BECK
THE POINT OF SYNERGY

@CHIP-RTOS - CHIP.INI Documentation

CHIP.INI

The IPC@CHIP® system configuration is controlled via the CHIP.INI file. At startup, the system reads the file `A:\chip.ini` and uses the settings found here to initialize the system. The CHIP.INI file has the following structure:

```
[SECTION]
Item1=value1
Item2=value2
```

Sections have to be unique. Lines inside a section can be commented out with a leading semicolon. The maximum size of the CHIP.INI file is 65534 bytes.

LOGINDELAY
LOGINRETRIES
USERx
PASSWORDx

[SSH]
ENABLE
SSHPORT
PRIO
PRIO_KEX
RSA_KEYFILE
DSA_KEYFILE
USERx
PASSWORDx
USERx_KEYFILE
LOCAL_FORWARD
REMOTE_FORWARD
FORWARD_ALL
REKEY_DATA
REKEY_TIME

NAT_TIMEOUT
POLICY_FILE
PRESHARED_KEY

[FTP]
ENABLE
PRIO
CMDPORT
LOGINDELAY
TIMEOUT
USERx
PASSWORDx
ACCESSRIGHTx
DRIVEx
ROOTDIRx
CONNECTIONS
SSL_ENABLE
SSL_VERSION
SSL_CACERTx
SSL_SERVERCERT
SSL_KEYFILE
SSL_CTRLPERM
SSL_DATAPERM

SSL_CONNECTIONS
SSL_TIMEOUT
SSL_HTTPVERSION
SSL_VERSION
SSL_CACERTx
SSL_CLICACERTx
SSL_SERVERCERT
SSL_KEYFILE
SSL_EPHFILE

[DNS]
NAME_SERVER1
NAME_SERVER2
CACHE_SIZE

[TRACE]
FLASHWRITE
INTNOTSUPP

[UDPCFG]
LEVEL
PRIO
IPCFG_PASSWORD
FLASH_PASSWORD

[USB]
MAX_POWER_SUPPLY
BULK_PER_FRAME

[PPPCLIENT]
ENABLE
PRIO

[SERIAL]
EXT_DMA
COM_DMA
SER2_DMA
SER3_DMA
SEND_DMA
SEND_DMA2
EXT_RECVQUEUE
EXT_SENDQUEUE
COM_RECVQUEUE
COM_SENDQUEUE
SER2_RECVQUEUE
SER2_SENDQUEUE
SER3_RECVQUEUE
SER3_SENDQUEUE
COM_BAUD
EXT_BAUD

[STDIO]
STDIN=Define standard input device

Define your device for standard input. You can define several devices simultaneously.

SC1x Comments

Valid devices are COM, EXT and TELNET. By default, COM and TELNET are used.

SC1x3 Comments

Valid devices are COM, EXT, TELNET and SSH. The SSH enables the pseudo terminal established with a SSH session. By default, COM, TELNET and SSH are used.

SC2x Comments

Valid devices are COM, TELNET and SSH. The SSH enables the pseudo terminal established with a SSH session. By default, COM, TELNET and SSH are used.

Example

The following example defines COM and TELNET for stdin.

```
[STDIO]
STDIN=COM TELNET
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V1.30	V1.30

Top of list
Index page

[STDIO]
STDOUT=Define standard output device

Define devices used for standard output. You can define several devices simultaneously.

SC1x Comments

Valid devices are COM, EXT and TELNET. By default, both COM and TELNET are used.

SC1x3 Comments

Valid devices are COM, EXT, TELNET and SSH. The SSH enables the pseudo terminal established with a SSH session. By default, COM, TELNET and SSH are used.

SC2x Comments

Valid devices are COM, TELNET and SSH. The SSH enables the pseudo terminal established with a SSH session. By default, COM, TELNET and SSH are used.

Example

The following example defines both COM and TELNET for stdout.

```
[STDIO]
STDOUT=COM TELNET
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V1.30	V1.30

Top of list
Index page

[STDIO]
FOCUS=Command shell and/or user executables

Set the stdio focus to the command shell and/or to the user executables.

Valid entries are USER or SHELL
.
If only USER is defined, stdio in the command shell is suppressed.
If only SHELL is defined, stdout and stdin in the user's DOS executables are disabled.

Comments

The following example enables stdio for both USER and SHELL:

```
[STDIO]
FOCUS=SHELL USER
```

By default, stdin and stdout for both SHELL and USER are enabled.

Important : If stdio is enabled for both, there is a rivalry between USER and SHELL.

At runtime, pressing of the focus key (default is Ctrl-F) toggles between these three modes and shows the current mode.

Related Topics

FOCUS KEY configuration

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[STDIO] FOCUSKEY=Key

Defines the key that switches the stdio focus.

Comments

The following example sets Ctrl-F (ASCII 6) as the current stdio focus key:

```
[STDIO]
FOCUSKEY=6
```

By default, the focus key is set to CTRL-F (ASCII 6)
At runtime pressing Ctrl-F keys on the console will then cycle the stdio between the three modes:

Stdio: User
Stdio: Shell
Stdio: Both

The new mode is shown on the console.

Key Range: 0..254

If the key is set to zero, then no focus key is defined and the switching of stdio is disabled.

Note:

The focus key code is filtered out by the system, and will not be visible to either the command shell or a DOS executable. This key code is therefore not usable by either the command shell or a DOS executable.

Related Topics

Initial **FOCUS** configuration

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[STDIO]
CTRL_C=0/1

Disable/enable the possibility to prevent processing of the **AUTOEXEC.BAT** via CTRL + C key. The following example disables the ctrl-c control.

```
[STDIO]
CTRL_C=0
```

By default, CTRL_C is enabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[STDIO]
PRIO=prio

Defines the task priority of the shell task (MTSK).

Allowed value are 2-127. Default is 12.

Example:

```
[STDIO]
PRIO=20
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.23	V1.23	V1.23	V1.15	V1.00

Top of list
Index page

[IP]

ADDRESSx=IP address of the Ethernet interface

Defines the IPv4 address(es) of the internal Ethernet interface of IPC@CHIP®.

Comments

Only numerical IP addresses are allowed here.

Example: `ADDRESS=192.168.200.1`

If no address entry was found or `ADDRESS=0.0.0.0` , the IP address will be set to `1.1.1.1`. This special behaviour is required to enable UDP communication between the UDP config server and a requesting CHIPTOOL application for setting a valid IP configuration.

SC1x Comments

The SC1x allows only one IP address setup here with the tag ADDRESS.

SC1x3/SC2x Comments

At SC1x3/SC2x a maximum of five IP addresses may be listed with tags ADDRESS and ADDRESS1 up to ADDRESS4.
Note: The tag ADDRESS0 is not recognized, instead the tag ADDRESS must be used.

Related Topics

IP command line entry of IPv4 address
Set_IPConfig() - Set IPv4 configuration API
IPv6 address initial value

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	n/a	V1.36	V1.36

Top of list
Index page

[IP]
NETMASKx=IP address mask of the Ethernet interface

Defines the IPv4 subnet mask(s) of the internal Ethernet interface of IPC@CHIP®.

Comments

Example: `NETMASK=255.255.255.224`

If no subnet mask entry was found, the subnet mask will be set to `255.255.255.0`.

SC1x Comments

The SC1x allows only one IP address mask setup here with the tag NETMASK.

SC1x3/SC2x Comments

At SC1x3/SC2x a maximum of five IP address masks may be listed with tags NETMASK and NETMASK1 up to NETMASK4.
Note: The entry NETMASK0 is not recognized, instead the tag NETMASK must be used.

Related Topics

NETMASK command line
Set_IPConfig() - Set IPv4 configuration API

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	n/a	V1.36	V1.36

Top of list
Index page

[IP]
GATEWAY=Gateway IP Address

Setting the default IPv4 gateway.

Comments

Example: `GATEWAY=195.243.140.65`

The TCP/IP stack of the IPC@CHIP® supports only one valid default gateway for all device interfaces:
Ethernet and PPP Interface.
So see also PPP Server GATEWAY if you are using PPP.

We provide some additional API functions for modifying the default gateway:

- AddDefaultGatewayEx() - add a default gateway
- DelDefaultGateway() - delete default gateway
- GetDefaultGateway() - get default gateway

Important:
As a valid chip.ini gateway entry at this section [IP], the TCP/IP stack accepts only a gateway ip address, which matches to the net address of the internal Ethernet device ip configuration.
If the user adds an own device driver interface (e.g. second Ethernet on SC1x3) with a new gateway setting, he has to set this gateway inside of his application by calling AddDefaultGatewayEx.

Related Topics

CHIP.INI entry PPPSERVER GATEWAY
GATEWAY command line
Set_IPConfig() - Set IPv4 configuration API

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	n/a	V0.90	V1.00

Top of list
Index page

[IP]
DHCP=0/1 Ethernet interface

Set to 1 if DHCP client should be used to get the IP configuration for the internal Ethernet interface from a DHCP server.
If defined as 0, a static network configuration is used.

Comments

Any settings for IP Address, subnet mask and gateway are ignored if DHCP is used.

Related Topics

DHCP command line

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	n/a	V0.90	V1.00

Top of list
Index page

[IP]
HOSTNAME_OPT=0/1 Insert device name at DHCP requests as option 12

If set to 1, the IPC@CHIP® insert at its DHCP request its hostname as option 12. This is the name defined at Device name.
If the additional entry CLIENT_ID is not defined, the device name string is also inserted at the DHCP request as client identifier option 61.

Comments

By default this feature is disabled.

Related Topics

DHCP client identifier option 61

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.03	V1.00	V1.00	V0.90	V1.00

Top of list

Index page

[IP]
FORWARDING=0/1 Enables/disables ip forwarding

If set to 1, the TCP/IP stack of the IPC@CHIP® forwards ip packets between all network interfaces. Using forwarding the IPC@CHIP® acts as router between two networks. By example if an ip packet arrives on eth0 and the destination network address is equal to that one of eth1 the packet will be forwarded to eth1.
If defined as 0, ip forwarding is disabled for all interfaces.

SC1x Comments

This option is enabled by default.

SC1x3/SC2x Comments

If **SAFEMODE** is off, this option is enabled by default. If safe mode is on, this option is disabled by default.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.25	V1.25	V1.25	V1.20	V1.20

Top of list
Index page

[IP]
CLIENT_ID=Insert client identifier string at DHCP requests as option 61

If this entry is specified, the IPC@CHIP® insert at its DHCP request the specified string as DHCP option 61.
If this entry is not specified but the entry **HOSTNAME_OPT** is yet defined, the IPC@CHIP® inserts here its **Device name**.
If CLIENT_ID is set to the reserved word "MAC_ADDR", the IPC@CHIP® builds a 7 Byte long client identifier option consisting of:

1 Byte Hardware type, with value 1 for Ethernet
6 Bytes physical Ethernet address.

Example: `CLIENT_ID=MAC_ADDR`

Comments

By default this feature is disabled.
Maximum name size: 20 characters

Related Topics

DHCP hostname option 12

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.20	V1.20	V1.20	V0.91	V1.00

Top of list
Index page

[IP]
DHCP_TRIALS=Number of DHCP configuration trials

Specify the number of DHCP configuration trials.

Comments

The IPC@CHIP will try to obtain an IP configuration via DHCP several times. The number of trials can be configured with this option. Valid values are between 3 and 128. If you specify a value of 0, the configuration via DHCP will be retried endlessly (Not on SC1x!). Note that the configuration via **AutoIP** will be implicitely disabled in this case.

There are time intervals between each DHCP configuration attempt. At the first three DHCP trials, the DHCP client works with fixed time intervals.
The first three time intervals of the SC1x are 3, 3 and 6 seconds.
The first three time intervals of the SC1x3/SC2x are 3, 3 and 3 seconds.
After 3 DHCP configuration trials, the time interval is calculated by the term "1 << (trialNumber - 1)". E.g. Trial number 4 has a time interval of 8 seconds.
Time intervals are generally limited to 32 seconds.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.20	V1.20	V1.20	V0.91	V1.00

Top of list
Index page

[IP]
AUTOIP=0/1

If AUTOIP = 1 and DHCP = 1, the (first) Ethernet interface will be configured by the AutoIP algorithm, if the DHCP configuration fails.

Comments

Default is 1. The AutoIP procedure configures the Ethernet interface with a random IP address between 169.254.1.0 and 169.254.254.255. After selecting a random IP, it executes an address collision detection by sending ARP probes. The procedure repeats with new random IP addresses, until no collision can be detected.
It is also possible to call the AUTO IP procedure "by Hand" with the command **IPETH A**.

Note that if the **DHCP_TRIALS** parameter is set to 0, DHCP configuration will never reach the failed state and thus the configuration via AutoIP will not be started.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x

V1.20	V1.20	n/a	V1.05	V1.00

Top of list
Index page

[IP]
TCPIPMEM=Size

Set the size of the TCP/IP memory block in kBytes. This block is allocated at the start of the TCP/IP stack.

Example: `TCPIPMEM=60`

SC1x:

Valid Range: Between 30 kBytes and 160 kBytes (An out of range value for TCPIPMEM will be set to closest of these limit values.)

Default value: 90 kBytes when @CHIP-RTOS configured without PPP capability (server or client)
98 kBytes when @CHIP-RTOS configured with PPP capability

SC1x3/SC2x:

Valid Range: Between 60 kBytes and 1000 kBytes (An out of range value for TCPIPMEM will be set to closest of these limit values.)

Default value: 140 kBytes

Comments

The Get_TCPIP_Memory_Status() API reports the current used memory of the TCP/IP stack.

SC12 Comments

Since SC12 @CHIP-RTOS version 1.02B we allow configuring a maximum value of 160 kBytes, because some application programmers may require more then the old limit of 132 kByte.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V1.06	V1.00

Top of list
Index page

[IP]
TCP_PRIO=prio

Defines the task priority of the TCP timer task (TCPT).

Allowed value are 2-127. Default is 4.

Example:

```
[IP]
TCP_PRIO=10
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.23	V1.23	V1.23	V1.15	V1.00

Top of list
Index page

[IP]
ETH_PRIO=prio

Defines the task priority of the Ethernet receiver task (ETH0).

Allowed value are 2-127. Default is 5.

Example:

```
[IP]
ETH_PRIO=11
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.23	V1.23	n/a	V1.15	V1.00

Top of list
Index page

[IP]
ETH_MODE=Select operating mode of the internal Ethernet device

The internal Ethernet controller is able to operate in different operating modes. Select the following operating modes of the internal Ethernet device:

0: Auto negotiation, Default mode
1: Fixed 010 MBit/s, Half duplex
2: Fixed 010 MBit/s, Full duplex
3: Fixed 100 MBit/s, Half duplex
4: Fixed 100 MBit/s, Full duplex

Example: `ETH_MODE=1`

Comments

For avoiding possible configuration errors at different networks, we recommend the usage of the auto negotiation mode.

Related Topics

BIOS_Ethernet_State() - Detect Ethernet link state API function

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	V1.00	n/a	V0.90	V1.00

Top of list
Index page

[IP]
ETH_PHY=Configure Ethernet phy

This entry (an 8 bit hex value) provides a (possibly required) alternative configuration of the Ethernet PHY.
For experts only!
We strongly recommend that you leave the default settings (0x22) unchanged. Change the default settings only after having contacted us (support@beck-ipc.com).
Bit 7: Select cable type
0: UTP(100 Ohm) cable type (Default)
1: STP(150 Ohm) cable type
Bit 6: Receive Level Adjust
0: Normal (default)
1: Receive Squelch levels reduced by 4.5dB
Bit 5-2: Transmit Output Level Adjust:
4 Bit value for adjusting the transmit output levels from approx. -14% to +16% in steps of 2%.
Default value for Bit5-2: 1000b (0%)
Bit1-0: Transmitter Rise/Fall time
2 Bit value for changing the adjustment range from -0.25ns to +0.5ns in steps of 0.25ns.
Default value for Bit1-0: 10b.

Example for changing the cable type from UTP to STP: ETH_PHY=0xA2

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	V1.00	n/a	n/a	n/a

Top of list
Index page

[IP]
ETH_ENABLE=0/1

Defines whether the internal Ethernet controller of the IPC@CHIP® is started or not. If ETH_ENABLE=0 the Ethernet device is not started. By default the Ethernet device is enabled.

Comments

If ETH_ENABLE=0, the RTOS doesn't initialize the Ethernet controller at boot time.
Since SC1x3 RTOS 1.05 initialization of the Ethernet device can be done later by executing the command IPETH.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.20	V1.20	n/a	V0.90	V1.00

Top of list
Index page

[IP]
MTU=Bytes

Sets the Maximum Transfer Unit of the internal Ethernet interface.

```
[IP]
MTU=1300
```

By default, MTU is 1500. Min. value is 68, max. value is 1500.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.11	V1.01	n/a	V0.90	V1.00

Top of list
Index page

[IP]
ARPENTRIES=Entries

Set the max. number of possible entries at the internal ARP/Routing table of the TCP/IP stack.

Valid Range: Between 32 and 512 .
Default value: 64.

Example: `ARPENTRIES=128`

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.21	V1.21	V1.21	V1.06	V1.00

Top of list
Index page

[IP]
ARPSMART=0/1

Set to 1 if the TCPIP stack should store all incoming ARP replies.
Set to 0 if the TCPIP stack should cache only incoming ARP replies, which are requested by itself.
Sc1x Default value: 1, ARPSMART enabled
Sc1x3/SC2x Default value: 0, ARPSMART disabled

Example: `ARPSMART=0`

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.21	V1.21	V1.21	V1.06	V1.00

Top of list
Index page

[IPV6]
ENABLE=0/1 Enable the IPv6 protocol

Enable generally the usage of IPv6 protocol. By default IPv6 is disabled.

Related Topics

IPv6 stateless **auto configuration**
IPv6 address **initial value**
IPV6CFG - IPv6 configuration display command
Set IPv6 address **command**
Overview **IPv4/v6** - Dual layer stack

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[IPV6]
ADDRESSx=IPv6 address of the Ethernet interface

Specify an IPv6 address for the Ethernet interface. It is possible to configure up to 5 fixed IPv6 addresses for the Ethernet interface.

Comments

Example:

```
[IPV6]
ADDRESS0=3ffe:ffff:0:f101::1/64
ADDRESS1=4ffe:ffff:0:f101::1/64
```

The prefix length (e.g. the "/64" in above examples) must always be specified.

Related Topics

Enable **IPv6** protocol
IPv4 address initial value
IPv6 stateless **auto configuration**
IPV6CFG - IPv6 configuration display command
Set IPv6 address **command**
Overview **IPv4/v6** - Dual layer stack

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[IPV6]
STATELESS_AUTOCONF=0/1 Stateless auto configuration

Disable/Enable stateless IPv6 auto configuration. By default stateless auto configuration is enabled.

Comments

With stateless address auto configuration, a device interface is automatically assigned a link-local IPv6 address by the system. This link-local IPv6 address is generated with a fixed local address prefix (fe80::) to a token derived from the MAC address. (The address is verified to be unique.) This allows each IPv6 interface to have at least one source address that can be used by Neighbor Discovery. If an IPv6 router on the network advertises network prefixes in router advertisements, IPv6 derives IPv6 addresses based on the network interface identifier of the interface and on the network prefixes advertised.

There are 4 addresses left for further configurations by using the Neighbour discovery protocol.

Related Topics

Enable **IPv6** protocol
IPv6 address **initial value**
IPV6CFG - IPv6 configuration display command
Set IPv6 address **command**
Overview **IPv4/v6** - Dual layer stack

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[IPSEC]
ENABLE=0/1 Enable IP Security

Disable/Enable IP security. By default IP security is disabled. Note that the IP Security may also be started at run time with the **IPsec_Start()** API.

Related Topics

IKE command
IPsec_Start() API
IPsec_Add_Policy() API
NAT enable

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[IPSEC]
IKE_BUF_SIZE=Receiver buffer size

This entry specifies the size of the receiver buffer used by the Internet Key Exchange (IKE).

Example:

```
[IPSEC]
IKE_BUF_SIZE=4000
```

Comments

The default IKE receiver buffer size is 2048 bytes. This may be too small for cases where a significant chain of certificates is to be received from the IKE peer. The sum of the certificate file sizes in the anticipated CA chain can be used to judge the size of the required receiver buffer. Padding this sum by an additional 300 bytes should provide sufficient buffer space.

This entry will be limited internally to the range from 1024 bytes minimum up to 10000 bytes maximum.

Note that the **IKE** console command lists the peak IKE message size that has been received from the IKE peers. Appearance of the IKE **error** 100 indicates that you need to select a larger receiver buffer size with this INI file entry.

Related Topics

IKE command

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.15	V1.00

Top of list
Index page

[IPSEC]
IKE_CACERTx=CA certificate file name

File name(s) for IKE CA certificates. IKE_CACERT0 must be the name of the IKE's root CA certificate. Up to four further CA certifications can be listed with tags IKE_CACERT1 up to IKE_CACERT4 to indicate a certification chain. Possible file formats are *.PEM and *.DER.

The system scans for these entries in CHIP.INI starting at IKE_CACERT0 and terminates the scan at the first non-existant entry in the sequence.

Key lengths longer than 4096 bits are not supported.

Comments

Example:

[IPSEC]
IKE_CACERT0=ROOTCERT.DER

Related Topics

IPSEC **IKE_LOCALCERT** INI entry
IPSEC **IKE_KEYFILE** INI entry

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.15	V1.00

Top of list
Index page

[IPSEC]
IKE_CLICACERTx=File name of IKE peer's CA certificate

Up to five CA certificates for IKE peers may be specified with the 'x' in the above tag replaced with numbers '0' through '4'. The system scans for these entries in CHIP.INI starting at '0' and terminates the scan at the first non-existant entry.

Key lengths longer than 4096 bits are not supported.

Comments

Example:

[IPSEC]
IKE_CLICACERT0=CACERT.DER

Related Topics

IPSEC **IKE_CACERTx** INI entry

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.15	V1.00

Top of list
Index page

[IPSEC]

IKE_FQDN=FQDN Identification

This entry specifies the Fully Qualified Domain Name to be used in the IKE identification payloads. A check for this entry is made just before the IKE task is started.

Example:

```
[IPSEC]
IKE_FQDN=myDomainName
```

Comments

IKE outputs an identification payload both for phase 1 (Main Mode) and phase 2 (Quick Mode). For phase 1, this FQDN ID will only be used when **preshared key** form of identification is used. When **certificates** are used, the ID sent for phase 1 will always be the ID_DER_ASN1_DN (9) type with the payload containing the ID information extracted from the certificate.

Explicit control over whether the FQDN form of ID is used in either IKE phase 1 or phase 2 can be specified by prefixing the domain name with either "PH1," or "PH2," as in the following example which calls for FQDN ID use in phase 1, but not for phase 2.

```
[IPSEC]
IKE_FQDN=PH1,myDomainName
```

The "PH1," or "PH2," prefixes must be in upper case as shown, with no space characters before the domain name. No FQDN ID will be used for the above example when certificates are used for identification, as was noted above. This example would apply only when a preshared key is used.

If an '@' character appears in the FQDN string then the ID_USER_FQDN (3) type ID payload is sent instead of the ID_FQDN (2) type.

The configuration resulting from this CHIP.INI entry is visible at the **IKE** shell command.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.36	V1.36

Top of list
Index page

[IPSEC]
IKE_LOCALCERT=File name of our certificate

Defines the name of this system's local certificate. Possible file formats are *.PEM and *.DER.

Comments

Example:

```
[IPSEC]
IKE_LOCALCERT=OurCERT.DER
```

If this INI file entry is present and the certificate is successfully loaded, then the IKE will use Public Key Infrastructure (PKI) authentication method in phase 1 of the key negotiation process

unless:

1. The peer has initiated the IKE negotiation and has requested the use of **preshared key** for authentication.
2. The user has specified preshared key authorization method in the optional transform(s) set with the **IPsec_IKE_Phase1_Set()** API.

The **IPSEC_IKE_KEYFILE** and **IKE_CACERT0** files must also be specified for PKI operation.

Related Topics

IPSEC **IKE_CACERTx** INI entries
IPSEC **IKE_KEYFILE** INI entry

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.15	V1.00

Top of list
Index page

[IPSEC]
IKE_KEYFILE=File name of the IKE key file

Specifies the name of private and public key file. used for IKE. Possible file formats are *.PEM and *.DER.

Key lengths longer than 4096 bits are not supported.

Comments

Example:

[IPSEC]
IKE_KEYFILE=PRIVKEY.DER

Related Topics

IPSEC **IKE_LOCALCERT** INI entry

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.15	V1.00

Top of list
Index page

[IPSEC]
IKE_PH1_AGGRESSIVE=IKE Phase 1 Aggressive mode

This entry specifies the Internet Key Exchange (IKE) should use aggressive mode for phase 1 SA negotiation.

Example:

```
[IPSEC]
IKE_PH1_AGGRESSIVE=1
```

Comments

By default this option is deselected (value 0), which causes the phase 1 IKE negotiation to operate in the "main mode", also referred to as Identify Protection Exchange. (See RFC 2408 section 4.5.)

When this switch is set to 1 as shown in the example, then the phase 1 IKE will use the Aggressive Exchange which does not hide identifies. A few less messages must travel between computers in this case. (See RFC 2408 section 4.7.)

Note:

> Microsoft operating systems have not supported this mode. You will likely get an IKE error code 7, "Invalid exchange type", when Aggressive mode is used with a Microsoft peer system.

Related Topics

IPsec_Set_Option() API
IPsec_IKE_Phase1_Set() API

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.20	V1.20

Top of list
Index page

[IPSEC]
IKE_PH2_PFS=PFS mode for IKE phase 2

This entry specifies the PFS mode to be used by Internet Key Exchange (IKE) Quick Mode.

Example:

```
[IPSEC]
IKE_PH2_PFS=0
```

Comments

Perfect Forward Secrecy (PFS) mode will be used during Quick Mode SA key negotiation when this switch is set to '1'. This is the default state of this switch.

PFS mode can be disabled by setting this switch value to zero.

Refer to RFC 2409 section 3.3 for a precise definition of PFS.

Related Topics

IPsec_Set_Option() API

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.20	V1.20

Top of list
Index page

[IPSEC]
IKE_PRIORITY=priority

This entry specifies the priority at which the Internet Key Exchange task, "_IKE", executes.

Example:

```
[IPSEC]
IKE_PRIORITY=35
```

Comments

Default priority is 50. This task can occupy 100% of the CPU for an expended period of time (seconds) during the key exchange processing. Consequently, this low default task priority is assigned to it.

Related Topics

IPsec_Start() API
IPsec_Add_Policy() API

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.10	V1.00

Top of list
Index page

[IPSEC]
NAT=0/1 Enable IP Sec NAT-Transversal

Disable/Enable IP Security transversal of NAT devices (Network Address Translator). By default IP Security NAT-Transveral mode is enabled.

When this mode is enabled, the IKE will perform NAT detection and negotiate NAT-transversal mode with a peer as specified in RFC-3947 and use, when necessary, the UDP encapsulated ESP per RFC-3948 in order to provide IP security across the NAT. The UDP port 4500 is used for this purpose.

Normally it does no harm to leave this switch in its default enabled state. However there may be cases with some routers that interfere with the IKE operation as they try to "help" due to being IP Security aware, where it becomes necessary to disable the NAT-Transversal functionality in the IPC@CHIP® in order to get the IP Security to operate across a NAT.

Note that **AH protocol** (Authentication Header) is not supported when a NAT is present on the path between IPsec peers. The security **policy** must be set accordingly (ESP only!).

Note that the top level **IPSEC ENABLE** switch must be enabled for this switch to have affect.

Related Topics

IPsec **NAT** operation
IPsec **ENABLE**
NAT_TIMEOUT keep-alive
NAT_LINGER keep-alive

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.16	V1.01

Top of list
Index page

[IPSEC]
NAT_LINGER=NAT Keep-Alive Linger Time

This entry specifies the number of minutes that the NAT keep-alive UDP packets will continue to be sent out after the related SA (Security Association) have been deleted. (Ref: RFC 3948 section 4.)

```
[IPSEC]
NAT_LINGER=6
```

Comments

Default NAT keep alive linger time is 5 minutes. The range of legal entry values is from 1 to 720 minutes. Values outside this range will be changed to the nearest in range value.

The purpose of the NAT keep-alive UDP packets is to maintain the NAT address/port mappings. The linger time allows these mappings to be preserved as SA (Security Association) are re-negotiated after SA timeouts have occurred.

Related Topics

NAT enable
NAT_TIMEOUT keep-alive

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.16	V1.01

Top of list
Index page

[IPSEC]
NAT_TIMEOUT=NAT Keep-Alive Packet Interval

Specify the interval in seconds at which NAT keep-alive UDP packets will be output when IP Security SA (Security Association) are established with the IPC@CHIP® lying behind a NAT device. (Ref: RFC 3948 section 4.)

```
[IPSEC]
NAT_TIMEOUT=30
```

Comments

Default interval is 20 seconds. The range of legal entry values here is from 1 to 3600 seconds. Values outside this range will be changed to the nearest in range value.

Related Topics

NAT enable
NAT_LINGER keep-alive

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.16	V1.01

Top of list
Index page

[IPSEC]
POLICY_FILE=Restore IP Security Policies from file

This entry specifies a binary **file** that contains the IP Security policies and/or preshared keys to be installed either at startup (assuming IPSEC **ENABLE** set to 1) or at run time when **IPsec_Start()** API is called.

Example:

```
[IPSEC]
POLICY_FILE=ipsec.ips
```

Comments

This file can be constructed with the **IPsec_Store_Policy** C-library function or with Beck's IP Security Policy Editor tool, BeckIPSec, available for use with a Windows PC. The file extension on this policy data file can be anything. The IPS extension is the convention used by the BeckIPSec tool.

Related Topics

IPsec_Store_Policy() API
tIPSEC_FILE_HEADER Header format

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[IPSEC]
PRESHARED_KEY=IKE Peer ID ; Key String

This entry specifies both the identity of an IKE (Internet Key Exchange) peer and an ASCII string to be installed for a preshared key used with that peer. A semicolon is used to delimit the two fields of this entry.

Example:

```
[IPSEC]
PRESHARED_KEY=192.168.30.145;MyPresharedKey#1
```

Comments

Only one preshared key may be specified in this INI file. Additional keys may be added at runtime using the IPsec_Add_Preshared_Key() API or from a policy file.

The peer identity can be a string specifying an IPv4 address, an IPv6 address, a fully qualified domain name or a user name. This string must match the manner that the peer uses to identify itself within the IKE protocol.

White space should be avoided immediately after the semicolon, otherwise this white space will be included as the first part of your key.

The total string entered after the "PRESHARED_KEY=" is limited to 260 characters. The string pair is truncated after 260 characters total.

Related Topics

IPsec_Add_Preshared_Key() API
IPSEC IKE_LOCALCERT INI entry

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[UDPCFG]
LEVEL=mask

Defines the supported functions of the configuration server.

LEVEL is a bit mask. Each bit represents one function. Setting the resp. bit to 1 enables the function, setting it to 0 disables the function.
The bit assignment is as follows:

- **Bit 0:** Detection of the IPC@CHIP® on the network
- **Bit 1:** IP configuration
- **Bit 2:** Password protection for IP configuration (see here)
- **Bit 3:** Reserved for future use
- **Bit 4:** Programming of flash (RTOS update)
- **Bit 5:** Password protection for programming of flash (see here)
- **Bit 6:** Disable transmission of Hello answers via PPP connections
- **Bits 7-14:** Reserved for future use
- **Bit 15:** This bit can be set to start the server without any function enabled. This could be useful if you want to set from an application at run-time.

If the configuration level is set to 0x00, the configuration server task will not be started at all.

Example:
```
[UDPCFG]
LEVEL=0x03
```

This would allow detection on the network and IP configuration, but no RTOS update.

Comments

The password protection functions (bits 2 and 5) are only available on SC1x3 and SC2x.

SC1x Comments

The default value is 0x13.

SC1x3/SC2x Comments

If SAFEMODE is off, the default value is 0x0013. If safe mode is on, the default value is 0x8000.

Related Topics

BIOS_Config_Server_Level() - Run-time adjustment of configuration level

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V1.15	V1.00

Top of list
Index page

[UDPCFG]
PRIO=prio

Defines the task priority of the configuration server (CFGS).

Legal values are in the range 2-127; default is 7.

Example:
```
[UDPCFG]
PRIO=30
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.20	V1.20	V1.20	V1.05	V1.00

Top of list
Index page

[UDPCFG] IPCFG_PASSWORD=password

Sets the password for IP configuration of the IPC@CHIP® via the configuration server. The password must consist of up to 16 printable non-whitespace characters.

Password protection must be enabled by setting the respective bit in the configuration level.

This password is read in from the CHIP.INI file only at system startup time. Any changes made thereafter to this value will not take affect until the system is rebooted.

Examples:
```
IPCFG_PASSWORD=test
IPCFG_PASSWORD=abc123
IPCFG_PASSWORD=,zyx-.#321
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.15	V1.00

Top of list
Index page

[UDPCFG] FLASH_PASSWORD=password

Sets the password for flash programming of the IPC@CHIP® via the configuration server. The password must consist of up to 16 printable non-whitespace characters.

Password protection must be enabled by setting the respective bit in the configuration level.

This password is read in from the CHIP.INI file only at system startup time. Any changes made thereafter to this value will not take affect until the system is rebooted.

Examples:
```
FLASH_PASSWORD=test
FLASH_PASSWORD=abc123
FLASH_PASSWORD=,zyx-.#321
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x

n/a	n/a	n/a	V1.15	V1.00

Top of list
Index page

[DNS]
NAME_SERVER1=IP address of the first name server

Adds a name server to the DNS resolver's name server list.
Normally the DNS resolver determines name server addresses automatically from network devices configured via DHCP or PPP. In addition you can specify two name servers in the CHIP.INI configuration file and via the function `setNameServer()` API. Name servers found via DHCP have the lowest priority. Next come name servers found via PPP, then the two configured in CHIP.INI. The name server added via function call has the highest priority.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.05	V1.00

Top of list
Index page

[DNS]
NAME_SERVER2=IP address of the second name server

See NAME_SERVER1.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.05	V1.00

Top of list
Index page

[DNS]
CACHE_SIZE=Number of entries of the DNS cache

Defines the number of entries in the DNS cache. More entries speed up DNS functions. But note that each entry consumes about 520 bytes of RAM memory. The default value is 30.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.05	V1.00

Top of list
Index page

[PPPCLIENT]
ENABLE=0/1

Disable/enable PPP client task

```
[PPPCLIENT]
ENABLE=1
```

By default, PPP client is enabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPCLIENT]
PRIO=prio

Defines the task priority of the PPP client task (PPPC).

Allowed value are 2-127. Default is 6.

Example:

```
[PPPCLIENT]
PRIO=8
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.23	V1.23	V1.23	V1.15	V1.00

Top of list
Index page

[PPPSERVER]
ENABLE=0/1

Disable/enable PPP server

```
[PPPSERVER]
ENABLE=1
```

By default, PPP server is disabled.

Related Topics

PPP server configuration instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
PRIO=prio

Defines the task priority of the PPP server task (PPPS).

Allowed value are 2-127. Default is 6.

Example:

```
[PPPSERVER]
PRIO=8
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.23	V1.23	V1.23	V1.15	V1.00

Top of list
Index page

[PPPSERVER]
IPV6_USE=0/1 Enable the IPv6 protocol

Enable the usage of IPv6 protocol at the PPP server and disable IPv4. By default IPv6 is disabled, the PPP server runs with IPv4.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
MODEMTRACE=0/1

Disable/enable the trace of the control communication between IPC@CHIP® and a connected modem.
The modem strings (AT commands and answers) defined in chip.ini will be printed on STDOUT, if MODEM_TRACE=1.
This can be useful for testing the modem configuration and debugging the PPP dial procedures.

```
[PPPSERVER]
MODEMTRACE=1
```

By default, tracing is disabled.

Comments

If a command should be send from IPC@CHIP® to the modem, the MODEMTRACE output indicates only, that the command was successful placed at the send queue of the serial port. It doesn't indicate, that the command was physically send on the wire (e.g. if hardware flow control blocks the send queue).
Received characters with an ASCII value smaller than 0x20 are printed as numbers.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
COMPORT=Define serial device for the PPP server

Define your serial device for the PPP server.

SC1x:
Valid devices are COM or EXT.

SC1x3:
Valid devices are COM, EXT, SER2 or SER3.

SC2x:
Valid devices are COM, SER2 or SER3.

Comments

The following example defines EXT as device for the PPP server:

```
[PPPSERVER]
COMPORT=EXT
```

By default, no serial port is enabled.

Related Topics

PPP server configuration instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
ADDRESS=IP Address of the PPP server interface

Defines the IP address for the PPP server.

Comments

Only numerical IP addresses are allowed here.
Example: `ADDRESS=192.168.205.1`
If no address entry was found, the address will be set to `1.1.2.1`.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
REMOTEADDRESS=IP Address for the remote PPP client

Defines the IP address for the remote PPP client.

Comments

Only numerical IP addresses are allowed here.
Example: `REMOTEADDRESS=192.168.205.2`
If no address entry was found, the remote address will be set to `1.1.2.2`.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
NETMASK=IP Address subnet mask of the PPP server

Defines the IP address subnet mask of the PPP server.

Comments

Example: `NETMASK=255.255.255.0`

If no subnet mask entry was found, the subnet mask will be set to `255.255.255.0`.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
GATEWAY=IP Address of gateway

Defines the IP address of the gateway if PPP Connection is established.

Comments

Example: `GATEWAY=195.243.140.65`

If no gateway entry was found, the gateway will be untouched.

The TCP/IP stack of the IPC@CHIP® supports only one valid default gateway for all device interfaces:
Ethernet, PPP Interface.

If you define a gateway in the PPPSERVER section of the `chip.ini`, it becomes the default gateway for all interfaces when a PPP link to the server is established. The default gateway must be the same IP Address as the remote peer.
It does not make sense to define a different IP, because the remote Peer is the only peer, which is reachable. If a different IP than the remote IP is defined, the remote IP will used for the gateway entry automatically.
During a PPP server connection the command **`ipcfg`** indicates this default gateway. After the PPP session, the old gateway (if any was defined) will be restored.

Also we provide some functions for modifying the default gateway:

- **AddDefaultGatewayEx()** - add a default gateway
- **DelDefaultGateway()** - delete default gateway
- **GetDefaultGateway()** - get default gateway

Related Topics

CHIP.INI entry **IP (Ethernet) GATEWAY**
PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
AUTH=0/1/2

Set PPP authentication mode for the remote PPP client

0: No authentication
1: PAP authentication
2: CHAP authentication

Comments

The following example selects PAP authentication mode:

```
[PPPSERVER]
AUTH=1
```

By default, authentication is disabled.

If AUTH!=0 you must define two **user name** / **password** pairs used to authenticate the PPP client. The client must use one of these pairs to get connected to the IPC@CHIP® PPP server.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
MTU=Bytes

Sets the Maximum Transfer Unit of the PPP server interface.

```
[PPPSERVER]
MTU=1300
```

By default, PPP server MTU is 1024. Min. value is 68.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.11	V1.01	V1.01	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
IDLETIME=Seconds

Sets the idle time, after which the PPP server closes the connection.

```
[PPPSERVER]
IDLETIME=500
```

By default, PPP server idle time is 120 seconds. A value of 0 means no idle timeout.

Related Topics

PPP server configuration instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
FLOWCTRL=0/1/2

Set flow control mode of the PPP server's serial device:

0: none
1: XON/XOFF (See caution below!)
2: RTS/CTS

Example: Here XON/XOFF flow control is enabled

```
[PPPSERVER]
FLOWCTRL=1
```

By default, `FLOWCTRL=2` (RTS/CTS)

Comments

Caution:

If you use the default DMA mode for the selected COM port, it is not recommended to choose XON/XOFF flow control mode (see Fossil API Xon/XOff usage (Chap.3)).

Related Topics

PPP server configuration instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
MODEM=0/1

Disable/enable usage of a modem

```
[PPPSERVER]
MODEM=1
```

By default, the usage of a modem is disabled.

Related Topics

PPP server configuration instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
DCDPIO=Pio number

Enable and define a pio as low-active modem DCD input signal for the PPP server

```
[PPPSERVER]
DCDPIO=13
```

By default, a DCD pio is not defined

Comments

The serial ports of the SC1x/SC1x3/SC2x are offering no DCD input for modem line detection. If a DCD pio is defined, the PPP server will (at the connected state) cyclic read the pio (connected with the DCD output of the modem) and close an established connection, if it detects low signal at the pio.

Related Topics

PPP server configuration instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[PPPSERVER]
USERx=user

Define the user name for PPP server, using PAP or CHAP authentication

Comments

You can define a USER0 and a USER1.
Default user is 'ppps', default password is 'ppps' for both USER0 and USER1.
You must specify both the user name and password.

If PAP authentication is used, neither user name nor password are case sensitive.
If CHAP authentication is used, user name and password specified here are converted to lower case. The remote side must also specify the user name and password in lower case, if CHAP is used.

The entries are only valid if **AUTH** != 0 is specified.

Maximum name size: 49 characters

Important notice: To avoid security leaks you must define both user names and passwords.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
PASSWORDx=password

Define the password for a PPP server user, using PAP or CHAP authentication.

Comments

You can define a PASSWORD0 for USER0 and a PASSWORD1 for USER1
Default user is 'ppps', default password is 'ppps'.

If PAP authentication is used, neither user name nor password are case sensitive.
If CHAP authentication is used, user name and password specified here are converted to lower case. The remote side must also specify the user name and password in lower case, if CHAP is used.

The entries are only valid if **AUTH** != 0 is specified.

Maximum password size: 49 characters

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
BAUD=BAUD Rate

Sets the BAUD rate of the PPP server serial port.

Comments

The following example sets the PPP server serial port to 19,200 BAUD.

```
[PPPSERVER]
BAUD=19200
```

By default, PPP server BAUD rate is 38400 (with 8 data bits, no parity, 1 stop bit).

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
INITCMDx=modem command

Defines modem commands to initialize your modem connected to the IPC@CHIP®
at the start of the IPC@CHIP® PPP server and after a modem hang-up following a PPP session.

Comments

You can define a maximum of 3 modem commands e.g. `INITCMD0=ATZ` .
The entries are only valid if **MODEM**=1 is specified.
The maximum length for each command string is 25 characters.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
INITANSWERx=modems answer of init command x

Defines the expected modem answer x for the initialize command x.

Comments

You can define a maximum of 3 modem answers e.g. `INITANSWER0=OK` .
The entries are only valid, if **MODEM**=1 is specified.
The maximum length for each answer string is 80 characters.

Default for all answer strings: OK

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
INITTIMEOUTx=timeout in seconds for wait an the modem's answer

Define the timeout value in seconds for waiting on an answer from the modem.
A value of 0 means wait forever for the modem answer.

Comments

Example: `INITTIMEOUT0=2`
The entries are only valid if **MODEM**=1 is specified.

Default value: 3 seconds

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
INITRETRIESx=Retries, if the modem init answer failed

Define the number of retries used when the modem initial answer fails.

Comments

Example: `INITRETRIES0=2`
This entry is only valid if **MODEM**=1 is specified.

Default value: 1

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
MODEMCTRL=0/1/2

Allow modem online control by PPP server.

```
[PPPSERVER]
MODEMCTRL=1
```

By default, the usage of a modem online control is disabled.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
CTRLTIME=Seconds

Sets the idle interval time, at which the PPP server executes the configured control commands (see **CTRLCMDx**).

```
[PPPSERVER]
CTRLTIME=120
```

By default, PPP server idle control time is 60 seconds.

If the PPP server doesn't receive regular PPP data during this interval, it executes the control commands . If execution of one of the control commands fails, the PPP server then closes the connection.

The CTRLTIME must be a smaller value than the **IDLETIME** of the PPP server.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
CTRLCMDx=modem online control command

Defines modem command to control if modem is online or not at the start of the IPC@CHIP® PPP server and after a modem hang-up following a PPP session.

Comments

You can define a maximum of 3 modem commands e.g. `CTRLCMD0=+++` .
The maximum length for each control command string is 25 characters.

The entries are only valid, if **MODEMCTRL**=1 is specified.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
CTRLANSWERx=modems answer of ctrl command x

Defines the expected modem answer x for the online control command x.

Comments

You can define a maximum of 3 modem answers e.g. `INITANSWER0=OK` .
The entries are only valid, if **MODEMCTRL**=1 is specified.
The maximum length for each answer string is 80 characters.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
CTRLTIMEOUTx=timeout in seconds for wait on the modem's answer

Defines the timeout value in seconds for waiting on an answer from the modem.
A value of 0 means wait forever for the modem answer.

Comments

Example: `CTRLTIMEOUT0=2`
The entries are only valid if **MODEMCTRL**=1 is specified.

Default value: 3 seconds

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
CTRLRETRIESx=Retries, if the modem online control answer failed

Defines the number of retries used when the modem control answer fails.

Comments

Example: `CTRLRETRIES0=2`
This entry is only valid if **MODEMCTRL**=1 is specified.

Default value: 1

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
CMDMODE=switch to modem command mode

Defines the string which switches the modem into the command mode.

Comments

The entries are only valid if **MODEM**=1 is specified.

Default string for CMDMODE:+++

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
HANGUPDELAY=Time in seconds for switching modem into command mode

Defines the time in seconds for switching modem into command mode for hang-up commands.

Comments

The entries are only valid if **MODEM**=1 is specified.

Default time: 2 seconds

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER] HANGUPCMDx=modem command

Defines modem commands to hang-up the modem connected to the IPC@CHIP®.

Comments

You can define a maximum of 3 modem hang-up commands e.g. `HANGUPCMD0=ATH` , which will be executed if the PPP connection is closed.
The maximum length for each command string is 25 characters.

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER] HANGUPANSWERx=modems answer for hang-up command x

Defines the expected modem answer x for the hang-up command x

Comments

You can define a maximum of 3 modem answers e.g. `HANGUPANSWER0=OK` .

The maximum length for each answer message is 80 characters.
Default for all answer strings: OK

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER] HANGUPTIMEOUTx=timeout in seconds for wait on answer from modem

Defines the timeout value in seconds used when waiting on the modem's answer.
A value of 0 means wait forever.

Comments

Example: `HANGUPTIMEOUT0=2`

Default value: 3 seconds

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER] HANGUPRETRIESx=Retries, if the modem hang-up answer failed

Defines the number of retries used if the modem hang-up answer fails.

Comments

Example: `HANGUPRETRIES0=2`

Default value: 1 try

Related Topics

PPP server configuration instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER] CONNECTMSGx=modem message

Defines the expected modem message to get connected to a peer modem

Comments

You can define a maximum of three modem messages e.g. `CONNECTMSG0=RING` .
The maximum length for each connect message is 25 characters.

Defaults:
`CONNECTMSG0=RING`
`CONNECTMSG1=CONNECT`

Related Topics

PPP server configuration instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER] CONNECTANSWERx=modem command for incoming connect message x

Defines the expected modem answer x for the incoming connect message x

Comments

You can define a maximum of three modem answers e.g. `CONNECTANSWER0=ATA` .
The maximum length for each answer string is 80 characters.

Defaults: `CONNECTANSWER0=ATA`

Related Topics

PPP server configuration instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[PPPSERVER]
CONNECTTIMEOUTx=timeout seconds for wait on the modem's connect message x

Defines the timeout value in seconds used when waiting on the modem connect message.
A value of 0 means wait forever.

Comments

Example: `CONNECTTIMEOUT0=0`

Default values:
`CONNECTTIMEOUT0=0`
`CONNECTTIMEOUT1=60`

Related Topics

PPP server **configuration** instructions

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[RAMDRIVE]
SIZE=size

Set the size in KByte of the RAM drive C:.
If defined as 0, no RAM drive is configured.

SC1x:
Maximum size is 256 Kbyte

SC1x3:
Maximum size is 6144 Kbyte, if internal memory is used.
Maximum size is 8192 Kbyte, if external memory is used.
Default is to use internal memory.

SC2x:
Maximum size is 6144 Kbyte.

Comments

Default size: 0 (no RAM drive C:)

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V1.05	V1.00

Top of list
Index page

[RAMDRIVE]
SEGMENT=segment

Set the start segment of the RAM drive C:.
If defined, the RAM drive memory is taken from external UCS# memory.
Default is to use internal memory.

Example:

```
[RAMDRIVE]
SEGMENT=0x8000
SIZE=512
```

Comments

Segment must be greater or equal 0x8000.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.05	n/a

Top of list
Index page

[RAMDRIVE]
WS=wait states

Set the number of wait states if RAM drive uses external memory.
Allowed values are 0-15
Default is to use 9 WS.

Example:

```
[RAMDRIVE]
WS=4
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.05	n/a

Top of list

Index page

[TIMER]
1C=ms

Sets the interval in milliseconds for timer interrupt 0x1C.
Range: 1 to 32767, Default value=55 ms.

Related Topics

BIOS_Set_Timer_0x1C() - API to set timer interrupt 0x1C's interval

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[TIMER]
AF=ms

Sets the interval in milliseconds for timer interrupt 0xAF.
Range: 1 to 32767, Default value=4 ms.

Related Topics

BIOS_Set_Timer_0xAF() - API to set timer interrupt 0xAF's interval

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[TIMER]
RTI=kHz

SC1x3/SC2x:

On SC1x3/SC2x systems, the Real-Time Interrupt (RTI) can be operated at rates above the normal 1000 Hz rate. This value is specified in kHz and can be set to any of the following legal rates:

[1, 2, 3, 4, 5, 6, 8, 10, 12, 15, 16, 20, 24, 25, 30, 32, 40, 48, 50]

SC1x Comments

For SC1x targets, the RTI is fixed at a 1 kHz rate.

SC1x3/SC2x Comments

The default rate of 1 kHz will be used for invalid entries not in the set of legal rates listed above. The RTI rate in effect is visible in the CX register returned by the **RTX_RTI_COUNT** API. The C-library function **RTX_Get_RTI_Rate** accesses this count.

If you will be using the C-library **RTX_GetTick_us** API, then for proper operation of this library function you are restricted to the following RTI settings:

[1, 2, 4, 5, 8, 10, 20, 25, 40, 50]

Related Topics

Using higher RTI rates

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.05	V1.00

Top of list
Index page

[FTP]
ENABLE=0/1

Define if the FTP server should be activated.

Comments

Use 0 to disable, 1 to enable.

SC1x Comments

By default, the FTP server is enabled.

SC1x3/SC2x Comments

If **SAFEMODE** is off, the FTP server is enabled per default. If safe mode is on, the FTP server is disabled per default.

Related Topics

FTP enable/disable command
BIOS_Server_On_Off() - API to Suspend/Resume system servers

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[FTP]
PRIO=prio

Defines the task priority of the FTP server task (FTPS).

Allowed value are 2-127. Default is 41.

Example:

```
[FTP]
PRIO=80
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.23	V1.23	V1.23	V1.15	V1.00

Top of list
Index page

[FTP]
CMDPORT=port

Set the command port number of the FTP server.
Default FTP command port: 21
Example:

```
[FTP]
CMDPORT=5000
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[FTP]
LOGINDELAY=0/1

Define if the delayed login of the FTP server should be (de)activated.

Comments

Use 0 to deactivate, 1 to activate.

By default, the delayed login is enabled.
The delay time starts with 400 milliseconds.
After each following failed login, the delay time will be doubled until it reaches 20 seconds.
After a successful login the delay time will be set back to 400 milliseconds.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.01	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[FTP]
TIMEOUT=sec

Defines the inactivity timeout for the FTP server in seconds.
The minimum value for the timeout is 20 seconds and maximum is 65535 seconds.

Comments

Default FTP timeout is 300 seconds.
RFC 1123 states that the minimum idle timeout should be 5 minutes.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[FTP]
USERx=user

Defines the user name for FTP.

Comments

You can define a USER0 and a USER1.
Default users are: 'anonymous' (no password) and 'ftp' (password is 'ftp').
You must specify both the user name and their password.
Neither user name nor password are case sensitive.
Maximum name size: 19 characters

Important notice: To avoid security leaks you must define both user names and passwords.

Related Topics

FTP user write protection

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[FTP]
PASSWORDx=password

Define the password for a FTP user

Comments

You can define a PASSWORD0 for USER0 and a PASSWORD1 for USER1
Default users are anonymous (no password) and ftp (password is 'ftp').
Neither user name nor password are case sensitive.
Maximum password size: 19 characters

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[FTP]
ACCESSRIGHTx=Access rights for defined Users

This CHIP.INI entry allows you to deny write access to FTP USER0 or USER1.

0 - write and read access enabled
1 - write access denied, read access enabled

Example which disables FTP write access for USER0:

```
[FTP]
USER0=otto
PASSWORD0=otto53pass
ACCESSRIGHT0=1
```

Comments

You can only forbid write access if you have defined the respective user with the FTP **USERx** and **PASSWORDx** entries.

By default write access is enabled for both FTP users.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x

V1.02	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[FTP]
DRIVEx=Set user's FTP drive

Set user's FTP drive.

Comments

Entries DRIVE0 or DRIVE1 can be made to specify a particular drive for use by FTP USER0 and USER1 respectively. The drive numbers are coded as follows:

0: Drive A
1: Drive B
2: Drive C
3: Drive D (SC1x3/SC2x only)
25: Drive Z (SC1x3/SC2x only)

If the DRIVE entry and the ROOTDIR entry do not exist, a virtual root directory will be set, that lists all available drives.
If only the DRIVE entry does not exist, the default drive A will be set.

The following example defines the root drive for USER0 to be on B: drive.

```
[FTP]
DRIVE0=1
```

Related Topics

FTP user's root directory

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.01	V1.00	V1.00	V1.10	V1.00

Top of list
Index page

[FTP]
ROOTDIRx=Name of the user's FTP server root directory

Defines the name of user's FTP server root directory.

Comments

The following example defines the root directory for USER1

```
[FTP]
```

```
ROOTDIR1=USERDIR
```

The default FTP root directory is the drive root directory, "\". If the specified FTP directory doesn't exist, the FTP server closes the connection.

If ROOTDIRx is set you must also specify the FTP **DRIVEx** entry.

Maximum ROOTDIRx path string length: 64 characters

Important notice:

To avoid security leaks you should define one "normal" user with a directory below the "\" root directory. A user with the ROOTDIR setting "\" has access to every subdirectory on every drive. If the ROOTDIR is set to a subdirectory below the "\", the file access is restricted to the defined subdirectory and its subfolders on the specified drive **DRIVEx**.

Related Topics

FTP user's **DRIVE**

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.01	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[FTP] CONNECTIONS=Number of FTP server connections

Configure how many concurrent FTP connections the FTP server can handle. Default on SC1x3/SC2x are 2 connection. Default on SC1x is 1 connection. Maximum connections are 5. This setting will affect the TCP/IP memory pool usage amount.

Example:

```
[FTP]
CONNECTIONS=2
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
1.27	1.27	1.27	V1.30	V1.30

Top of list
Index page

[FTP] SSL_ENABLE=0/1

Defines whether the FTP server supports SSL (FTPS) connections. IF SSL_ENABLE=1 the FTP server task is started with SSL support. IF SSL_ENABLE=0 the FTP server task is started without SSL support.

You have to provide a CA certificate filename(s), a local server certificate filename, a private and public key pair filename.

Example:

```
[FTP]
SSL_ENABLE=1
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[FTP]
SSL_VERSION=Select the supported SSL version

Defines what SSL version the SSL FTP server should support.
1 = SSL 3.0
2 = TLS 1.0
3 = SSL 3.0 and TLS 1.0 (default)

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[FTP]
SSL_CACERTx=CA certificate filename

Defines the name of FTP server's CA certificate. SSL_CACERT0 must be the name of the FTP server's root CA certificate. If needed SSL_CACERT1 is the next CA certificate in the certificate chain. SSL_CACERT2 is then the next in the chain and so on. Possible file formats are *.PEM and *.DER.

A maximum of five CA certificate files may be listed with tags SSL_CACERT0 up to SSL_CACERT4. The entry tag search is terminated at the first non-existant tag in this sequence.

Comments

Example:

[FTP]
SSL_CACERT0=ROOTCERT.DER

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[FTP]
SSL_SERVERCERT=File name of the server certificate

Defines the name of FTP server's own local certificate. Possible file formats are *.PEM and *.DER.

Comments

Example:

[FTP]
SSL_SERVERCERT=SRVCERT.DER

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[FTP]
SSL_KEYFILE=Filename of the server key file

Defines the name of FTP server's private and public key file. Possible file formats are *.PEM and *.DER.

Key lengths longer than 4096 bits are not supported.

Comments

Example:

[FTP]
SSL_KEYFILE=PRIVKEY.DER

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[FTP]
SSL_CTRLPERM=Permission of the FTP control connection

Whether or not the FTP server allows or requires authentication on the control channel.

1 = SSL is allowed (default)
2 = SSL is required

Comments

Example:

[FTP]
SSL_CTRLPERM=2

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[FTP]
SSL_DATAPERM=Permission of the FTP data connection

Whether or not the FTP server allows or requires authentication on the data channel.

1 = SSL is allowed (default)
2 = SSL is required

Comments

Example:

[FTP]
SSL_CTRLPERM=2

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[WEB]
ENABLE=0/1

Define if the Web server should be activated.

Comments

Use 0 to disable, 1 to enable.

SC1x Comments

By default, the Web server is enabled.

SC1x3/SC2x Comments

If **SAFEMODE** is off, the Web server is enabled per default. If safe mode is on, the Web server is disabled per default.

Related Topics

BIOS_Server_On_Off() - API to Suspend/Resume system servers

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[WEB]
PRIO=prio

Defines the task priority of the web server task (WEBS).

Allowed value are 2-127. Default is 41.

Example:

```
[WEB]
PRIO=80
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.23	V1.23	V1.23	V1.15	V1.00

Top of list
Index page

[WEB]
MAINPAGE=Name of the main page

Defines the name of Web server's main page. The Web server opens this page if a browser request like `http://192.168.200.4/` is received. Typical names are "main.htm" (default) or "index.htm". The console command **`webstat`** shows the current main page.

Related Topics

Set Web Server **Main** Page API Function

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[WEB]
MAINPAGE_POST=Name of the main page (POST method)

Defines the name of Web server's main page when POST method is used. The Web server opens this page if a browser request like `http://192.168.200.4/` is received. Typical names are "main.htm" or "index.htm". If this entry is not present, the **MAINPAGE** entry will specify the mainpage for both methods (GET and POST). The console command **webstat** shows the current main page.

Related Topics

Set Web Server **Main** Page API Function

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.25	V1.25	V1.25	V1.20	V1.20

Top of list
Index page

[WEB]
TEMPPATH=Name of a temporary Web server path

Defines a temporary path for finding files.
If the Web server cannot find the file in its default directory, it will try to find it in this temporary path.
Pathname should include the drive specification.

Comments

This function allows the Web server to locate HTML files produced on the IPC@CHIP® RAMDISK by application programs.

The maximum string length is 32.

Example:

```
[WEB]
TEMPPATH=C:\web
```

The **webstat** command shows the current temporary path.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[WEB]
DRIVE=Set Web server's disk drive

Set Web server's disk drive.

0: Drive A
1: Drive B
2: Drive C
3: Drive D (SC1x3/SC2x only)
25: Drive Z (SC1x3/SC2x only)

If the drive does not exist, the default drive A will be set.
The console command `webstat` shows the current Web server drive.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V1.10	V1.00

Top of list
Index page

[WEB]
ROOTDIR=Name of the root directory

Defines the name of Web server's root directory. If the directory does not exist, the Web server sets "\" as the default root directory.

The console command `webstat` shows the current root directory.

Comments

Important notice: To avoid security leaks you should define a directory below the "\" directory. If you use "\" as web root directory, everybody can read all your files.

Related Topics

Set Web Server Root Directory API Function

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[WEB]
UPLOAD_PATH=Name of the drive and directory used for file upload

Defines the name of Web server's drive and directory used for http file upload. If the directory does not exist, file uploads will fail.
Example:

```
[WEB]
UPLOAD_PATH=B:\uploads
```

The console command **webstat** shows the current upload path.

Comments

Important notice: To avoid security leaks you should define a directory below the "\" directory. If you use "\" as upload directory, everybody can overwrite all your files.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[WEB]
MAXCGIENTRIES=Maximum number of available CGI entries

Set the maximum number of entries for the Web server (Default: 10)
Range: 2 to 128

The console command **cgistat** shows the current number.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[WEB]
WEBSERVERSTACK=Stack size

Sets the stack size (bytes) for the Web server task. The default and minimal stack size is 2048 Bytes.

Programmers of CGI functions who are using Microsoft C-Compilers with C-Library functions such as `sprintf`, which require a lot of stack space, should increase the stack size to 6144 (6 KBytes).

The maximum value is 65500 bytes.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x

V1.27	V1.27	V1.27	V1.30	V1.30

Top of list
Index page

[WEB]
HTTPPORT=port

Sets the port number of the web server.
Default HTTP port: 80

Example:

```
[WEB]
HTTPPORT=81
```

The console command **webstat** shows the current HTTP port number.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[WEB]
POST_SIZE=size

Sets the size of the post method argument working buffer.
Default and minimum size is 768 byte, maximum is 8192 bytes. The web server allocates this memory for each of the active 4 connection handlers

Example:

```
[WEB]
POST_SIZE=1024
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.20	V1.20	V1.20	n/a	n/a

Top of list
Index page

[WEB]
HEADER_SIZE=size in bytes

Sets the web server maximum header length in bytes. This applies to all lines in the HTTP header, the method line as well as all header fields.

Default header length: 1792 characters
Allowed value range: 256 - 8192

Example:

```
[WEB]
HEADER_SIZE=2000
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.05	V1.00

Top of list
Index page

[WEB]
HEADER_REJECT=0/1

Choose whether or not to reject incoming requests that contain headers longer than the internal buffer can store. If this is set to 0, and the received header is longer than the default or configured maximum header length, the corresponding header will be truncated. If this is set to 1 (default) the request will be rejected.

Example:

```
[WEB]
HEADER_REJECT=0
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.05	V1.00

Top of list
Index page

[WEB]
BODY_SIZE=size in bytes

Sets the web server maximum body length in bytes.

Default body length: 3000 bytes
Allowed value range: 256 - 32768

Example:

```
[WEB]
BODY_SIZE=4000
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.23	V1.23

Top of list
Index page

[WEB]
CONNECTIONS=Number of web server HTTP connections

Configure how many concurrent HTTP connections the web server can handle. Default on SC1x3/SC2x are 8 connections. Maximum connections are 30. This setting will heavily affect the TCP/IP memory pool usage amount.

Example:

```
[WEB]
CONNECTIONS=10
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.15	V1.00

Top of list
Index page

[WEB]
TIMEOUT=Seconds for HTTP connection timeout

Configure after which timeout the web server closes an idle connection. Minimum is 1 second. Maximum is 3600 seconds. Default are 60 seconds.

Example:

```
[WEB]
TIMEOUT=10
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[WEB]
HTTPVERSION=HTTP version

Configure which HTTP version the web server will use.
2 = HTTP 1.0 (web server will close the connection)
3 = HTTP 1.1 (web server will leave the connection open)
Default is to use the HTTP 1.1 version.

Example:

```
[WEB]
```

```
HTTPVERSION=2
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[WEB]
DETECT_ENCODING=Detect file encoding

If this option is set to 1 (default: 0), static files will be opened to check their first few bytes for a magic signature. Currently only gzip compressed files are supported. If such a file is detected a special content encoding header (Content-Encoding: gzip) will be sent to the HTTP client. The client then knows that the file needs to be unpacked.
By setting this option you can compress any files before putting them onto your chip without having to rename them and thus without having to adapt links etc.

Example:

```
[WEB]
DETECT_ENCODING=1
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.51	V1.51

Top of list
Index page

[WEB]
UPLOAD=0/1

Define if the Web server file upload should be activated.

Comments

Use 0 to disable, 1 to enable.
By default, the Web server file upload is disabled.

SC1x Comments

A special RTOS variant that supports the web file upload feature is required. In this special version the web file upload feature is always enabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.10	V1.00

Top of list
Index page

[WEB]
UPLOAD_MAX_SIZE=0..2147483647

Define the maximum size for HTTP uploaded files in bytes.

Comments

When the uploaded file exceeds that size it will not be saved. As result the web server returns an server error to the browser. The default max upload size is 2147483647.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[WEB]
USER0=User name for Web Server file upload

Defines a user name for transferring files to the Web server's root directory using the HTTP web file upload feature. The standard user name is 'WEB'.

The console command `webstat` shows the user name and password.

Comments

Important notice: To avoid security leaks you should define a user name and password.

SC1x Comments

A special RTOS variant that supports the web file upload feature is required.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.02	V1.00	V1.00	V1.10	V1.00

Top of list
Index page

[WEB]
PASSWORD0=Password for Web Server file upload

Defines the password used to transfer files to the Web server's root directory using the HTTP web file upload feature. The standard password is 'WEB'.

The console command **webstat** shows the Password and Username.

Comments

Important notice: To avoid security leaks you should define a user name and password.

SC1x Comments

A special RTOS variant that supports the web file upload feature is required.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.02	V1.00	V1.00	V1.10	V1.00

Top of list
Index page

[WEB]
SECURE=Activated the Web security feature for the Web Server

Defines whether the security feature for the Web Server is active or not.

0 = security feature deactivated (default)
1 = security feature activated

Comments

The Web Server security feature allows up to two (SC1x) or ten (SC1x3/SC2x) **paths** to be protected with **user name** and **password**. When this security feature is activated, users must then authenticate themselves to get Web access to these paths.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.10	V1.00	V1.00	V1.40	V1.40

Top of list
Index page

[WEB]
SEC_URLx=Define a path for the security feature

Defines a specified URL for the Web security feature. The user can define SEC_URL0 to SEC_URL1 (SC1x) or SEC_URL0 to SEC_URL9 (SC1x3/SC2x).

All sub URLs of SEC_URLx are then protected by user name and password. The SEC_URL0 path is protected by **SEC_USER0** user name and **SEC_PASSWORD0** password, and SEC_URL1 by **SEC_USER1** user name and **SEC_PASSWORD1** password, and so on.

The maximum length for the paths is 63 characters.

Comments

If the **security** feature is activated, the user should define a path, **user name** and **password**.

In the example below all sub URLs of "[IP]/sec" are protected (e.g. "192.168.200.4/sec/page.htm").

Example:

```
[WEB]
SEC_USER0=otto
SEC_PASSWORD0=web
SEC_URL0=/sec/
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.10	V1.00	V1.00	V1.40	V1.40

Top of list
Index page

[WEB] SEC_USERx=Define a user name for the security feature

Defines a user name for the Web security feature.
The user can define SEC_USER0 to SEC_USER1 (SC1x) or SEC_USER0 to SEC_USER9 (SC1x3/SC2x).
The max length of the user name is 19 characters.

Comments

If the **security** feature is activated, the user should define a **path**, user name and **password**.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.10	V1.00	V1.00	V1.40	V1.40

Top of list
Index page

[WEB] SEC_PASSWORDx=Define a password for the security feature

Defines a password for the Web security feature.
The user can define SEC_PASSWORD0 to SEC_PASSWORD1 (SC1x) or SEC_PASSWORD0 to SEC_PASSWORD9 (SC1x3/SC2x).
The max length of the password is 19 characters.

Comments

If the security feature is activated, the user should define a path, user name and password.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.10	V1.00	V1.00	V1.40	V1.40

Top of list
Index page

[WEB]
SEC_REALMx=Define a realm string for the security feature

Defines a realm string for the Web security feature. The user can define SEC_REALM0 to SEC_REALM1 (SC1x) or SEC_REALM0 to SEC_REALM9 (SC1x3/SC2x).
The max length of the realm string is 259 characters.

Comments

The default strings for are "Secure URL0" for SEC_REALM0 and "Secure URL1" for SEC_REALM1. The other realm strings are empty by default.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.40	V1.40

Top of list
Index page

[WEB]
SSL_ENABLE=Enable the SSL web server

Defines whether the SSL web server is used or not. IF SSL_ENABLE=1 the SSL web server task is started and running. IF SSL_ENABLE=2 the SSL web server task is started but suspended. In this case the user program must enable the server with Suspend/Resume system servers.

You have to provide a CA certificate filename(s), a local server certificate filename, a private and public key pair filename and a optional ephemeral key pair filename.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.20	V1.20

Top of list
Index page

[WEB]
SSL_PRIO=prio

Defines the task priority of the SSL web server task (SSLW).

Allowed value are 2-127. Default is 41.

Example:

```
[WEB]
SSL_PRIO=80
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.15	V1.00

Top of list
Index page

[WEB]
SSL_DRIVE=Set SSL Web server's disk drive

Set SSL Web server's disk drive.

0: Drive A
1: Drive B
2: Drive C
3: Drive D
25: Drive Z

If the drive does not exist, the default drive A will be set.
The console command `webstat` shows the current Web server drive.

If this entry is not present the normal Web server/s disk drive and root directory will be used.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.10	V1.00

Top of list
Index page

[WEB]
SSL_ROOTDIR=Name of the SSL web server's root directory

Defines the name of SSL Web server's root directory. If the directory does not exist, the Web server sets "\" as the default root directory.

The console command `webstat` shows the current root directory.

If the SSL_DRIVE entry is not present the normal Web server/s root directory will be used.

Comments

Important notice: To avoid security leaks you should define a directory below the "\" directory. If you use "\" as web root directory, everybody can read all your files.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.91	V1.00

Top of list
Index page

[WEB] SSL_HTTPPORT=port

Sets the port number of the SSL web server. Default SSL HTTP port: 443

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[WEB] SSL_CONNECTIONS=Number of web server HTTPS connections

Configure how many concurrent HTTPS connections the web server can handle. Default on SC1x3/SC2x are 4 connections. Maximum connections are 30. This setting will heavily affect the TCP/IP memory pool usage amount.

Example:

```
[WEB]
SSL_CONNECTIONS=5
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.15	V1.00

Top of list
Index page

[WEB] SSL_TIMEOUT=Seconds for HTTPS connection timeout

Configure after which timeout the SSL web server closes an idle connection. Minimum is 1 second. Maximum is 3600 seconds. Default are 60 seconds.

Example:

```
[WEB]
SSL_TIMEOUT=10
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[WEB]
SSL_HTTPVERSION=HTTP version

Configure which HTTP version the SSL web server will use.
2 = HTTP 1.0 (SSL web server will close the connection)
3 = HTTP 1.1 (SSL web server will leave the connection open)
Default is to use the HTTP 1.1 version.

Example:

```
[WEB]
SSL_HTTPVERSION=2
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[WEB]
SSL_VERSION=Select the supported SSL version

Defines what SSL version the SSL web server should support.
1 = SSL 3.0
2 = TLS 1.0
3 = SSL 3.0 and TLS 1.0 (default)

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[WEB]
SSL_CACERTx=CA certificate filename

Defines the name of Web server's CA certificate. SSL_CACERT0 must be the name of the web server's root CA certificate. If needed SSL_CACERT1 is the next CA certificate in the certificate chain. SSL_CACERT2 is then the next in the chain and so on. Possible file formats are *.PEM and *.DER.

A maximume of five CA certificate files may be listed with tags SSL_CACERT0 up to SSL_CACERT4. The entry tag search is terminated at the first non-existant tag in this sequence.

Comments

Example:

[WEB]
SSL_CACERT0=ROOTCERT.DER

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.91	V1.00

Top of list
Index page

[WEB]
SSL_CLICACERTx=File name of client's CA certificate

Defines the name of Web server client's CA certificate. If this entry is present, the web server will request a client authentication in his ServerHello message. The server will list all CAs added by these entries in his message.

A maximume of five CA certificate files may be listed with tags SSL_CLICACERT0 up to SSL_CLICACERT4. The entry tag search is terminated at the first non-existant tag in this sequence.

Comments

Example:

[WEB]
SSL_CLICACERT0=CLICA.DER

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.91	V1.00

Top of list
Index page

[WEB]
SSL_SERVERCERT=File name of the server certificate

Defines the name of Web server's own local certificate. Possible file formats are *.PEM and *.DER.

Comments

Example:

[WEB]
SSL_SERVERCERT=SRVCERT.DER

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[WEB]
SSL_KEYFILE=Filename of the server key file

Defines the name of Web server's private and public key file. Possible file formats are *.PEM and *.DER.

Key lengths longer than 4096 bits are not supported.

Comments

Example:

[WEB]
SSL_KEYFILE=PRIVKEY.DER

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[WEB]
SSL_EPHFILE=Filename of the ephemeral key file

Defines the name of Web server's ephemeral private and public key file. Possible file formats are *.PEM and *.DER. When exportable cipher suite is used and the server's certificate is RSA with key length more than 512 bits, we can not use that RSA key pair to exchange keys, because the exportable cipher suite is only allowed to have a maximum key size of 512 bits. Instead we have to use ephemeral RSA keys. To generate the RSA key pair in real-time would have cost an excessive amount of CPU time, that's why this is not supported. In these cases the user must add ephemeral RSA keys, with a maximum size of 512 bits, using this entry.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list

Index page

[SSH]
ENABLE=Enable the SSH server

Defines whether the SSH server is used or not. IF ENABLE=1 the SSH server is started. If you do not provide your own key filename, a default RSA key will be used. You can provide a **RSA Key filename**, and/or **DSA Key filename**.

To enable STDIO on SSH sessions, you have to set **STDIN** and **STDOUT** to SSH.

If **SAFEMODE** is off, the SSH server is enabled per default. If safe mode is on, the SSH server is disabled per default.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[SSH]
SSHPORT=port

Sets the port number of the SSH server.

Default SSH port: 22

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[SSH]
PRIO=prio

Defines the task priority of the SSH server tasks (SSHx) during the session.

Allowed value are 2-127. Default is 11.

Example:

```
[SSH]
PRIO=11
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[SSH]
PRIO_KEX=prio

Defines the task priority of the SSH server tasks (SSHx) during the key exchange.
The key exchange will occupy 100% of the CPU performance for a long time period. Thus the task will have a different task priority during the key exchange.

Allowed value are 2-127. Default is 50.

Example:

```
[SSH]
PRIO_KEX=60
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[SSH]
RSA_KEYFILE=Filename of the SSH server RSA key file

Defines the name of SSH server's private key file, used to authenticate the server to the client.
The file must be in the 'OpenSSH' format.

Key lengths longer than 4096 bits are not supported.

Comments

Example:

[SSH]
RSA_KEYFILE=rsa_key

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[SSH]
DSA_KEYFILE=Filename of the SSH server DSA key file

Defines the name of SSH server's private key file, used to authenticate the server to the client.

The file must be in the 'OpenSSH' format.

Key lengths longer than 4096 bits are not supported.

Comments

Example:

[SSH]
DSA_KEYFILE=dsa_key

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[SSH]
USERx=user

Defines a user name for SSH.

You can define a USER0 and a USER1.
User name and password are case sensitive.
The default user name and password is 'ssh'.
Maximum user name size: 19 characters

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[SSH]
PASSWORDx=password

Define the password for a SSH user.

You can define a PASSWORD0 for USER0 and a PASSWORD1 for USER1.
User name and password are case sensitive.
The default user name and password is 'ssh'.
Maximum password size: 19 characters

Comments

If no password is specified for a user, public key authentication will be the only allowed user authentication method. In this case the section **USERx_KEYFILE** must be provided.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[SSH]
USERx_KEYFILE=File name with user public keys

Defines a filename for SSH USER0 or USER1, that contains one or multiple public keys for user authentication. This is an alternate method for user authentication, beside the password method. It is more secure than the password method.

If the SSH server should allow password and public key authentication, both entries (PASSWORDx and USERx_KEYFILE) must be provided.

The file must be in 'OpenSSH authorized_keys' format. Every key must be listed in one line. Within one key no line breaks are allowed.

Key lengths longer than 4096 bits are not supported.

Comments

Example:

[SSH]
USER0=myself
USER0_KEYFILE=user0_keys

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[SSH]
LOCAL_FORWARD=0/1

Enables/Disables the SSH Local TCP Port Forwarding. By default the Local Forwarding option is disabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[SSH]

REMOTE_FORWARD=0/1

Enables/Disables the SSH Remote TCP Port Forwarding. By default the Remote Forwarding option is disabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[SSH]
FORWARD_ALL=0/1

Allows/Disallows TCP connections to remote forwarded ports from any host. By default only connections from the Local Loopback device are allowed (forwarded over SSH).

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.07	V1.00

Top of list
Index page

[SSH]
REKEY_DATA=Data size before rekeying

Sets the transfered (sent + received) data size, before rekeying is requested. A value of 0 will disable the rekeying by reason of data size.
The default data size is 1073741824 Bytes (1 GB).

Comments

Remember that the SSH client can also request rekeying. Most clients also allow to configure this setting.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.11	V1.00

Top of list
Index page

[SSH]
REKEY_TIME=Time before rekeying

Sets the time in milliseconds, before rekeying is requested. A value of 0 will disable the rekeying by reason of timeout.
The default time is 28800000 ms (3600000*8 = 8 hours).

Comments

Remember that the SSH client can also request rekeying. Most clients also allow to configure this setting.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.11	V1.00

Top of list
Index page

[TFTP]
TFTPPORT=port

Sets the port number of the TFTP server.
Default TFTP port: 69

Example:

```
[TFTP]
TFTPPORT=4000
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
on request	on request	on request	V0.90	V1.00

Top of list
Index page

[TFTP]
PRIO=prio

Defines the task priority of the TFTP server task (TFTD).

Allowed value are 2-127. Default is 41.

Example:

```
[TFTP]
PRIO=80
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
on request	on request	on request	V1.15	V1.00

Top of list
Index page

[TELNET]
ENABLE=0/1

Define if the Telnet server should be activated.

Comments

Use 0 to disable, 1 to enable.

SC1x Comments

By default, the Telnet server is enabled.

SC1x3/SC2x Comments

If SAFEMODE is off, the Telnet server is enabled per default. If safe mode is on, the Telnet server is disabled per default.

Related Topics

BIOS_Server_On_Off() - API to Suspend/Resume system servers

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[TELNET]
PRIO=prio

Defines the task priority of the Telnet server task (TELN).

Allowed value are 2-127. Default is 11.

Example:

```
[TELNET]
PRIO=10
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.23	V1.23	V1.23	V1.15	V1.00

Top of list
Index page

[TELNET]
TELNETPORT=port

Sets the port number of the Telnet server.
Default Telnet port: 23

Example:

```
[TELNET]
TELNETPORT=5000
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[TELNET]
TIMEOUT=Telnet timeout minutes

Telnet session will automatically close after TIMEOUT minutes without any characters received from the client. A TIMEOUT setting of zero means no timeout.

Default Value: `TIMEOUT=0` (no timeout)

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[TELNET]
LOGINDELAY=0/1

Define if the delayed login of the Telnet server should be activated.

Comments

Use 0 to deactivate, 1 to activate.
By default, the delayed login is enabled.
The delay time starts with 400 milliseconds.
After each following failed login the delay time will be doubled until it reached 20 seconds.
After successful login the delay time will be set back to 400 milliseconds.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x

V1.01	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[TELNET]
LOGINRETRIES=number of login retries

Defines the number of login retries until Telnet session will be closed.

Example:

```
[TELNET]
LOGINRETRIES=3
```

Comments

The default value is 5.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.01	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[TELNET]
USERx=user

Defines a user name for Telnet.

Comments

You can define a USER0 and a USER1.
Default user is 'tel', password is 'tel'.
You must specify both the user name and their password.
Neither user name nor password are case sensitive.

Maximum user name size: 19 characters

Important notice: To avoid security leaks you must define both user names and passwords.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[TELNET]

PASSWORDx=password

Define the password for a Telnet user

Comments

You can define a PASSWORD0 for USER0 and a PASSWORD1 for USER1.
Default user is 'tel', password is 'tel'.
Neither user name nor password are case sensitive.

Maximum password size: 19 characters

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[DEVICE]
FILESHARING=0/1

Disable/Enable the file sharing. See also BIOS_Get_File_Sharing_Mode() and BIOS_Set_File_Sharing_Mode() API functions.

Comments

0=disable (default), 1=enable

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.02	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[DEVICE]
POWERSAVE=1/2/4

Set the used power save mode. See also the BIOS_Power_Save() and BIOS_Power_SaveEx() API.

Comments

1=run with full speed (default), 2=run with 1/2 clock, 4=run with 1/4 clock

SC1x3/SC2x Comments

If POWERSAVE=4, the internal Ethernet controller runs fixed in 10MBit, Half duplex mode.

This entry also influences the USB API. If POWERSAVE=2 the USB API cannot be used at all. The SC1x3/SC2x will crash if you do. If POWERSAVE=4 the USB API can only be used in Device mode, because Host mode needs better performance.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	V1.00	V1.00	V1.06	V1.00

Top of list
Index page

[DEVICE]
PFI_ENABLE=0/1

Set to 1 to enable PIO9 as Power Fail Interrupt (PFI) pin so that the system recognizes this interrupt. The PFI pin is checked during boot. The RTOS boot process will be suspended as long as the PFI signal is high. This assures that the system only starts with a valid supply voltage.

Default is 0, which indicates to not use the PFI pin.

Comments

As of @CHIP-RTOS-x86 version 1.53 the Power Loss Protected (PLP) disk drives will by default be write protected at the Power Fail Interrupt. This default behavior can be changed with the PFI_PLP_WR_EN CHIP.INI entry.

WARNING: Don't enable this pin if you don't have a power fail circuit connected to the PFI pin. Otherwise the IPC@CHIP® will not boot and the @CHIP-RTOS-x86 will not come up anymore, because the PIO9 pin, if unconnected, is normally high (input with pullup).

Related Topics

hal_install_isr() installs handler
pfe_enable_int() alternative way to enable PFI

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[DEVICE]
PFI_PLP_WR_EN=A: B: D:

List here the Power Loss Protected (PLP) disk drives which should remain write enabled after a Power Fail Interrupt (PFI). By default, all PLP drives will be prevented from performing any further writes after a Power Fail Interrupt.

Comments

This example below will configure the system so that after a PFI, the A: drive and B: drive will remain write enabled even when these drives have been **formatted** with the PLP option.

```
[DEVICE]
PFI_PLP_WR_EN= A: B:
```

WARNING: When this entry is used to allow PLP drives to write after the Power Fail Interrupt, the user must assure that sufficient power remains to cover the complete time spent writing to the disk. Otherwise disk corruption could result due to the drive violating the PLP **condition** that only the current sector being written can be altered. The **hal_install_pfaildone_callback()** API may help in determining this required time.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.53	V1.53

Top of list
Index page

[DEVICE] NAME=name

Define the name of this device.

Comments

This name will show up with the 'Chiptool' software when the network is scanned.

Maximum name size: 20 characters

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[DEVICE] SAFEMODE=0/1

Disable (0=default) / Enable (1) the CHIP.INI safe mode.

Comments

If the safe mode is disabled, then all internal services, like FTP server, Web server, Telnet server, UDP config server, SSH server etc. are enabled with default settings. This allows an easy startup with the IPC@CHIP®. Services must be disabled explicitly.

If the user enables the safe mode setting, all internal services are disabled per default. Services

must be enabled explicitly. This mode is useful when preparing for a release. It also assures that after an update to a future RTOS release, all possible new services that this release implements are disabled per default.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.30	V1.30

Top of list
Index page

[TRACE] FLASHWRITE=0/1

Trace the activity of flash writes for debug purposes.

Comments

Set to 1 to enable the debug output. On STDOUT a message is printed at every call to the low-level physical flash write routine.

By default the debug output is disabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.10	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[TRACE] INTNOTSUPP=0/1

Trace the activity of calls to not supported interrupts or functions.

Comments

Set to 0 to disable the debug output. On STDOUT a message is printed at every call to a not supported interrupt vector or not supported function.

By default the debug output is enabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.10	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[SERIAL]
EXT_DMA=0/1

Disable/enable DMA receive mode on the serial port 0 (EXT). If DMA receive mode is disabled, the serial port 0 (EXT) works with the standard serial interrupt.
The recommended mode is the DMA receive mode. It is only necessary to disable the DMA receive mode for the serial port 0 (EXT) if the DMA channel is needed by an external device. See all possible **DMA configurations**. In the IRQ receive mode, you may loose characters if the system gets lots of interrupts (e.g. network) or if you are writing to the flash disk (file system calls).
See documentation of the **Hardware API**.

Example which disables DMA receive mode on the serial port 0 (EXT).

```
[SERIAL]
EXT_DMA=0
```

By default, DMA receive mode is enabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	n/a

Top of list
Index page

[SERIAL]
COM_DMA=0/1

Disable/enable DMA receive mode on the serial port 1 (COM). If DMA receive mode is disabled, the serial port 1 (COM) works with the standard serial interrupt.
The recommended mode is the DMA receive mode. It is only necessary to disable the DMA receive mode for the serial port 1 (COM) if the DMA channel is needed by an external device.
See all possible **DMA configurations**. In the IRQ receive mode, you may loose characters if the system gets lots of interrupts (e.g. network) or if you are writing to the flash disk (file system calls).
See documentation of the **Hardware API**.

Example which disables DMA receive mode on the serial port 1.

```
[SERIAL]
COM_DMA=0
```

By default, DMA receive mode is enabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[SERIAL]
SER2_DMA=0/1

Disable/enable DMA receive mode on the serial port 2. If DMA receive mode is disabled, the serial port 2 works with the standard serial interrupt.
The recommended mode is the DMA receive mode. It is only necessary to disable the DMA receive mode for the serial port 2 if the DMA channel is needed by an external device. See all possible **DMA configurations**. In the IRQ receive mode, you may loose characters if the system gets lots of interrupts (e.g. network) or if you are writing to the flash disk (file system calls).
See documentation of the **Hardware API**.

Example which disables DMA receive mode on the serial port 2.

```
[SERIAL]
SER2_DMA=0
```

By default, DMA receive mode is enabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[SERIAL]
SER3_DMA=0/1

Disable/enable DMA receive mode on the serial port 3. If DMA receive mode is disabled, the serial port 3 works with the standard serial interrupt.
The recommended mode is the DMA receive mode. It is only necessary to disable the DMA receive mode for the serial port 3 if the DMA channel is needed by an external device. See all possible **DMA configurations**. In the IRQ receive mode, you may loose characters if the system gets lots of interrupts (e.g. network) or if you are writing to the flash disk (file system calls).
See documentation of the **Hardware API**.

Example which disables DMA receive mode on the serial port 3.

```
[SERIAL]
SER3_DMA=0
```

By default, DMA receive mode is enabled.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[SERIAL]
SEND_DMA=0/1/2

Selects the DMA send mode for a serial port.

SC12/SC1x3:
Use 0 to enable the DMA send mode for the serial port 0 (EXT).

Use 1 to enable the DMA send mode for the serial port 1 (COM).

SC2x:

Use 1 to enable the DMA send mode for the serial port 1 (COM).

SC11/SC13:

Use 0 to enable the DMA send mode for the serial port 0 (EXT).
Use 1 to enable the DMA send mode for the serial port 1 (COM).
Use 2 to enable the DMA send mode for both ports.

Example which enables the DMA send mode on the serial port 1 (COM).

```
[SERIAL]
SEND_DMA=1
```

By default, DMA send is disabled.

SC12/SC1x3 Comments

Important:
By default both useable DMA channels for the serial ports EXT and COM are used for receiving characters. If you want to use the DMA send mode for one serial port (e.g. the EXT port), the second port (e.g. the COM port) automatically switches over to using the serial port's receiver interrupt.

Possible DMA configurations:

COM_DMA	EXT_DMA	SEND_DMA		DMA1	DMA0
0	0	not present		DRQ1	DRQ0
0	1*	not present		DRQ1	RECV EXT
1*	0	not present		RECV COM	DRQ0
1*	1*	not present		RECV COM	RECV EXT
don't care	0	0		SEND EXT	DRQ0
don't care	1*	0		SEND EXT	RECV EXT
0	don't care	1		DRQ1	SEND COM
1*	don't care	1		RECV COM	SEND COM

1* = Value 1 is default so the entry must not be present

SC2x Comments

Important:
By default both the DMA channel for the serial port COM is used for receiving characters.

Possible DMA configurations:

COM_DMA	SEND_DMA		DMA1	DMA0
0	not present		not used	not used
1*	not present		RECV COM	not used
0	1		not used	SEND COM
1*	1		RECV COM	SEND COM

1* = Value 1 is default so the entry must not be present

SC11/SC13 Comments

Important:
The IPC@CHIP®'s SC11, SC13 have four DMA channels, which are available for the serial ports EXT and COM. By default two DMA channels are used for receiving characters from the COM

and EXT ports.

Possible DMA configurations:

COM_DMA	EXT_DMA	SEND_DMA		DMA3	DMA2	DMA1	DMA0
0	0	not present		not used	not used	DRQ1	DRQ0
0	1*	not present		not used	RECV EXT	DRQ1	DRQ0
1*	0	not present		RECV COM	not used	DRQ1	DRQ0
1*	1*	not present		RECV COM	RECV EXT	DRQ1	DRQ0
0	0	0		SEND EXT	not used	DRQ1	DRQ0
0	0	1		not used	SEND COM	DRQ1	DRQ0
0	1*	0		SEND EXT	RECV EXT	DRQ1	DRQ0
0	1*	1		not used	RECV EXT	DRQ1	SEND COM
1*	0	0		RECV COM	not used	SEND EXT	DRQ0
1*	0	1		RECV COM	SEND COM	DRQ1	DRQ0
1*	1*	0		RECV COM	RECV EXT	SEND EXT	DRQ0
1*	1*	1		RECV COM	RECV EXT	DRQ1	SEND COM
0	0	2		SEND EXT	SEND COM	DRQ1	DRQ0
0	1*	2		SEND EXT	RECV EXT	DRQ1	SEND COM
1*	0	2		RECV COM	SEND COM	SEND EXT	DRQ0
1*	1*	2		RECV COM	RECV EXT	SEND EXT	SEND COM

1* = Value 1 is default so the entry must not be present

SC1x Comments

Note that **RS485** is not available with serial send DMA.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.02	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[SERIAL]
SEND_DMA2=0/1

Selects the DMA send mode for a serial port.

Comments

Use 0 to enable the DMA send mode for the serial port 2.
Use 1 to enable the DMA send mode for the serial port 3.

Example which enables the DMA send mode on the serial port 3.

```
[SERIAL]
SEND_DMA2=1
```

By default, DMA send is disabled.

SC1x3/SC2x Comments

Important:

By default the two DMA channels for port 2 and 3 are used for receiving characters from the ports 2 and 3. If you want to use the DMA send mode for one serial port (e.g. the port 2), the second port (e.g. the port 3) automatically switches over to using the serial port's receiver interrupt.

Possible DMA configurations:

SER3_DMA	SER2_DMA	SEND_DMA2		DMA3	DMA2
0	0	not present		DRQ3	DRQ2
0	1*	not present		DRQ3	RECV port2
1*	0	not present		RECV port3	DRQ2
1*	1*	not present		RECV port3	RECV port2
don't care	0	0		SEND port2	DRQ2
don't care	1*	0		SEND port2	RECV port2
0	don't care	1		DRQ3	SEND port3
1*	don't care	1		RECV port3	SEND port3

1* = Value 1 is default so the entry must not be present

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[SERIAL] EXT_RECVQUEUE=size

Sets the receive queue size of the EXT port. Minimum size is 1024.
Maximum size is 10240 byte.

Comments

Example:

```
[SERIAL]
EXT_RECVQUEUE=2048
```

By default, the receive queue size is 1024 byte (SC1x and SC1x3/SC2x before RTOS V1.35) or 2048 byte (SC1x3/SC2x since RTOS V1.35).
If it is planned to use PPP server or PPP client the size should set to 4096.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	n/a

Top of list
Index page

[SERIAL] EXT_SENDQUEUE=size

Sets the send queue size of the EXT port. Minimum size is 1024.

Maximum size is 10240 byte.

Comments

Example:
```
[SERIAL]
EXT_SENDQUEUE=2048
```

By default, the send queue size is 1024 byte (SC1x and SC1x3/SC2x before RTOS V1.35) or 2048 byte (SC1x3/SC2x since RTOS V1.35).
If it is planned to use PPP server or PPP client the size should set to 4096.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	n/a

Top of list
Index page

[SERIAL]
COM_RECVQUEUE=size

Sets the receive queue size of the COM port. Minimum size is 1024.
Maximum size is 10240 byte.

Comments

Example:
```
[SERIAL]
COM_RECVQUEUE=2048
```

By default, the receive queue size is 1024 byte (SC1x and SC1x3/SC2x before RTOS V1.35) or 2048 byte (SC1x3/SC2x since RTOS V1.35).
If it is planned to use PPP server or PPP client the size should be set to 4096.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[SERIAL]
COM_SENDQUEUE=size

Sets the send queue size of the COM port. Minimum size is 1024.
Maximum size is 10240 byte.

Comments

Example:

```
[SERIAL]
COM_SENDQUEUE=2048
```

By default, the send queue size is 1024 byte (SC1x and SC1x3/SC2x before RTOS V1.35) or 2048 byte (SC1x3/SC2x since RTOS V1.35).
If it is planned to use PPP server or PPP client the size should be set to 4096.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[SERIAL]
SER2_RECVQUEUE=size

Sets the receive queue size of the serial port2. Minimum size is 1024.
Maximum size is 10240 byte.

Comments

Example:

```
[SERIAL]
SER2_RECVQUEUE=2048
```

By default, the receive queue size is 1024 byte (SC1x and SC1x3/SC2x before RTOS V1.35) or 2048 byte (SC1x3/SC2x since RTOS V1.35).
If it is planned to use PPP server or PPP client the size should be set to 4096.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[SERIAL]
SER2_SENDQUEUE=size

Sets the send queue size of the serial port 2. Minimum size is 1024.
Maximum size is 10240 byte.

Comments

Example:

```
[SERIAL]
SER2_SENDQUEUE=2048
```

By default, the send queue size is 1024 byte (SC1x and SC1x3/SC2x before RTOS V1.35) or 2048 byte (SC1x3/SC2x since RTOS V1.35).
If it is planned to use PPP server or PPP client the size should be set to 4096.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[SERIAL]
SER3_RECVQUEUE=size

Sets the receive queue size of the serial port 3. Minimum size is 1024.
Maximum size is 10240 byte.

Comments

Example:

```
[SERIAL]
SER3_RECVQUEUE=2048
```

By default, the receive queue size is 1024 byte (SC1x and SC1x3/SC2x before RTOS V1.35) or 2048 byte (SC1x3/SC2x since RTOS V1.35).
If it is planned to use PPP server or PPP client the size should be set to 4096.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[SERIAL]
SER3_SENDQUEUE=size

Sets the send queue size of the serial port 3. Minimum size is 1024.
Maximum size is 10240 byte.

Comments

Example:

```
[SERIAL]
SER3_SENDQUEUE=2048
```

By default, the send queue size is 1024 byte (SC1x and SC1x3/SC2x before RTOS V1.35) or 2048 byte (SC1x3/SC2x since RTOS V1.35).
If it is planned to use PPP server or PPP client the size should be set to 4096.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V0.90	V1.00

Top of list
Index page

[SERIAL]
COM_BAUD=BAUD Rate

Sets the BAUD rate of the COM port.

Comments

Example:

```
[SERIAL]
COM_BAUD=9600
```

By default, the BAUD rate of the COM port is 19200 (with 8 data bits, no parity, 1 stop bit).

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[SERIAL]
EXT_BAUD=BAUD Rate

Sets the BAUD rate of the EXT port.

Comments

Example:

```
[SERIAL]
EXT_BAUD=9600
```

By default, the BAUD rate of the EXT port is 19200 (with 8 data bits, no parity, 1 stop bit).

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	n/a

Top of list
Index page

[DOSLOADER]
MEMGAP=Paragraphs

Sets a memory gap between the loaded DOS programs as a memory reserve.

Comments

Some programs compiled with Borland C 5.02, Beck Paradigm Edition (other compilers??) try to increase their program memory block at runtime. This can occur, for example, when opening a file with Borland/Paradigm C-library function `fopen`, where some additional memory is required. The C-library `fopen` function calls int 21h **0x4A**, which is not directly visible to the application programmer. This memory resize call fails if another program is loaded after the previous one, because now there is no memory space left for increasing the memory size of the previously executed program. The program then returns from `fopen` with an error. In this case, the global program variable errno is set to value 8 (not enough memory).

To prevent this error, the @CHIP-RTOS allows a memory gap of a defined size between loaded programs. This memory gap size is specified as a number of paragraphs(where 1 paragraph equals 16 bytes for SC1x, and 256 bytes for SC1x3/SC2x systems).
This strategy can fail when programs are terminated and restart again.

Example:

```
[DOSLOADER]
MEMGAP=128
```

By default, MEMGAP is set to 0. The maximum value is 2048 paragraphs, where any value larger than this is truncated to 2048

SC1x3/SC2x Comments

The **BeckHeap library** provides a less rigid alternative solution to this memory problem.

Related Topics

BIOS_Set_Memory_Gap() API function

Developer Notes

It is not necessary to set this entry if the application doesn't show the described error. Only if a C-library function call sets errno to 8, should this value be defined. We recommend in that case a value of 128 paragraphs (2048 Bytes). The described problem was noticed when the Borland C-library function `fopen` was used. The same can happen with usage of C-library function `malloc` using memory model Large. The `malloc` returns a NULL pointer in this case.

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.02	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[BATCH]
BATCHMODE=0/1

Sets the **batch file** execution mode of programs.

`BATCHMODE=0` : (Default mode)

The programs listed in the batch file will be **executed concurrently** , starting one after

another, without waiting for completion (or going resident) of the predecessor program. The only exceptions are the **WAIT** and **REBOOT** commands.

`BATCHMODE=1` :

The listed programs will be **executed sequentially** , one at time (similar to DOS). The execution of the successor program will be delayed until the current program either finishes, terminates resident by calling DOS Interrupt **21h Service 0x31** or makes the @CHIP-RTOS **BIOS_Batch_Continue()** batch file wakeup API.

The maximum delay time for execution of the next listed program in the batch file is 15 seconds, unless this limit has been deactivated with the EXECTIMEOUT=0 **configuration** control.

Important:

If `BATCHMODE=1` take care that every program in your batch file which has a successor program either exits **(int21h 0x4C)** or terminates resident with **int21h 0x31**. A program which runs forever should call from the main function @CHIP-RTOS **BIOS_Batch_Continue()** API, which immediately enables the further batch file sequencing.

Related Topics

BATCHMODE command
BIOS_Set_Batch_Mode() - Run-time batch mode selection API

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[BATCH] EXECTIMEOUT=0/1

Disable (=0) / enable (=1) the **batch file** program execution delay time limit for BATCHMODE=1.
`EXECTIMEOUT=0` :

The successor program in a batch file waits forever if the predecessor program neither finishes nor calls **BIOS_Batch_Continue()** API.

`EXECTIMEOUT=1` : (Default mode)

The maximum delay time for execution of the next listed program in a batch file is 15 seconds.

Comments

This Boolean control applies only to BATCHMODE=1.

Example:

```
[BATCH]
EXECTIMEOUT=0
```

Related Topics

BATCHMODE Configuration

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
V1.00	V1.00	V1.00	V0.90	V1.00

Top of list
Index page

[USB] MAX_POWER_SUPPLY=Maximum power supplied via USB

Use this entry to tell the RTOS USB-driver how much power can be supplied to a bus-powered device in Host mode. The value is given in mA. The default value is 500. This means that if you don't specify a value here, the RTOS USB-driver assumes that your hardware is able to supply up to 500 mA to a USB device.

Example:

```
[USB]
MAX_POWER_SUPPLY=100
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.06	V1.00

Top of list
Index page

[USB] BULK_PER_FRAME=Maximum number of BULK transfers

This entry defines the maximum number of transfers that will be performed for each BULK endpoint per frame. A higher value increases the bandwidth of each BULK endpoint. If you for example set this entry to 8, a maximum of 500 kb/sec can be transfered on an endpoint with a maximum packet size of 64 (8 * 64 bytes per millisecond).
However a higher value also increases the number and duration of USB interrupts. This may lead to decreasing performance.

Valid values are between 1 and 12. The default value is 6.

Example:

```
[USB]
BULK_PER_FRAME=8
```

Supported since or modified in @CHIP-RTOS version

SC12	SC13	SC11	SC1x3	SC2x
n/a	n/a	n/a	V1.20	V1.20

Top of list
Index page

End of document