



Release notes SC12 @CHIP-RTOS

This document lists all modifications, additional features and bugfixes of the @CHIP-RTOS versions since version V1.10 Beta. The release note lists are sorted by our internal change request numbers "CR Number" and by the "Type" of the change request. We defined four different types of change request: "Defect", "Suggestion", "New" and "Checkup". "Checkup" means a necessary verification and possible improvement of a @CHIP-RTOS component. The report fields "Component" and "Category" are describing the affected parts of the @CHIP-RTOS. "Synopsis" and "Description" are used for description of the change request.

- [SC12 @CHIP-RTOS V1.10](#)
- [SC12 @CHIP-RTOS V1.10 Beta](#)

Release notes SC12 @CHIP-RTOS V1.10

Type: Defect, Component: CHIP.INI (2 item(s))

CR Number: 947

Type: Defect

Component: CHIP.INI

Category: Comments

Synopsis: Changed comment behavior

Description: Comments with semicolon are now only possible at start of line, because the semicolon maybe is used in modem or user strings.

Fix: Fixed.

CR Number: 975

Type: Defect

Component: CHIP.INI

Category: Whitespaces

Synopsis: Errors while parsing CHIP.INI

Description: If a line in CHIP.INI has many leading spaces in front of a CHIP.INI item, the entry of the item is not read correct and next lines in CHIP.INI can also get corrupted.

Fix: Fixed

Type: Defect, Component: Filesystem (1 item(s))

CR Number: 976

Type: Defect

Component: Filesystem

Category: Long filenames

Synopsis: Directory summary is not always shown correct.

Description: If a compactflash drive is connected to the IPC@CHIP and the compactflash contains long filenames, the directory summary in bytes is sometimes not shown correct.

Fix: Fixed

Type: Defect, Component: Fossil API (1 item(s))

CR Number: 977

Type: Defect

Component: Fossil API

Category: XON/XOFF

Synopsis: XON/XOFF is sometimes not recognized.

Description: When DMA receive mode, UART bit modes like 7E1, 7O1 and XON/XOFF software flow control is used, the flow control doesn't work correct.

Fix: Fixed

Type: Defect, Component: FTP server (2 item(s))

CR Number: 972

Type: Defect

Component: FTP server

Category: MDTM command

Synopsis: Failures if MDTM is used

Description: It can lead in rare cases to invalid opcode interrupts and/or corrupted memory, if a FTP client sends the MDTM command to the IPC@CHIPs FTP server.

Fix: Fixed

CR Number: 982

Type: Defect

Component: FTP server

Category: File upload

Synopsis: Illegal erase of files at terminated incomplete file uploads

Description: If a ftp session is terminated while uploading a file from the IPC@CHIP, the filesystem deletes this file at the IPC@CHIP.

Fix: Fixed

Type: Defect, Component: HW API (1 item(s))

CR Number: 962

Type: Defect

Component: HW API

Category: Non-volatile data

Synopsis: REMA.BIN is sometimes not created new, if size of non-volatile data increases.

Description: If non-volatile data increases, but stays below next cluster boundary, the file REMA.BIN was not created new and the `init_non_volatile_data()` function returns with an error.

Fix: Fixed

Type: Defect, Component: TCPIP API (1 item(s))

CR Number: 932

Type: Defect

Component: TCPIP API

Category: Function 0x74
Synopsis: Packet statistics
Description: Counting outgoing ICMP packets doesn't work exactly
Fix: Fixed

Type: Defect, Component: Webserver (2 item(s))

CR Number: 957
Type: Defect
Component: Webserver
Category: CGI
Synopsis: Undefined maximum length of CGI pages
Description: The maximum length of pages, produced by a user's CGI function is undefined, because of the usage of non-normalized response buffer pointers inside of the @CHIP-RTOS.
Fix: Fixed. The maximum allowed size of an CGI page is 65519 characters. If the size of page will go beyond this limit, the Webserver returns HTTP errorcode 503. This errorcode is also returned, if memory allocation will fail, inside of the IPC@CHIP webserver.

CR Number: 960
Type: Defect
Component: Webserver
Category: CGI functions 0x07 and 0x08
Synopsis: Argument length
Description: The max. length of arguments (name and value) are not checked. This can lead to a buffer overrun, if CGI function 0x07 is executed, without checking the length of the given arguments of the incoming HTTP-Requests.
Fix: Fixed

Type: Suggestion, Component: BIOSINT API (1 item(s))

CR Number: 72
Type: Suggestion
Component: BIOSINT API
Category: sprintf API call
Synopsis: Add API call for sprintf
Description: Add an API call that makes it possible to use the internal "sprintf" function of the RTOS. This way the user can build smaller executables without the standard library of Borland. The CLIB V2.00 uses this new call for the function "helper_printf".
Fix: Implemented

Type: Suggestion, Component: Docu (1 item(s))

CR Number: 952
Type: Suggestion
Component: Docu
Category: Documentation
Synopsis: RTOS kernel description
Description: For better understanding of the CHIP-RTOS multitasking system, the API documentation must be

complemented.

Fix: Added API documentation by a chapter named "Multitasking with @Chip-RTOS"

Type: Suggestion, Component: SPI (1 item(s))

CR Number: 974

Type: Suggestion

Component: SPI

Category: Additional function

Synopsis: Should add a combined SPI read / write function

Description: Some SPI components require a combined read / write function.

Fix: Implemented

Type: Suggestion, Component: Webserver (1 item(s))

CR Number: 954

Type: Suggestion

Component: Webserver

Category: Mimetypes

Synopsis: Mimetype for *.xsl files

Description: The webserver should return MIMETYPE /text/xml for files with extension *.xsl.

Fix: Implemented

Total: 14

Release notes SC12 @CHIP-RTOS V1.10 Beta

Type: Defect, Component: @CHIP-RTOS (1 item(s))

CR Number: 642

Type: Defect

Component: @CHIP-RTOS

Category: Batch file execution

Synopsis: Batchmode 1

Description: If an user application calls BIOSINT function 15h (allow further batch file execution) at the beginning of the program, the batch file execution hangs. This happens only, if the task which calls the BIOSINT function will be circular waked by a RTOS timer procedure.

Fix: Fixed

Type: Defect, Component: BIOSINT API (1 item(s))

CR Number: 814

Type: Defect

Component: BIOSINT API

Category: FastFindFirst function

Synopsis: FastFindFirst failed

Description: FastFindFirst call leads to unexpected faults (Invalid Opcode, corrupted memory), if the requested

file name not exists.

Fix: Fixed

Type: Defect, Component: CHIP.INI (1 item(s))

CR Number: 584

Type: Defect

Component: CHIP.INI

Category: Chip.ini entries

Synopsis: Search for chip.ini entries may fail

Description: Reading an entry from chip.ini fails, when the name exists also as a substring in two or more sections of the file. Example for a search fault, searching with IniGetString for entry "MODEM":

```
[SECTION1]
MODEM=value1
MODEMCMD=value2
```

```
[SECTION2]
MODEM=value3
```

Fix: Fixed

Type: Defect, Component: Command shell (1 item(s))

CR Number: 578

Type: Defect

Component: Command shell

Category: Batch files

Synopsis: Invalid execution of batch files

Description: If batch files are stored at current filesystem drive, and the invalid command * is executed, the command shell executes wrongly the first found batch file.

Fix: Fixed

Type: Defect, Component: Filesystem (2 item(s))

CR Number: 580

Type: Defect

Component: Filesystem

Category: Write files

Synopsis: Writing files to disk when disk is near full

Description: The last file system cluster sometimes could not be used.
e.g.: If there's only one cluster free (1024 bytes) on drive A: and a file smaller than 1024 bytes should be written, sometimes an error message will be returned, because there's no space left. On the flash a file with size of 0 was created.

Fix: Fixed.

CR Number: 586

Type: Defect

Component: Filesystem

Category: Error detection

Synopsis: Incomplete error detection at file write faults

Description: Faults while writing a file sometimes return without reporting an occurred error.

Fix: Fixed

Type: Defect, Component: Fossil API (4 item(s))

CR Number: 577

Type: Defect

Component: Fossil API

Category: Receive

Synopsis: Framing/Parity Error check

Description: If serial ports are running at interrupt receive mode, we must check the framing and parity error bit inside of the interrupt service functions and removing those bad characters from the serial receive buffer queue.

Fix: Fixed.

CR Number: 615

Type: Defect

Component: Fossil API

Category: Long breaks

Synopsis: Sending continuous long breaks at lower baudrates

Description: Sending continuous long breaks at lower baudrates (equal or smaller 1200 Baud) can lead to Watchdog Reset (Reboot) of the IPC@CHIP

Fix: Fixed

CR Number: 740

Type: Defect

Component: Fossil API

Category: Long/Short Break

Synopsis: Invalid data at fossil receive queue

Description: Serial port used with interrupt receive mode:
After receiving long/short breaks, the invalid data bytes should not be added to the serial receive buffer queues.

Fix: Fixed

CR Number: 770

Type: Defect

Component: Fossil API

Category: Send DMA

Synopsis: Send DMA does not work correct

Description: Send DMA does not work correct when external interrupts (INT5/INT6) are generated.

Fix: Fixed

Type: Defect, Component: HW API (2 item(s))

CR Number: 681
Type: Defect
Component: HW API
Category: Get HW API Function Pointers
Synopsis: Incorrect return value
Description: The return value of the Hardware API function 90h is not correct.
Fix: Fixed

CR Number: 728
Type: Defect
Component: HW API
Category: Saving Retentive Data, Function 0xC1
Synopsis: Saving retentive data
Description: Calling Hardware API function 0xC1 with a size of non-volatile data over 32768 leads to invalid opcode interrupts.
Fix: Fixed

Type: Defect, Component: INT21h (1 item(s))

CR Number: 827
Type: Defect
Component: INT21h
Category: Function 0x36 / Get disk free space
Synopsis: Invalid drive is not correct reported
Description: When INT21h function 0x36 / Get disk free space is called with an invalid drive number, the return value in AX is not correct. (Should be -1)
Fix: Fixed

Type: Defect, Component: PPP Client (2 item(s))

CR Number: 699
Type: Defect
Component: PPP Client
Category: User name and password
Synopsis: User/Password empty string
Description: If user or/and password are an empty string (""), the PPP Client open does not work correct (the task which executes the PPP Client Open will be removed).
Fix: Fixed.

CR Number: 716
Type: Defect
Component: PPP Client
Category: PPP Client Open
Synopsis: PPPClient_Open (TCPIPAPI call 0x41)
Description: If the parameters of this API call are incorrect, the PPPClient_Open call removes the calling task from the system.
Example for incorrect parameters: Auth=1 without username and password parameters.
Fix: Fixed

Type: Defect, Component: RTOS API (1 item(s))

CR Number: 815

Type: Defect

Component: RTOS API

Category: Creating/Deleting tasks

Synopsis: Error when Creating/Deleting tasks

Description: Creating/Deleting high number of tasks in very fast order could lead to a watchdog reset (reboot) or invalid opcode interrupts. Occurs only, if a very high number of tasks (up to 25) are deleted by RTX_DELETE_TASK in very fast order.

Fix: Fixed

Type: Defect, Component: TCPIP API (1 item(s))

CR Number: 829

Type: Defect

Component: TCPIP API

Category: Ping_Open, TCPIP API function 0x75

Synopsis: Sending pings request endless

Description: It is not possible to start an endless loop of ping requests, which is not limited by the user provided parameter "count".

Fix: Fixed.

If the user provided parameter count has value zero, the ping process runs, until PING_CLOSE (API function 0x76) is called by user.

Type: Defect, Component: Telnet Server (3 item(s))

CR Number: 494

Type: Defect

Component: Telnet Server

Category: Local Echo; Binary Mode

Synopsis: Negotiation for the local echo

Description: The negotiation for the local echo doesn't work correct:
The Telnet server has switched on its echo (all received data is echoed back to the client). But on negotiation the server told the client that he does not generate an echo. This causes problems with the Windows XP and the Linux Telnet client.
The "\r\n" sequence was interpreted wrong from Linux Telnet Client.

Fix: Now the Telnet server tells the client at connection startup that it will echo all data.

Now we have also switched the Telnet server into textmode (so "\r\n" is no problem anymore using Linux).

CR Number: 704

Type: Defect

Component: Telnet Server

Category: Linux telnet clients

Synopsis: Detection of backspace key

Description: The Backspace key does not work at telnet sessions with linux telnet clients.

Fix: Fixed

CR Number: 805

Type: Defect

Component: Telnet server

Category: Malfunction

Synopsis: Telnet malfunction

Description: Toggle the Stdio focus key at telnet client sessions with a high frequency can lead to fatal Telnet errors.

Fix: Fixed

Type: Defect, Component: Webserver (5 item(s))

CR Number: 618

Type: Defect

Component: Webserver

Category: Mainpage

Synopsis: A CGI name could not be used as mainpage.

Description: If a CGI name will be set as mainpage, the webserver returns wrongly the standard IPC@CHIP mainpage instead.

Fix: Fixed.

CR Number: 731

Type: Defect

Component: Webserver

Category: HTTP Header

Synopsis: Date at HTTP Response header

Description: At the HTTP Response header the month entry is incorrect.
E.g. instead Wed 5 Dec 2001 is in the header Wed 5 Nov 2001!
If the current month of the RTOS date is January, the returned date of the HTTP response contains garbage data.

Fix: Fixed

CR Number: 807

Type: Defect

Component: Webserver

Category: HTTP Redirect directive

Synopsis: Using HTTP Redirect at CGI functions

Description: Using HTTP Redirect at CGI functions:
Documents, which are loaded via Redirect directive inside if a CGI function contains sometimes garbage data.

Fix: Fixed

CR Number: 811

Type: Defect

Component: Webserver

Category: Redirect

Synopsis: Invalid Redirect-Path

Description: The redirect option of the CGI Interface does not return the correct redirect path (some characters will be appended).

Fix: Fixed.

CR Number: 891

Type: Defect

Component: Webserver

Category: Web file upload

Synopsis: Incorrect file transfer

Description: Incorrect upload of files. There is always one byte more transmitted than the filesize.

Fix: Fixed

Type: Suggestion, Component: @CHIP-RTOS (5 item(s))

CR Number: 729

Type: Suggestion

Component: @CHIP-RTOS

Category: Exception handler

Synopsis: Extend the ESC opcode exception handler (0x07)

Description: Extend the ESC opcode exception handler (Vector 0x07), so that WATCOM C programs can run with floating point emulation.

Fix: Implemented (thanx to Frank v. Münchow-Pohl)

CR Number: 739

Type: Suggestion

Component: @CHIP-RTOS

Category: BIOSINT API 0x20, Error Handler

Synopsis: User error handler for low memory

Description: The user error handler should also be called, if a memory allocation call failed because of low memory.

Fix: Now User Fatal Error Handler will called with error code 9, if an alloc fails.

CR Number: 741

Type: Suggestion

Component: @CHIP-RTOS

Category: "Boot Ok" message

Synopsis: Add a "Boot ok" message

Description: Add a "Boot ok" message at RTOS startup, if no fatal errors were detected (e.g. "Fatal network error", Flash defect, CRC16 BIOS checksum error,). This allows an easier checking of possible defects for users and production.

Fix: Implemented

CR Number: 745

Type: Suggestion

Component: @CHIP-RTOS

Category: API interrupts

Synopsis: Missing "Int not supported" message

Description: TCPIP API, RTOS API, CGI API I2C API should print the default "Int not supported" message

Fix: If an API function will be called with an invalid function number at the AH-Register, the message "Int xx AH xx not supported" will now be printed at stdout.

CR Number: 821

Type: Suggestion

Component: @CHIP-RTOS

Category: Performance data

Synopsis: SC12/SC13: Measure and compare performance data

Description: Measure and compare performance data between SC12 and SC13:

- 1) IP Benchmarking
- 2) Interrupt Latency time
- 3) Task context switch

Fix: Done. See API documentation.

Type: Suggestion, Component: BIOSINT API (1 item(s))

CR Number: 742

Type: Suggestion

Component: BIOSINT API

Category: Execute

Synopsis: Return value of a RTOS shell command

Description: The BIOSINT API Function 0x07 (Execute a shell command) should return an error code.

Fix: Implemented.

Type: Suggestion, Component: CHIP.INI (1 item(s))

CR Number: 648

Type: Suggestion

Component: CHIP.INI

Category: Comments in Chip.ini

Synopsis: Comments should be possible in chip.ini

Description: Comments are now possible in chip.ini

Fix: Implemented, lines can be commented with semicolon

Type: Suggestion, Component: Command shell (3 item(s))

CR Number: 630

Type: Suggestion

Component: Command shell

Category: IPETH command

Synopsis: Command 'IPETH' should read the chip.ini.

Description: The command 'ipeth' should read the chip.ini. If the user changes the entries in chip.ini section IP, the new settings should become active when the 'ipeth' command is entered.

Fix: Implemented.

CR Number: 685

Type: Suggestion

Component: Command shell

Category: Boot messages

Synopsis: Error messages at boot time

Description: Printed messages of detected fatal errors at boot time (e.g. Flash Defect) are not good readable and could be easy overlooked.

Fix: Error message will now be shown at the end of the boot message.

CR Number: 819

Type: Suggestion

Component: Command shell

Category: IPCFG command

Synopsis: Complete listing of ip configuration

Description: The command IPCFG should list the current IP configuration of installed device interfaces: Ethernet, Internal loopback, PPP server, PPP client

Fix: Implemented.

In addition to the modified command, TCPIP API function 0x8D also returns the IP configuration of all installed device interfaces.

Type: Suggestion, Component: Ethernet (1 item(s))

CR Number: 828

Type: Suggestion

Component: Ethernet

Category: Send/Receive

Synopsis: Disabled interrupts

Description: During sending and receiving ethernet packets, interrupts were disabled for time intervals up to 300-500 mikroseconds. This lead to loss or blockade of other interrupts (e.g. Serial EXT/ COM interrupt or RTOS timer interrupt).

Fix: Fixed

Interrupts are no longer disabled during send/receive ethernet packets.

Type: Suggestion, Component: Filesystem (2 item(s))

CR Number: 664

Type: Suggestion

Component: Filesystem

Category: Write protection

Synopsis: Improve protection for writing flash memory

Description: Internal flash programming protection must be improved.

Fix: Implemented:

The internal flash write functions are checking now the flash sector number. (Write is only executed, if the requested flash sector is part of the filesystem).

CR Number: 733

Type: Suggestion

Component: Filesystem

Category: EXTIDE Auto-Format

Synopsis: Auto-format of external disk should be disabled

Description: If external drive open fails, the format call should be done by the user.

Fix: Auto-Format removed

Type: Suggestion, Component: Fossil API (1 item(s))

CR Number: 735

Type: Suggestion

Component: Fossil API

Category: Baudrate calculation

Synopsis: Baudrate calculation should be improved for more exact baudrate divisor

Description: The divisor for 1200 baud was not precise.

Fix: Implemented

Type: Suggestion, Component: Int21h (1 item(s))

CR Number: 793

Type: Suggestion

Component: Int21h

Category: Character out

Synopsis: Remove kernel sleep calls at Int21h character out functions

Description: For better performance, we should remove the kernel sleep calls, inside of Int21h 0x09 and 0x06
0x06: Sending a character to stdout.
0x09: Sending a string to stdout.

Fix: Removed sleep call.

Type: Suggestion, Component: PPP Client (1 item(s))

CR Number: 698

Type: Suggestion

Component: PPP Client

Category: Open call

Synopsis: PPP Client struct is only valid for one PPP_Client_Open Call.

Description: PPP Client struct is only valid for one PPP_Client_Open Call (TCPIPAPI function 0x41).
Closing a connection (by function 0x42) and reuse of the same PPPClient_struct at a following
open call doesn't work.

Fix: Fixed

Type: Suggestion, Component: RTOS API (1 item(s))

CR Number: 813

Type: Suggestion

Component: RTOS API

Category: RTOS resources

Synopsis: Limited number of event groups and timer procedures

Description: Because of a misunderstanding of available RTOS resources, we provided a fixed limit of event groups and timer procedures.
It should be possible to create as much event groups or timer procedures as semaphores are available from the RTOS kernel.

Fix: Implemented:

It is now possible to create as much event groups or timer procedures as semaphores are available from the RTOS kernel.

The sum of semaphores + event groups + timer procedures is in maximum 60.

Type: Suggestion, Component: TCPIP API (1 item(s))

CR Number: 736

Type: Suggestion

Component: TCPIP API

Category: SNMP MIB support

Synopsis: Extended SNMP MIB support

Description: For a better SNMP MIB support some of the internal SNMP variables must stored at new table structures. It's necessary to allow read access to the internal ARP cache and socket table for requesting actual state.

Fix: Implemented. See TCP/IP description:

Interrupt 0xAC service 0x8D: GET_IFACE_ENTRIES,
Interrupt 0xAC service 0x60: Get internal TCPIP SNMP variables,
Interrupt 0xAC service 0x8A: GET_ARPROUTE_CACHE,
Interrupt 0xAC service 0x23: API_FINDALL_SOCKETS

Type: Suggestion, Component: UDP Config Server (1 item(s))

CR Number: 887

Type: Suggestion

Component: UDP config server

Category: Communication protocol

Synopsis: Additional search key for identifying IPC@CHIPs at the network

Description: Because of the new IPC@CHIP products the serial number is no longer a unique key for that purpose. E.g. it can happen, that a IPC@CHIP variant SC13 has the same serial number as a SC12. In that case, it is not possible to update the software of an IPC@CHIP over TCP/IP UDP by using the serial number at the UDP config commands. The probability of such a conflict situation is very slight, but it is necessary to add a new identify method to the UDP config protocol.

Fix: From now on, the worldwide unique 48bit MAC-Address of the internal ethernet controller can also be used at UDP config commands instead of the serial number. The old method (identify by serial number) is still supported, because of compatibility reasons.

We will provide a new detailed description of the UDP config server protocol at our download page at the internet. The new extensions are used at our new CHIPTOOL release 4.0.1.8.

Type: Suggestion, Component: WebServer (2 item(s))

CR Number: 619

Type: Suggestion

Component: WebServer

Category: CGI

Synopsis: Case sensitive CGI names

Description: CGIs should be non-case-sensitive

Fix: Implemented

CR Number: 662

Type: Suggestion

Component: Webserver

Category: CGI

Synopsis: Memory leak if the CGI returns incorrect parameters

Description: If the users cgi function returns a value unequal to null in the fResponseBufferLength although the HTTP State tells that now data will be returned (e.g HTTP State "CGIHTTPNOTMODIFIED") every request allocates memory but does not free it.

Fix: Fixed

Type: New, Component: @CHIP-RTOS (3 item(s))

CR Number: 579

Type: New

Component: @CHIP-RTOS

Category: Customer product data

Synopsis: User specific flash area for customer product data

Description: Should provide a customer flash area, which is not part of the filesystem and not erasable by filesystem calls. The user is able to read/write this data by BIOSINT API calls.

Fix: Implemented,
BIOSINT API calls 0x45, 0x46 are provided for reading/writing the data

CR Number: 806

Type: New

Component: @CHIP-RTOS

Category: Detecting errors

Synopsis: Add a command which displays the detected errors

Description: Useful for usage of IPC@CHIP without serial console, because error messages that lead to a disabled autoexec.bat are currently only displayed at boot time over serial ports.

Fix: Implemented "errors" command.

CR Number: 888

Type: New

Component: @CHIP-RTOS

Category: Product data

Synopsis: Additional product data information for BECK IPC@CHIP-based products.

Description: It is planned, to store additional product data information for IPC@CHIP-based products, developed by BECK (e.g. BC440 based on SC12) at the internal flash memory of the IPC@CHIP. For these products an own serial number, device name and a hardware revision number should be stored at a reserved region of the internal flash memory.

Fix: Implemented.
The extended BIOSINT 0xA0 function 0x00 returns the provided information. It is planned to program this data at IPC@CHIP-based products, developed by BECK.

CR Number: 781

Type: New

Component: Fossil API

Category: Enable / Disable receiver

Synopsis: Fossil API call for enable/disable of UART receiver

Description: Should implement a Fossil API call for enable/disable the receive/transmit modes of the serial ports.

Fix: Implemented (Fossil API, functions 0x84 and 0x85)

CR Number: 892

Type: New

Component: Fossil API

Category: User callback function

Synopsis: User callback function at serial port events

Description: Should provide the installation of a user callback function, which will be executed by the @CHIP-RTOS at events concerning the serial port.

Fix: Implemented.

See description of Fossil API function 0xA1

CR Number: 423

Type: New

Component: HW API

Category: Interrupt functions

Synopsis: Should add an API call for installing kernel interrupt service functions

Description: HW API function 0x84 should also allow installing of RTX interrupt service functions.

Advantage of RTX interrupt service functions:

Inside of kernel interrupt service functions it is possible to call RTOS API functions.

The current interrupt service function doesn't allow this.

Fix: Implemented

CR Number: 605

Type: New

Component: HW API

Category: Disable/enable external interrupts

Synopsis: Add call for mask / unmask external interrupts

Description: Should add API call for mask / unmask external interrupts

Fix: Implemented.

HW API function 0x92

CR Number: 643

Type: New

Component: HW API

Category: Timer prescale Bit

Synopsis: Add timer0/1 prescale feature by timer2

Description: The internal used Timer2 (Millisecond timer) could be act as prescale timer. Then the timer base for the timer0/1 is timer2.

Fix: Implemented (see HW API Function 0x8F, Initialize Timer Settings Ext).

Type: New, Component: SPI (1 item(s))

CR Number: 181

Type: New

Component: SPI

Category: API

Synopsis: Implement a software SPI interface

Description: Should implement a software SPI interface (MISO, MOSI, MCLK, MEN)

Fix: Implemented

Type: New, Component: TCPIP API (4 item(s))

CR Number: 200

Type: New

Component: TCPIP API

Category: ARP/Route cache access

Synopsis: Add ARP/route cache access

Description: Should provide read/write access to the internal TCPIP ARP/Route cache

Fix: Implemented:
TCPIP API calls 0x88-0x8A allows access to the internal ARP cache.

CR Number: 467

Type: New

Component: TCPIP API

Category: User device interface for TCP/IP

Synopsis: Should add TCPIP API calls for own user device interface for TCP/IP

Description: It should be possible for the application programmer to add own device drivers and interfaces for TCP/IP. Possible application: Device driver for a connected wireless ethernet controller.

Fix: The TCPIP API calls 0xA0-0xA7 provide implementation of user specific device interfaces for TCPIP.

CR Number: 738

Type: New

Component: TCPIP API

Category: IP Callback

Synopsis: Implement IP Callback functionality/filter

Description: It should be possible to install an IP Callback function, in which the user can access to the IP Packet and its contents.

This function can be used as a filter for unwanted incoming IP packets.

Fix: Implemented.
TCPIPAPI function 0x7A

CR Number: 880

Type: New

Component: TCPIP API

Category: ARP protocol

Synopsis: ARP user callback

Description: Implementing a TCPIP API call, which allows the user to install a callback function on incoming ARP requests. The content of the incoming ARP request should be readable inside of the callback function. This function can be used as a user filter function. Depending on the return value of the callback function, the function can signal the TCPIP stack to ignore this request.

Fix: Implemented:
TCPIP API function provides the described functionality.

Type: New, Component: Webserver (1 item(s))

CR Number: 647

Type: New

Component: Webserver

Category: Security

Synopsis: User name and password for web server

Description: Implement user name and password for the web server. If a user name and password for the web server are defined at chip.ini, web server access requires authentication.

Fix: Implemented.
The user can define a user name and password at chip.ini for a specific webserver filesystem path.

Total: 61
