accelerometer
        - Improved calculations (2G vs. 8G setting)
    - Generic PWM Controlled Backlight
        - device driver now supports an optional separate pin for controlling backlight power on/off.
    - LCD
        - Digital Image FG050720DSSWDG01 LCD driver added.
    - SDCard
        - The three versions of the SDCard driver have been merged into one (the SPI version) now that SSP device drivers are now SPI drivers.
    - Tone Generator
        - The Generic PWM tone generator can now be configured to fully turn off the PWM port to save power.
        - A Generic Timer-based tone generator with toggle pin output has been added.
- Platform Changes
    - General
        - Highly reworked platform system that uses 'Require' routines instead of #define UEZ_ENABLE_x. This method allows the linker to include only the source that is required automatically.
        - Large amount of file clean up, file name changes, and reorganization.
        - uEZPlatformAPI.h added to allow the platform to report common operating parameters (LCD size, processor frequency, memory organization, etc.)
    - DK-TS-RX62N Kits
        - Less graphics were used to make the demo fit better in memory.
    - Renesas RX62N RDK
        - Fixed improper FreeRTOS configuration (settings and interrupt priorities)
        - Tone generator via PWM added.
    - uEZ GUI Kits
        - UEZBSPDelay now uses nops instead using one of the system timers for doing short microsecond delays.
        - UEZBSPDelayUS() added to all uEZ GUI kits.
        - DAC device added
        - uEZGUI_LPC1788_70WVE now correctly programs the wait states on the flash.
- Processor Changes
    - General
        - Large number of Create routines added to all HAL drivers.

- Standardized the calling of SystemInit at startup before creating the heap and initializing memory.
- GPIO Pin Locking added to all HAL drivers to ensure a port pin is not double configured.
- PLL, SDRAM, and External bus configuration code separated from platform and put into processor code.
- GPIO drivers no longer have Configure, Activate, and Deactivate routines. This is deprecated from the API.

- NXP
  - LPC1756
    - Fixed Watchdog routine ClearResetFlag. Was not properly clearing flag.
    - *Removed temporarily (needs to be changed to the new system)*
  - LPC1768
    - *Removed temporarily (needs to be changed to the new system)*
  - LPC1788
    - Added GPIO Control for Set Config Bits to allow all GPIO settings to be set at once.
    - Fixed Watchdog routines ClearResetFlag and Start. Clear was not resetting and Start would reset the unit.
    - Watchdog now calculates timing based on 500 kHz instead of PCLK.
    - Fixed A/D to return proper size response when other than 10 bits.
    - EMAC correctly looks for PHY address. Was only working correctly in debug version.
    - SSP driver changed over to use SPI HAL driver interface.
    - Timer driver now supports SetMatchCallback and SetMatchRegisterFunctions routines.
    - Blue Screen of Death (BSOD) is now standard in processor routines.
  - LPC2478
    - All references to LPC24xx and LPC247x have been replaced with LPC2478.
    - Flash Acceleration turned on (it was off) for improved performance.
    - Timer now configurations PCONP for proper powered on setting.
    - Moved LPC2478_crt0.s and LPC2478_Startup.s into CrossWorks subfolder.
    - EMAC correctly looks for PHY address. Was only working correctly in debug version.
    - EXT3 interrupt is now properly cleared when activated.
    - Timer driver now supports SetMatchRegisterFunctions routines.
    - Blue Screen of Death (BSOD) is now standard in processor routines.
- Renesas
  - Added more MTU based PWM drivers.

- Library Changes
  - uEZ System
    - uEZGPIO system added to make manipulating IO pins as easy as a uEZGPIOSet(pin) function call. No more HAL_GPIOPort ** references required.
    - uEZTimer system added to access Timer device drivers. Can now easily setup a continuous or a one shot hardware timer with optional function call.
    - uEZ File UEZFileSystemMount/UEZFileSystemUnmount bug found and fixed by forum member pjanco.
    - uEZNetworkConnect added to make connecting to a wired or wireless network easier.
    - UEZNetworkResolveAddress() added to convert strings into IP address through the DNS system (if available).
    - uEZTimeDate routines added to support many different calculations on time and date structures (duration, day of week, previous/next day/week, parse, etc.)
    - uEZAccelerometer routines added for easy accelerometer usage.
    - uEZFileMakeDirectory added to file system and tested for FATFS.
    - ResourceCache system added for finding binary objects loaded into memory as one blob. The driver system allows for the cache to be stored in multiple places. The DirectAccess version assumes the whole cache is accessible directly in the memory space.
    - uEZRandom added to provide pseudo random numbers.
    - HALInterfaceFind will now cause a UEZFailureMsg when not found on debug versions.
    - UEZDeviceTableFind will now cause a UEZFailureMsg("Device not found!") when devices not found in debug versions.
    - Initial HAL workspaces are now filled with 0xCC bytes.
    - UEZBSPFatalError renamed to UEZBSP_FatalError.
    - uEZINI now allows values to be in hexadecimal when prefixed with '$' or '0x'.
    - UEZSystemInit now calls UEZGPIOReset() before starting the RTOS. UEZGPIOReset can register all the GPIO ports needed in the system.
  - BasicWeb
    - Moved to under Library/Network
    - Changed to use the uEZNetwork system instead of lwIP system.
  - C Library of Common functions
    - Added a generic library to store missing functions such as itoa() and ltoa() for better portability.
    - Alternative version of printf added.
  - FATFS File System
    - Fixed a special case where CTRL_SYNC would cause disk_ioctl to return an error. SYNC command now always works.
    - FS_FATFS_Rename added.
  - lwIP

        - SetupEMAC now properly moved into uEZ lwIP code. No longer is in each platform project.
        - MAC address is now retrieved with callback routine NVSettingsGetMACAddress.
    - MemoryTest
        - Memory test code used in demos moved into common library.
    - MIDI
        - MIDI Player defaults to "Speaker" tone generator instead of "Piezo" (Name "Piezo" is now deprecated)
    - Segger emWin
        - Trial and NXP specific libraries added to uEZ release.
        - Multi-buffer mode is now the default mode of operation.
    - SimpleUI
        - SUICallbackRGBConvert added to allow SimpleUI drawing routines to be placed in a library and still draw to any 16-bit graphics display.
        - SUISetSettings added to allow configuration of SimpleUI library. Currently only configures single or double size icons. Default is single sized.
    - StreamIO StdInOut
        - Common library for doing standard input output using putchar, printf, getchar, etc. has been created in /Library/StreamIO/StdInOut. All compiler differences are also placed here. By default, stdin and stdout is linked to the Stream device "Console".
    - SWIM
        - Fixed the definition of LIGHTBLUE
        - Fixed the calculation of the virtual y height when setting a title.
- RTOS Changes:
    - FreeRTOSConfig.h and SaveRTOSConfig.h has been moved into their respective folders.
    - uEZTaskSuspend and uEZTaskResume added.
    - _isr_uEZTaskContextSwitch added to allow interrupt service routines to request context switches so blocking processes can be run immediately instead of waiting for a pre-emptive task switch.
    - portEND_SWITCHING_ISR added to NXP LPC23xx and Renesas RX62N implementation.
    - Removed port.c and portISR.c from RTOS/FreeRTOS directory. No longer used.
- Utilities:
    - Removed PinsToH and Pins1788ToH utilities. No longer need with new Require system.
- Documentation Changes:
    - Doxygen configuration files added.

uEZ v1.12 – 10/18/2011

- uEZ General Changes
  - Doxygen added to uEZ System functions and available online at http://www.teamfdi.com/uez/docs
- Application Changes
  - Added MIDI Player demo and graphics
  - Brightness control demo has new light sensor option
  - Added Network Settings app
- Build Changes
  - Added DK-1788-43WQH
  - Added DK-RX62N-43WQT
  - Added DK-RX62N-47WQT
- HAL and Device Driver Changes
  - General/API
    - PWM HAL API now has SetMatchCallback feature (supported on RX62N only)
    - PWM Device API also has SetMatchCallback
  - Device Driver Changes
    - Backlight
      - Use 300 Hz timing for backlight of Inteltronic LMIX0560NTN53V1
    - LCD
      - Added Tianma TM047NBH01 driver
      - Added Tianma TM043NBH02 driver
      - Added Hitachi TX11D06VM2APA driver
    - Flash for Renesas RX62N
      - Fixed semaphore release issues with errors.
    - PWM
      - SetMatchCallback added to API to support interrupts on matches
    - Touchscreen
      - Added DirectDrive Touchscreen driver
    - VFD
      - Removed all VFD drivers. Use CharDisplay drivers instead.
- Platform Changes
  - DK -TS-RX62N Kits
    - Speaker now software controlled PWM driver instead of direct control.
    - Backlight fixed to work with SOMDIMM-RX62N Rev 2 hardware.
    - Renamed device Piezo to Speaker.
- Processor Changes
  - Renesas
    - PWM split into MTU based PWMs and TMR based PWMs.
    - Fixed allocation of workspaces for SCI5 and SCI6.
    - SPI performance on 8-bit transfers optimized for better throughput.
    - SPI pin configuration fixed (was using wrong –A, -B configuration).
    - RX62N Interrupts now use the whole 15 levels instead of 7.

        - Direct Drive configuration is now platform specific instead of processor specific.
        - Added Direct Drive LCD configuration files for:
            - Tianma TM035KBH02
            - Tianma TM047NBH01
            - Toshiba LTA057A3347F
- Library Changes
    - uEZ System
        - uEZ File
            - Fixed file handle allocation error when using NO_DYNAMIC_MEMORY_ALLOC.
            - Fixed semaphore error on failed FileOpen.
    - Audio
        - MIDI Library added.
- RTOS Changes:
    - RX62N FreeRTOS
        - Fixed FreeRTOS interrupt disabling issue that caused a rare corruption of the task stacks when doing context switching.
        - Corrected FreeRTOS priority level for non-RTOS interrupts.

uEZ v1.11 – 9/1/2011

- uEZ General Changes
  - .H files added for all HAL and Device drivers to help clean-up and simplify HALInit and PlatformInit files.
  - I2C Slave support added (LPC1756)
- Application Changes
  - Test console sdded to LPC1788 uEZ GUI Demos (via RS232)
- HAL and Device Driver Changes
  - General/API
    - Added USB HID Interface
    - Added CRC HAL driver interface
    - Added I2C Slave callbacks to I2C HAL interface
    - USBHost drivers now properly unload CPU when waiting for interrupt.
  - Device Driver Changes
    - USB Host now supports Interrupt endpoints for USB HID communications (LPC1788 and LPC2478, FS only)
    - Added Inteltronic LMIX0560NTN53V1 LCD driver
    - Freescale MMA7455 Accelerometer put back to more dependable 2G setting.
    - Redpine RS9110 RS22 network driver now supports firmware upgrading.
- Platform Changes
  - All platform files have been updated to use the new system wide driver .H files.
- Processor Changes
  - LPC1756
    - Added I2C Slave support
    - Removed erroneous EMAC driver (not available on LPC1756)
  - LPC1788
    - Added CRC HAL driver
    - Fixed LPC1788 SSP routines to allow 0 in MISO/MOSI data parameters.
  - LPC2478
    - Renamed DAC driver
    - Moved prebuilt USBHost object files from root to subfolders based on compiler
- Library Changes
  - uEZ System
    - Added uEZHID for easy USB HID identification and communication
  - Graphics
    - SWIM
      - Added font winfreesystem14x16_subset to SWIM library (font only contains characters 0x20 – 0x7F)

        - emWin
            - Upgraded emWin from v5.10 to v5.12
- RTOS Changes:
    - SafeRTOS
        - Changed priority level for Cortex-M3 to allow interrupts to make RTOS calls.

uEZ v1.10 – 7/29/2011

- uEZ General Changes
  - Renesas RDK RX62N development board support added.
  - Adding more .h files to make drivers better defined and platform files easier to setup.
- Application Changes
  - Added new Renesas RDK RX62N Demo project. Still needs TCP/IP Stack and USB.
- HAL and Device Driver Changes
  - General/API
    - Added improved Handles checking
  - Device Driver Changes
    - Added Button Bank driver that uses External Interrupts as the source of the button states.
    - Added External Interrupt Device driver that interfaces between HAL and device driver layer.
    - Added generic GPIO based LED Bank device driver. Now one bank can handle any list of GPIO's.
    - Added Analog Devices ADT7420 Temperature Sensor.
    - Added Analog Devices ADXL345 Accelerometer.
- Platform Changes
  - Added Renesas RDK RX62N development board (YRDKRX62N)
  - Fixed the initialization of the console variable so it defaults to 0/NULL.
- Processor Changes
  - RX62N Driver Changes/Additions
    - Added External Interrupts driver
    - Fixed EMAC driver to allow configuration of National DSP83640
    - Interrupt handler now goes to InterruptFatalError on unhandled interrupts.
    - Exceptions now go to UEZBSPFatalError when unhandled.
- Library Changes
  - uEZ System
    - Added uEZButton routines for easy access to Button Banks.
    - Added compile option for extensive handle checking.
  - Graphics
    - Added Glyph driver, primarily to support the RDK's Okaya LCD.
  - Network
    - Fixed link between interrupt driven lwIP and EMAC driver (goes through a proper API now)
    - Fixed HTTP Server to connect to a passed in parameter instead of being hardcoded. Also will use .ini file if available.
- RTOS Changes:
  - Support for Keil compilers and the LPC24788 is now working again.

uEZ v1.09 – 7/8/2011

- uEZ General Changes
  - System wide fix of misspelling of OSCILLATOR as OSCILATTOR.
  - Improved Exception Handling now puts blink code on status LED.
  - FreeRTOS upgraded to version 7.0
  - Touchscreen Sensitivity adjusted and better filtering
- Application Changes
  - TestWebServer Demo using new uEZ Network API (based on BasicWeb)
  - Remote Slideshow and Web Server Demo using Redpine RS22 Driver (see project DK-57TS-LPC2478/RemoteSlideshowDemo)
- HAL and Device Driver Changes
  - General/API
    - New uEZ Network API Added
  - Device Driver Changes
    - Accelerometer tweaks to Freescale MMA7455 and ST LIS3LV02DQ drivers to improve readings
    - Moved PCA9551 Button driver to correct directory.
    - Network
      - Added new uEZ Network API driver on top of lwIP TCP/IP Stack.
      - Redpine Signals RS22 Network Driver (object code only for Rowley Crossworks)
    - Touchscreen
      - MicroChip AR1020 Touchscreen Driver sensitivity changed from default values.
      - Generic Four Wire Touchscreen Driver properly filters out incorrect presses.
      - Sensitivity of screens adjusted (Seiko Touchscreen is the only one affected).
- Platform Changes
  - CARRIER_LPC1788 Platform created for Rev 2 and Rev 4 Hardware
- Processor Changes
  - LPC1788
    - Fixed Watchdog crashing bug due to improper configuration
  - LPC2478
    - EMAC MII/RMII Compile Option added
    - Improved SPI to allow easy 0xFF send / 0xFF receive using null pointers.
- Library Changes
  - uEZ System
    - uEZINI system added to allow easy .INI file access.
    - Rename added to File System API
- RTOS Changes:
  - FreeRTOS v7.0 added (with Timer Functions enabled)
  - Memory and Handle failures now go to hook routines.

uEZ v1.08 – 4/21/2011

- uEZ General Changes
  - SafeRTOS support added (but not included in the Open Source release). If you need SafeRTOS for your product, contact sales@teamfdi.com for licensing information. FreeRTOS is still the default OS.
  - Improved IAR Compiler support (more builds now clean in IAR)
    - External NOR flash driver added to UEZ GUI LPC2478 and UEZ GUI LPC1788 builds. Graphic images moved to NOR flash.
  - I2S Audio added with WAV file playback
  - Redpine RS9110_N_11_22 Wireless Network demo added. Still needs work for stability. uEZ Network API added to support both wireless and wired connections.
  - Segger emWin GUI trial version and demos added. If you need Segger emWin for your application, a license is required. Contact sales@teamfdi.com for more details.
- Build File Changes
  - Updated all build files with the latest changes uEZ v1.08 changes
  - Added DK-35TS-LPC2478
  - Added NOR flash drivers to uEZ GUI LPC1788 and LPC2478 builds
- HAL and Device Driver Changes
  - General/API
    - Audio Codec API added
    - I2S API added
    - Network API added (for both wired and wireless)
  - Device Driver Changes
    - New Device Drivers
      - RS485 Half Duplex Driver added
      - Watchdog Generic Driver added
    - Audio Codec Wolfson WM8731
      - Fixed output settings (was not correctly sending 16-bit samples)
    - Generic I2S driver added
    - Sharp LQ043T1DG01 4.3" LCD
      - Added feature to flip X & Y via SPI commands. Does not work on all Sharp LQ043 versions.
    - MassStorage SDCard driver now has configurable SPI interface speed.
    - Redpine RS9110_N_11_22 Wireless Network added. Still needs work for stability.
- Platform Changes
  - SystemInit() is now the standard platform startup function before main() is called.
  - UEZBSPFatalError added to all platforms to handle any unhandled interrupt or exception errors.
  - FDI CARRIER_2478 Added.

        - Supports Rev 2 and Rev 4 hardware.
        - I2S hardware support
        - Redpine wireless support
    - UEZ GUI 7.0 LPC1788 & LPC2478
        - IAR External NOR Flash driver added. Graphic images to NOR flash
- Processor Changes
    - LPC1768
        - Added LPC1768 Watchdog driver
        - LPC1768 Interrupt driver no longer enables interrupts when registered. InterruptEnable() must be called afterward.
    - LPC1788
        - Improved handling of exceptions and interrupts. Failures will now be caught in UEZBSPFatalError() with a blink code.
    - LPC2478
        - I2S HAL Driver added.
- Library Changes
    - New Libraries
        - Segger emWin GUI trial added
    - HTTP Server
        - Changed to support either new uEZ Network API or the older lwIP netconn API. Will be removing netconn API in the near future for uEZ Network API.
    - uEZ System
        - uEZEINT added for easy external interrupt support.
        - uEZFileSystemInit() added to file system. This routine MUST be called in the platform file.
        - uEZStackMemAlloc() added for allocation of just task stacks. Needed for protected memory architectures.
        - uEZPWM routines added.
        - uEZNetwork added as part of uEZ Network API addition
- RTOS Changes:
    - Added compile options to change between different RTOS's.
    - Increased default number of uEZ handles from 100 to 150.
    - SafeRTOS stubs added. FreeRTOS is still the default.
        - Stack sizes have been changed to support SafeRTOS.
        - NO_DYNAMIC_MEMORY_ALLOC option added to avoid alloc/free usage. This avoids any heap fragmentation problems.
- Demo Changes:
    - emWin GUI trial demo added to LPC2478 and LPC1788 demos
    - Newer high resolution icons added to FDI Demos on 7.0" displays. These are licensed icons from professional-icons.com. Contact sales@teamfdi.com for more information.
    - Slideshow demo now allows playing WAV files on I2S enabled hardware
    - Demos can now use static memory instead of dynamic memory

uEZ v1.07 -- 3/7/2011

- uEZ General Changes
  - Reorganization of DK-TS-KIT build files. Builds are now by kit type first, compiler second.
  - Any exception or interrupt error in the system is now sent to UEZBSPFatalError for proper handling
- HAL and Device Driver Changes
  - General/API
    - File System Drivers
      - Seek and Tell added to change file position
    - I2S API added
    - AudioCodec API added
    - Watchdog API added
  - Device Driver Changes
    - New Device Drivers
      - RS485 Half Duplex Driver added
      - Watchdog Generic Driver added
    - Audio Amp NXP 8551T
      - Fixed problem with proper setting when setting the volume more than once.
    - EEPROM:
      - Support of larger page size EEPROMs (128 byte pages)
    - LCD
      - Seiko 70WVW2T 7.0" LCD code can now be used without a power control pin.
    - SDCard
      - SPI SDCard driver now properly returns a not found error if the SDCard is indeed not found when reading size information.
    - USB Device Driver
      - Status change function is now given a default value of 0 to avoid pointer to function errors.
    - USB Mass Storage
      - Because some USB flash drives take longer to start up, the mass storage driver now waits even longer on drive initialization (up to 10 seconds).
- Platform Changes
  - UEZBSPFatalError added to all platforms to handle any unhandled interrupt or exception errors.
  - UEZ GUI 7.0 LPC2478
    - Compile option added to allow USB Port to be USB Device or USB Host.
    - Rev 2 hardware has touch screen lines properly swapped.
- Processor Changes
  - LPC1788

        - Added 1788 Watchdog
    - LPC2478
        - Redirect all unallocated interrupt vectors to InterruptFatalError() to catch any unregistered interrupts.
        - Redirect all exception handlers to UEZBSPFatalError with blink code.
- Library Changes
    - New Libraries
        - IntelHexParser added for parsing intel hex strings into binary.
    - FDICmd library added.
        - Provides a link for entering commands and getting responses to any uEZ Stream device.
        - Usually connected to "Console" or a serial port.
    - FATFS Library
        - Added seek and tell commands to go to byte locations in a file.
    - FDI SimpleUI
        - Provide compile option to turn on/off double sized icons
    - lwIP TCP/IP Library
        - MAC address is now retrieved through required callback routine NVSettingsGetMACAddress() instead of hard coded compile time option.
    - SWIM Library
        - Added 32-point font 'droidsansr32'
    - UEZ System
        - UEZFileSeekPostion and UEZFileTellPosition added
        - UEZToneGeneratorPlayToneContinuous added
    - USB Device
        - "GenBulk D" driver has been renamed into the "FDI Serial Bulk Driver".
            - Provides an alternative method for reading/writing data.
            - Example PC code provided through the GenBulkConsole.
- RTOS Changes:
    - No changes.
- Demo Changes:
    - New professional-icons.com icons added. Licensing is required to use these icons - see licenses for details. Open source icons still provided.
    - Generic FDI Demo Code
        - AppMenu system can now be interrupted to go into test mode.
        - Calibration uses wider coordinates for better calibration.
        - Improved slideshow queuing to avoid occasional pauses.

uEZ v1.06 -- 12/2/2010

- uEZ General Changes
  - NEW uEZ® License (headers changed and license.txt file added)
  - NXP LPC1788 Processor for FDI uEZ GUI added
- HAL and Device Driver Changes
  - General/API
    - UEZ_ERROR_BUSY type added.
    - SSP driver type is being removed and made into a SPI driver.
    - HALInterfaceFind now returns a null pointer if not found in addition to the normal error.
  - Device Driver Changes
    - New Device Drivers
      - AudioAmp Driver type added
      - AudioAmp NXP TDA8551T added
    - EEPROM
      - 16 bit I2C EEPROM Driver now supports devices larger than 64KB .
    - Flash
      - NOR Flash S29GL064N90 16-bit driver added
    - GPIO
      - GPIO Control feature added to control other settings (e.g. input buffer enabled/disabled).
    - LCD
      - Added NEC NL6448BC18-03F VGA LCD
      - Added Seiko 43WQW1T QVGA LCD
      - Added Seiko 70WVW2T WVGA LCD
    - SPI
      - TransferInOutBytes added to SPI interface.
    - Mass Storage
      - Converted SDCard code from SSP to SPI driver interfacing.
      - SDCard SDHC support added (4 GB+ cards)
    - RTC
      - NXP PCF2129 Added
    - Touchscreen
      - Improved interrupt driven Touchscreen
    - USB Device
      - Added LPC1788 USB Device driver
- Platform Changes
  - Added FDI Carrier Rev 3 Platform for LPC2478
  - Added FDI Carrier Rev 3 Platform for LPC1788
  - Added uEZGUI_1788_70WVE
  - Added uEZGUI_2478_70WVE
- Processor Changes

    - NXP LPC1788 added (under IAR and or Keil uVision 4 compiler)
        - Interrupts
        - GPIO Driver
        - ADC Driver
        - Battery RAM Driver
        - GPDMA Driver
        - I2C Driver
        - LCD Controller Driver
        - PWM Driver
        - RTC Driver
        - Serial Driver
        - SPI Driver (SSP)
        - Timers Driver
        - USB Device Controller
        - USB Host Driver
- Library Changes
    - BasicWeb
        - Fixed a lock up caused when network is not running but BasicWeb started
- RTOS Changes:
    - Fixed problem of semaphore handle being returned when really out of memory
- Demo Changes:
    - uEZ GUI 4.3 Blinking LCD added
    - DKTSKIT Demo for LPC1788 Added
    - DK-TS-KIT and DK-VTS-KIT uEZDemo now use FDI/UEZDemoCommon code.
    - uEZGUI 7.0” LPC2478 Demo Added
    - uEZGUI 7.0” LPC1788 Demo Added
    - Generic FDI Demo Code
        - Accelerometer now calibrates zero at start
        - AppMenu system now calls Idle function
        - Improve Slideshow demo with hour glass / feedback and better caching.

uEZ v1.05 -- 8/23/2010

- HAL and Device Driver Changes
    - General/API
        - SSP and SPI naming convention for in/out buffers changed to avoid confusion
        - LCD driver now supports feature to read back current backlight level
        - Flash Memory read/write subsystem added.
    - LPC2478 Drivers
        - BatteryRAM HAL driver added

        - DAC HAL driver added
        - GPDMA HAL driver added
        - SMSC LAN8720 Ethernet support added
        - Improved LCD Controller initialization (PINSEL10 and PINSEL11 registers)
        - PWM driver bug fixed (changing master counter was not resetting counter)
        - GPIO ports now support interrupts per port on falling or rising edges.
    - Device Driver Changes
        - LCD device drivers updated to support SPI and LCD API changes (see above).
        - Sharp LQ043T1DG01 LCD device driver added
        - Sharp LQ043T3DG02 LCD device driver added
        - EEPROM definitions for M24LC256 and M24LC64 added
        - EEPROM Driver for 16-bit devices added
        - ST LIS3LV02DQ Accelerometer driver added
        - SDCard Mass Storage Device Driver over SSP added.
        - Generic RTC Device Driver now handles Validate and SetClockOutHz commands correctly.
        - Generic SSP Device Driver correctly handles Start, Stop, and TransferInOut commands.
        - 4-Wire Touchscreen now uses GPIO interrupts.
- Platform Changes
    - New FDI uEZ GUI Platform added
- Processor Changes
- Library Changes
    - SWIM Library
        - Fonts droidsansr14 and droidsans76 added
        - SWIM fonts now use less memory.
        - SWIM fix to transparency spelling and flag usage.
    - USB Device
        - USB MassStorage Device driver linked to SDCard
        - USB MassStorage Device driver linked to RAM (aka RAM Drive)
    - FDI SimpleUI Added
        - Common Choices code and Drawing routines moved into reusable library
        - Choices now have a flag to repeat or not. Default is off in most places.
    - FATFS
        - Files open for append now append instead of clipping.
    - lwIP
        - Remove issue with lwIP warnings in Rowley Crossworks 2.0 compilers
- Compiler Changes:

    - Keil uVision 4.0 support added
- RTOS Changes:
    - Keil uVision 4.0 support added for LPC1768 (but not LPC2478)
- Demo changes:
    - uEZ GUI Demo added (Rowley Crossworks and IAR 5.4)
    - Used uEZDemoCommon with UEZ GUI Demo v1.04
- Subversion Revisions: 54-??
- Checksum generated with DK-TS-KIT UEZDemo v?.?? for QVGA display using Rowley Crossworks 2.0.

uEZ v1.04 -- 4/1/2010, Checksum 0x0436DF77

- uEZ Changes
    - Added LPC1768 Support
    - Added InterruptIsRegistered() command
    - General clean up of many cases of unused variables or braces in structures found by the GCC compiler used by CodeRed.
    - Added more powerful HTTPServer with inline macro parsing.
    - Fixed a bug in UEZFileFindStart that would crash on failure.
    - Added more checks into UEZ link with RTOS to ensure parameters are correct.
    - Added several UEZ System functions:
        - UEZADC
        - UEZCharDisplay
        - UEZKeypad
        - UEZLEDBank
        - UEZTemperature
        - UEZToneGenerator
    - Improved build system to support multiple build types and specifically Code Red support. Fixed various compiler setup issues (Rowley and IAR).
- HAL and Device Driver Changes:
    - Drivers added:
        - CharDisplay Driver for Lumex SO2004DSR
        - External Interrupt Driver for LPC1768
        - Keypad driver for NXP I2C PCA9555
        - SA56004X Temperature Driver
        - PWM driver (Generic)
        - Tone Generator using Generic PWM
    - Processor Support Added:
        - LPC1768
    - Fixed problem with Serial driver locking up when full.
- RTOS Changes:
    - FreeRTOS upgraded to 6.0.1
    - FreeRTOS LPC1768 support added.
- Demo changes:

    - IRDDemo v1.00 added for NXP IRD v1.00 base board.
- Subversion Revisions: 25-53
- Checksum generated with DK-TS-KIT UEZDemo v1.06 for QVGA display using Rowley Crossworks 2.0.

uEZ v1.03 -- 7/23/2009, Checksum 0x04272674

- uEZ Changes
    - Made the LCD configuration code simpler to setup and change between displays. One single configuration in Config_App.h.
- HAL and Device Driver Changes:
    - Backlight driver now correctly handles calculations for very low frequencies (<100 Hz).
    - Added OKAYA RV640480T VGA LCD driver.
    - Added OKAYA RV320240T QVGA I15 mode
    - Added OKAYA RH320240T QVGA LCD driver (with SPI configuration).
    - LCD Drivers now have a list of supported mode interfaces.
    - Fixed calculation of LCD dot clock to round down instead of round up.
- Demo changes (AppDemo v1.05):
    - Demo resized to work on 640x480 (VGA) displays as well as 320x240 (QVGA) displays.
    - Fixed bug that caused long delay when leaving Settings menu.
    - Improved speed of drawing icons and title screen.
    - Draw app now correctly reports if trying to load from USB Drive or SD Card and if either are not plugged in or missing a file.go
    - Slideshow app now correctly reports if trying to load from USB Drive or SD Card and if either are not plugged in or missing a file.
    - Increased the number of QVGA slides up to 38.
- Size (Rowley CrossWorks and ARM-57TS-KIT):
    - Internal ROM: 0x6C9FC of 0x80000 (Debug), 0x4D308 of 0x80000 (Release)
    - Internal RAM: 0xD66C of 0x10000 (Heap size 30,000 bytes)
    - SDRAM: Frame buffer
- Subversion Revisions: 22-24
- Checksums (Rowley CrossWorks)
    - ARM-57TS-KIT (Toshiba LTA057A347F): 0x04272674
    - ARM-35TS-KIT (Okaya RH320240T): 0x042571C3
    - ARM-57VTS-KIT (Okaya RV640480T): 0x0426B213

uEZ v1.02 -- 5/25/2009, Checksum 0x04299036

- uEZ Changes
    - Added support for IAR compiler v5.20
    - Added RTC device functions for validating clock and setting toggle line rate.

    - Platforms now create an initial task called uEZPlatformStartup that does a second phase of initialization before starting the main() task.
    - Many minor changes to improve overall code quality (removing unused variables and functions, fixing mixed types, returning proper error codes, correct headers, etc.)
- HAL and Device Driver Changes:
    - Fixed LPC2478_LCDController to correctly calculate the proper dot clock rate. It now properly rounds to a slower setting instead of a faster setting.
    - LPC2478 PWM is now correctly stopped before it is changed since otherwise it does not change.
- Demo changes:
    - Added feature in Time & Temperature to set the External RTC Date & Time.
    - Functional Test Loopback mode added. One unit can now be a slave and the other a host to allow FCT without the need for a separate CAN Demo board.
    - Functional test now only tests 8 MB of actual SDRAM instead of trying to size the memory and sometimes generating phantom results.
- Size (Rowley CrossWorks):
    - Internal ROM: 0x6C604 of 0x80000 (Debug), 0x4D198 of 0x80000 (Release)
    - Internal RAM: 0xD650 of 0x10000 (Heap size 30,000 bytes)
    - SDRAM: Frame buffer
    - Subversion Revisions: 14-21

uEZ v1.01 -- 4/10/2009, Checksum 0x0579BBD1

- uEZ Changes
    - EMAC is setup inside of TCP/IP thread instead of main thread to avoid a 7 second startup time delay. Now the EMAC is setup in parallel with the title screen to decrease the delay at start up.
- Demo changes:
    - Added black to the list of colors in the drawing demo to support an erase function. Black is drawn with a 3x3 brush instead of 1x1 to help with clearing issues on the erase function.
- HAL and Device Driver Changes:
    - USB Mass Storage device driver has been fixed to correctly report the drive as not initialized when no drive has been plugged in. Otherwise, the FATFS would make calls to Read with illegal data and cause a crash.
- Library Changes:
    - Turned off memp overflow checks in lwIP library because having it on will use the __FILE__ C preprocessor keyword and cause the checksum to change from compile to compile, making version tracking impossible.
- Size:
    - Internal ROM: 0x67ACC of 0x80000 (Debug), 0x48E70 of 0x80000 (Release)
    - Internal RAM: 0xD590 of 0x10000 (Heap size 30,000 bytes)

- SDRAM: Frame buffer
- Subversion Revision: 14

uEZ v1.00 -- 3/27/2009, Checksum 0x0577B471

- uEZ Changes
  - Fixed version information to not change between compiles to ensure that the checksum does not change when recompiled.
  - Changing CPU speed is now handled in one place.
- ARM-TS-KIT (IRD2.0) changes
  - Cleaned up SDRAM configuration to be a single setting.
- Demo changes:
  - Split out many of the code segments in main.c into separate files for clarity and organization.
  - Non-volatile EEPROM memory now stores the MAC address, IP Addr, IP Mask, and IP Gateway.
  - Simplified Time & Temperature demo to only show Temperature for now.
- HAL and Device Driver Changes:
  - Mass storage devices can now have logical unit addresses.
  - Fixed a problem with touch screen not correctly deleting event queue when closing.
  - Separated USB Host device driver from USB Host HAL driver.
  - SDCard driver has increased SPI rate for faster transfers.
  - USB Host device driver now correctly monitors enumeration and determines an appropriate registered device driver to call. This includes proper detection of a USB device plugged in and removed.
  - Fixed a problem where the EMAC would hang endlessly at boot.
- Size:
  - Internal ROM: 0x67E98 of 0x80000 (Debug), 0x491E8 of 0x80000 (Release)
  - Internal RAM: 0xEC25 of 0x10000 (Heap size 30,000 bytes)
  - SDRAM: Frame buffer

uEZ v0.11 3/11/2009, Checksum 0x043DC2C8 (BP)

- uEZ Changes
  - Improved calibration to ignore extremely bad results.
  - Touch screen routines now wait until screen is not being touched before sending out data.
  - Crashing bug fixed when closing a file.
  - Fixed FATFS not working with some USB flash drives due to stall on unsupported command. Now correctly continues.
  - Random crashing bug fixed. Problem in RTOS and EMAC interrupt routine.
  - Lowered memory requirements in several systems.
  - Code for checking memory integrity added into optional heap source.

    - Various small changes to memory allocation routines to track memory usage (heap blocks, HAL workspaces, and device workspaces).
    - Turned on compiler option to leave out uncalled/unused code and data.
- ARM-TS-KIT (IRD2.0) changes
    - Added power detect mode that lowers the LCD backlight intensity to save power when using a power source other than the AC supply.
    - Removed RTOS_HEAP. Everything is now running (except frame buffer) from internal memory.
- Demo changes:
    - Slideshow will load from USB drive first, SDCard second. If neither available, will use last images loaded.
    - Pressing and holding SW1 and SW4 while powering up will force the unit into calibration and allow touch screen recovery if config data is corrupted.
    - Added non-volatile settings section and routines.
        - Checksum added to non-volatile settings saved in EEPROM.
        - Settings for IP address, IP gateway, IP netmask, MAC address added.
    - Turned off telnet echo.
    - Lowered stack sizes to lower memory usage.
- HAL and Device Driver Changes:
    - Created generic device drivers for SPI, SSP, I2C and ADC and removed specific LPC2478 device drivers for these functions.
    - Converted SDCard's CRC16 table from a generated table into a hardcoded ROM based table to save memory.
- Platforms Changes:
    - Removed Customer specific and EA_OEM platforms
- Library Changes:
    - FATFS now has option for number of mass storage devices. Changed from 4 to 2 to save memory.
- Size:
    - Internal ROM: 0x6731C of 0x80000 (Debug), 0x48384 of 0x80000 (Release)
    - Internal RAM: 0xEFF4 of 0x10000 (Heap size 30,000 bytes)
    - SDRAM: Frame buffer

uEZ v0.10 -- 2/24/2009, Checksum 0x04F3587A (BP)

- Various Changes
    - Main clock speed changed from 57.6 MHz to 72 MHz
    - GPIO now has SetMux command to change functionality of pin
    - Fixed incorrect mapping of LPC2478 I2C Bus Device Driver (I2C1 and I2C2 fixed)
    - Added 15-bit color interface to LCDs and SWIM library
    - Increased lwIP memory space (now that RTOS moved to external memory)
    - Changed Generic HID driver to capture CAPS Lock key for Functional Test
    - Renamed LPC2478's RTC to "IRTC" for internal real time clock.

- Added following device drivers:
    - SDCard (via SPI) Mass Storage device driver
    - USB Mass Storage device driver
    - Generic RTC device driver
    - NXP PCF8563 RTC device driver
    - Generic I2C EEPROM device driver
    - NXP PCA9551 Button Bank device driver
    - NXP PCA9551 LED Bank device driver
    - Generic PWM Backlight device driver
    - Bosch BMA150 Accelerometer device driver
    - NXP LM75A Temperature device driver
    - Generic 4-Wire Touch resistive touch screen device driver
- Added following libraries:
    - FATFS (and linked to Mass Storage drivers)
- Added following platforms:
    - ARM CARRIER (for ARM-57TS-KIT)
- Demo changes:
    - RTOS Heap created in SDRAM instead of internal LPC2478 RAM
    - Added Accelerometer demo
    - Added Time & Temperature demo
    - Added Functional test (including simple CAN test code)
    - Slideshow now can show 1 to 9 slides instead of just 6
    - Many extra debugging tasks removed
    - Added PWM for Audio output
    - Added Settings menu item that goes to second menu
    - Added more graphics to support above changes
    - Pulled out common code being used for push buttons in one place
- Size:
    - Internal ROM: 0x7B808 of 0x80000 (Debug), 0x 53f60 of 0x80000 (Release)
    - Internal RAM: 0xFC24 of 0x10000 (Heap size 25,250)
    - SDRAM: Frame buffer and RTOS_HEAP

uEZ v0.09 -- 2/2/2009, Checksum 0x00F0B468

- Improved File System hierarchy
    - Improved File System FAT system.
    - Added LPC24xx USB Host driver.
    - Added Mass Storage device for USB Flash Drives.
    - USB Flash drive can now be unplugged and plugged back in without problem.
- Network improvements to lwIP
    - Ethernet MAC now interrupt driven instead of polled for better performance.
    - Ethernet EMAC now is configured in auto-crossover mode (Auto MDIX) to avoid requiring a hub.
    - Uses smaller blocks when receiving packets for better memory management.

- Backlight Driver added and incorporated into existing LCD drivers.
- LCD backlight is turned off as soon as possible now. LCD driver no longer automatically turns on backlight.
- Fixed EEPROM page size problem to allow smaller EEPROM reading/writing.
- Added LPC24xx RTC processor clock driver.
- Size:
    - Image size: 149,684 bytes
    - Internal ROM: 0x43A30 of 0x80000 (Debug), 0x34858 of 0x80000 (Release)
    - Internal RAM: 0xF0CC of 0x10000 (Heap size 29,000)
    - SDRAM: Only frame buffer

uEZ v0.08 -- 1/9/2009, Checksum 0x00D73536

- Added USB Device support
    - Both USB Host and Device can work simultaneously (as long as host is initialized first).
    - Generic Bulk USB library has been added that communicates with the libusb library.
- 8-bit graphics support added.
    - Default palette is 256 color 3:2:3 RGB format.
    - Writes to memory must be 16-bit values, not single 8-bit values.
    - Example code can be compiled for 8-bit or 16-bit graphics.
    - SetPaletteColor routines added.
- Touch sensitivity setting can now be passed to touch screen driver.
- Toshiba LCD display now correctly has backlight intensity control.
- TCP/IP stack uses less memory by using global memory instead of through the heap.
- File find routines added to enumerate through files in a directory. Currently only supports root directory.
- Demonstration code provided to show all features (slideshow, calibration, telnet, drawing, etc.)
- Size:
    - Image size: 133,960 bytes
    - Internal ROM: 0x3CAA0 of 0x80000 (Debug), 0x30AEC of 0x80000 (Release)
    - Internal RAM: 0xF128 of 0x10000
    - SDRAM: Only frame buffer

uEZ v0.07 -- 12/19/2008, Checksum ----

- RTOS moved timer from Timer 0 to Timer 1 to allow audio to use Timer 0.
- BSP delay routines moved from Timer 1 to Timer 2.
- Timer HAL driver added
- PWM HAL driver added
- Create Customer specific_XNC library with Audio, Flash (M25P40 support), and I2C routines (for RTC PCF8563 and EEPROM 24C08). Test routines added to main().

- LCD's backlight linked to PWM pin. LCD backlight routines now correctly set the backlight between 256 different levels.
- OKAYA LCD display fixed timing by adding to the front porch timing. This fixed the jitter problem when drawing with grays. Also upped clock to 8.0 MHz (theoretical max is 8.333 Mhz).
- Improved touch screen calibration and readings.
- Added USB Device Stack code (currently disabled due to conflict with USB Host)
- Added USB Device Controller to LPC2478 processor code.
- Added SPI HAL and Device driver
- Added SSP HAL and Device driver. Demonstrated reading/writing M25P40 flash.
- Added I2C HAL and Device driver Demonstrated reading/writing PCF8563 RTC.
- USB Host is now port selectable (port A or B) and corrected for XNC hardware.
- [[Added Virtual Comm USB driver]]
- [[Added SWIM library for graphics]]
- Pin modifications:
    - P3.25 (PIEZO) set to MAT0[0] (was GPIO)
- Size:
    - Internal ROM: 0x32068 of 0x80000 (Debug), 0x???? of 0x80000 (Release)
    - Internal RAM: 0xFD54 of 0x10000
    - SDRAM: Only frame buffer

uEZ v0.06 -- 11/26/2008, Checksum ----

- SRAM - slowed EMCStaticWaitRd timing (delay from chip select to read access) to be 2 clock cycles instead of 1 and fixed display problem.
- Serial driver - better use of hardware FIFO to lower number of interrupts.
- Default output is now referred to as Console
- Program Tag added for boot loader
- TCP/IP stack improvements
    - Use more hardware buffers
    - More internal buffers
    - Turn off debug output
    - Combine outgoing data chains for better buffer usage
- Raised OKAYA dot clock back up to where it should be to match spec.
- Improved performance of touch screen with multiple readings
- Fixed bug in deleting tasks when they exit (it was only deleting the handle, not the task as well).
- Mapped UART1 to Console
- Because the video memory required more than 128K of data and SRAM0 is at 0x80000000 and SRAM1 is at 0x81000000, the frame buffer memory was moved to a base address of 0x80FE0000 using the mirrored memory range of SRAM0 that then rolls over into SRAM1.
- Fixed macros used for setting initial pin configuration.
- Various pin modifications:

    - P0.17 (SEL_RS485) set to pull up and output high to enable 485 output as console
    - P0.18 (SEL_RS485_OPT) set to pull down
    - P2.29 (RGT/LFT_SCAN) set to output low with pull down
    - P2.30 (UP/DN_SCAN) set to output high with pull up
- Correctly initialize Micrel KSZ8041TL
- Size:
    - Internal ROM: 0x32068 of 0x80000
    - Internal RAM: 0xFD54 of 0x10000
    - SDRAM: Only frame buffer

uEZ v0.05 -- 11/21/2008, Checksum ----

- Setup platform configuration for hardware
- Configuration files were changed for more choices
- Fixed touch screen press detection
- Program Tag information added for boot loader
- Updated USB Host FAT16 code with file overwrite feature
- SRAM is now initialized
- Size:
    - Internal ROM: 0x34008 of 0x80000
    - Internal RAM: 0xFA6C of 0x10000
    - SDRAM: Only frame buffer

uEZ v0.04 -- 11/19/2008, Checksum ----

- Renamed uFlex to uEZ to avoid trademark issues.
- Added BSD-style socket functions
- Added USB Host support
- Added TCP/IP support
- Added Raw touch screen support for Embedded Artists board
- Size:
    - Internal ROM: 0x341EC of 0x80000
    - Internal RAM: 0xFA64 of 0x10000
    - SDRAM: Only frame buffer

uFlex v0.03 -- 11/10/2008, Checksum ----

- Added OKAYA LCD display support
- Size:
    - Internal ROM: 0xDD84 of 0x80000
    - Internal RAM: 0x6BA4 of 0x10000
    - SDRAM: Only frame buffer

uFlex v0.02 -- 11/9/2008, Checksum ----

- Added Stream and Serial support.

- Size:
    - Internal ROM: 0xD89C of 0x80000
    - Internal RAM: 0x6B34 of 0x10000
    - SDRAM: Only frame buffer

uFlex v0.01 -- 11/10/2008, Checksum ----

- Initial version with LCD, touch screen, SPI, and SDRAM
- Size:
    - Internal ROM: 0xBAE4 of 0x80000
    - Internal RAM: 0x62A0 of 0x10000
    - SDRAM: Only frame buffer