



easyToWeb-AVR

module with ATmega128[®] and CS8900A
Starter-Guide (Hardware-Version 1.31)

Forschungs- und Transferzentrum Leipzig e.V.
Wächterstr. 13
D-04107 Leipzig
Germany
phone +49 341 3076 1136
fax +49 341 3076 1220
info@easyToWeb.net
www.easyToWeb.net

In this manual are descriptions for copyrighted products that are not explicitly indicated as such. The absence of the trademark (™) and copyright (©) symbols does not imply that a product is not protected. Additionally, registered patents and trademarks are similarly not expressly indicated in this manual.

The information in this document has been carefully checked and is believed to be entirely reliable. However, FTZ Leipzig assumes no responsibility for any inaccuracies. FTZ Leipzig neither gives any guarantee nor accepts any liability whatsoever for consequential damages resulting from the use of this manual or its associated product. FTZ Leipzig reserves the right to alter the information contained herein without prior notification and accepts no responsibility for any damages which might result.

Additionally, FTZ Leipzig offers no guarantee nor accepts any liability for damages arising from the improper usage or improper installation of the hardware or software. FTZ Leipzig further reserves the right to alter the layout and/or design of the hardware without prior notification and accepts no liability for doing so.

© Copyright 2002 Forschungs- und Transferzentrum Leipzig e.V. an der HTWK Leipzig (FH). Rights - including those of translation, reprint, broadcast, photomechanical or similar reproduction and storage or processing in computer systems, in whole or in part - are reserved. No reproduction may occur without the express written consent from FTZ Leipzig

Forschungs- und Transferzentrum Leipzig e.V.
Udo Jakobza / Thomas Minner
Wächterstr.13
D-04107 Leipzig
Germany
phone +49(0)341-3076-1136
fax +49(0)341-3076-1220
e-mail: info@easyToWeb.net
<http://www.easyToWeb.net>

Table of contents

History	4
As-delivered conditions and important remarks	5
Steps for the activation.....	5
Connection of connector	5
Settings at the DC adapter.....	6
Parameter of Prog-AVR14	6
Hardwareeinstellungen:	7
Editor Settings of CodeVisionAVR-IDE	7
Parameter of Terminal-Programm	7
After the setting of the voltage	8
LED-signals	8
Messages of Terminal-Program	8
Debug-Modi	9
Debug-modes	9
Software debug-interface	9
Terminal-Commands	10
Overview of terminal-commands	10
Settings for the Ethernet-interface	11
Tools to start the Ethernet-interface.....	12
easyToWeb-Software and CodeVisionAVR	15
Parameter of C-Compiler and Programmer-Interface	16

History

- Version 1.2a
 - 13.09.2002: first release
 - 19.09.2002: some corrections and additional figures
 - 20.09.2002: additional notes to opening the ethernet interface
- Version 1.2b
 - 26.09.2002: some new setting to the Prog adapter + some corrections
 - 27.09.2002: additional explanations to the settings of CodeVisionAVR Project
- Version 1.3a
 - 29.01.2003: supplementation and upgrading to the "easyToWeb-AVR"-version 1.3
 - 30.01.2003: description of new software debug-commands
 - 27.02.2003: first release of english version
- Version 1.41
 - 08.05.2006: revision for software 2.xx and hardware 1.31

As-delivered conditions and important remarks

- **Hardware-Version 1.31**
- Software-Version 2.xx;
- All external links are equiped;
- The **ISP_Switch (S1)** on the "easyToWeb-AVR" module is ON;
- The microcontroller is programmed;
- Software: Sample-Projekt eTW_AVR_HTTP (see \samples\eTW_AVR_http)
 - **IP-Adresse:** 192.168.0.2;
 - **GATEWAY-Adresse:** 192.168.0.100;
 - **SUBMASK-Adresse:** 255.255.255.0;
 - The software is compiled with following debug-settings;
 - #define DEBUG // enable all global debug functions, see debug.c
 - #define WEB_DEBUG // enable web debug functions, see web_debug.c
 - #define WEB_DEBUG_CS8900 // enable CS8900 debug functions, see cs8900.c
 - #define WEB_DEBUG_BUFFER // enable BUFFER debug functions, see buffer.c
 - #define WEB_DEBUG_ETHERNET // enable ETHERNET debug functions, see ethernet.c
 - #define WEB_DEBUG_ARP // enable ARP debug functions, see arp.c
 - #define WEB_DEBUG_IP // enable IP debug functions, see ip.c
 - #define WEB_DEBUG_ICMP // enable ICMP debug functions, see icmp.c
 - #define WEB_DEBUG_TCP // enable TCP debug functions, see tcp.c
 - #define WEB_DEBUG_TCP_OFFLINE // enable TCP debug functions, see tcp_debug.c
 - #define WEB_DEBUG_HTTP // enable HTTP debug functions, see http_server.c
 - see also project.h in the directory \samples\eTW_AVR_http;
- **MAC-address:** is programmed in the PROM-memory of the DS2430A;
- Source-Code works with 1.24.3b und **1.24.6** of CodeVisionAVR C-Compiler;
- **REMARK:** The most actual informations on the software are to be found in the file "changelog.txt".

Steps for the activation

Connection of connector

- *Debug-interface*
connect link **DEBUG_AVR128** (JP1) on the "easyToWeb-AVR" module and the prog-adapter;
switch OFF **ISP_Switch (S1)** from the "easyToWeb-AVR" module;
- *serial interface*
link **UART0** (JP5) with the enclosed adapter (10 pin on SUBD9) and the SUBD9-cord extension cable with the PC ;
- *Voltage supply:*
link main power DC transformer (2,1mm jack plug; 8 .. 12V and 200mA) with the plug P1 on "easyToWeb-AVR" module;
- *Ethernet:*
connect Ethernetcable to socket JP9;

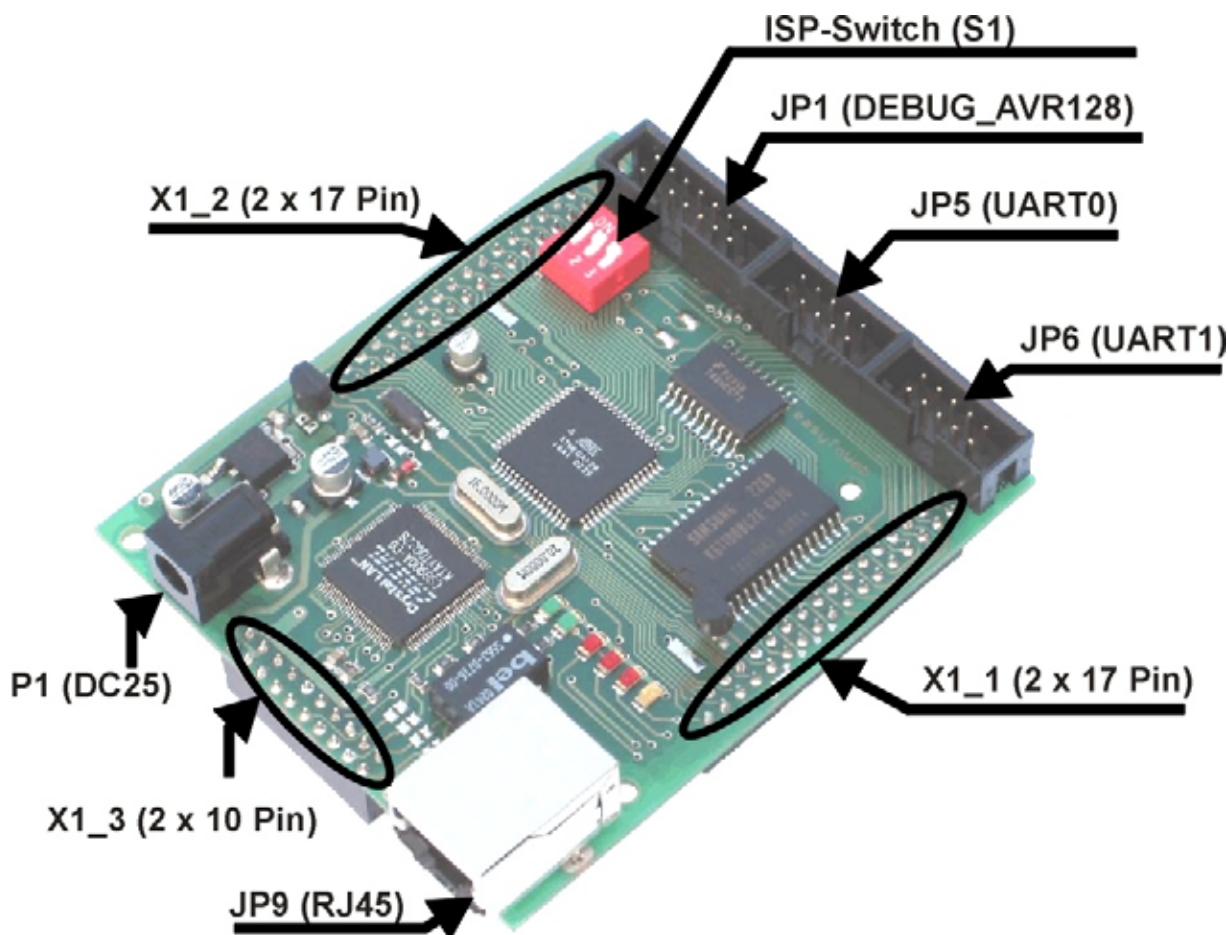


Figure 1: Position of connections of the "easyToWeb-AVR" module

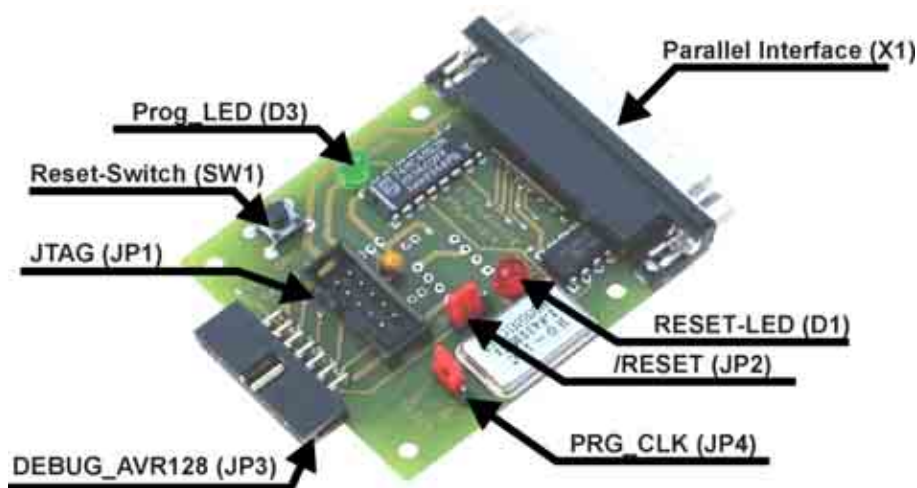


Figure 2: Position of connections of the Prog-Adapter

Settings at the DC adapter

The right voltage (9V) should be set up and the passing jack plug-socket (2.1mm) should be chosen at the DC adapter. The polarity must **not** absolutely be respected.

Parameter of Prog-AVR14

Hardwareeinstellungen:

- PRG_CLK (JP4) **OFF** (see Hardware Guide of AVR-Prog14);
- /RESET (JP2) **ON** (very important);

The programming of the ATmega128 takes place directly from the CodeVisionAVR-development environment.

Therefore, you should proceed to the settings under <Settings><Programmer> as figure 3 shows.

REMARK: The "Delay Multiplier" ensures the decrease of the download-speed (fast computers >1Ghz) in case of problems with the programming interface.

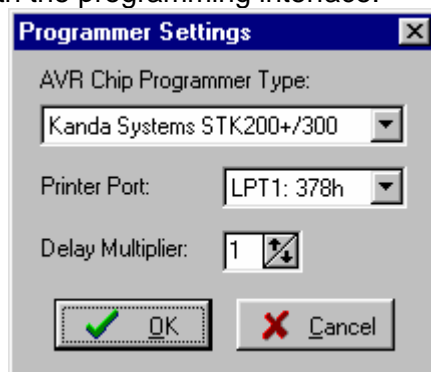


Figure 3: Programmer settings of CodeVisionAVR-IDE

Editor Settings of CodeVisionAVR-IDE

The correct display of the source codes needs some adjustments of the CodeVisionAVR-IDE. You have to change the "Tab size" to 4 signs (Default is 8). The settings in figure 4 are to be done under <Settings><Editor>.

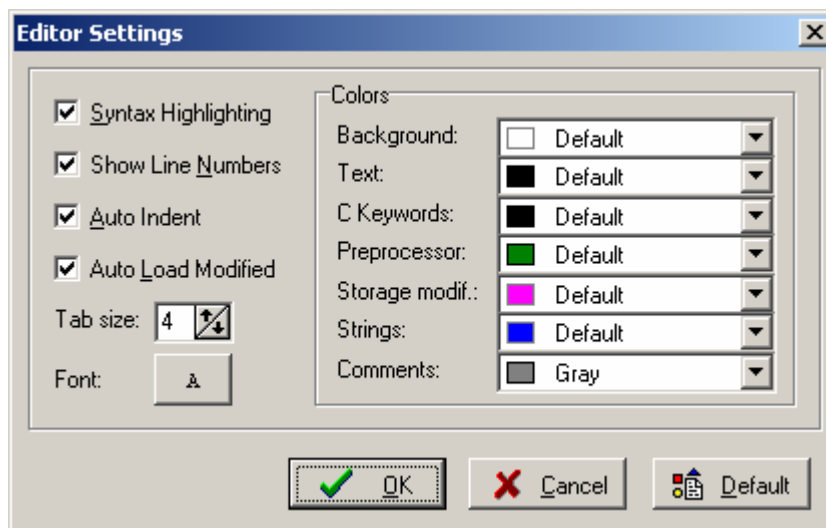


Bild 4 Dialog of Editor Settings from the CodeVisionAVR-IDE

Parameter of Terminal-Programm

It is advisable, to use the terminal of the CodeVisionAVR for the first steps. The settings in figure 5 are to be done under <Settings><Terminal>.

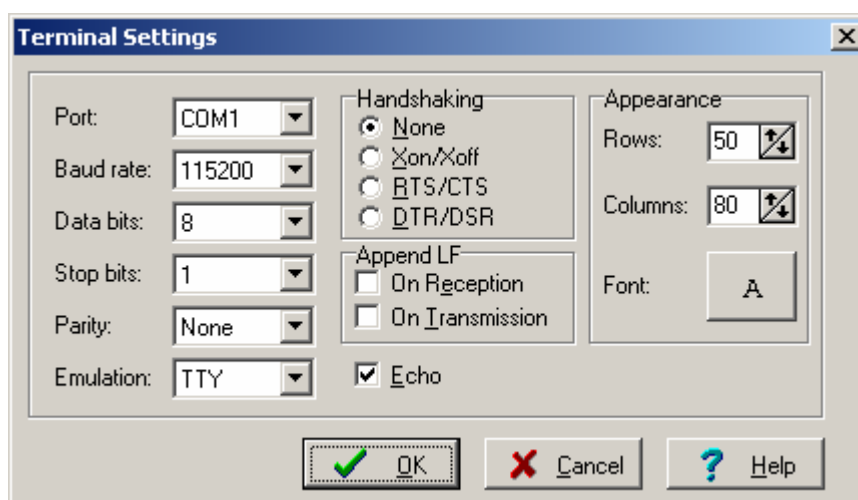


Figure 5: Terminal settings

In case of use of a different terminal, the same settings are to be made.

After the setting of the voltage

LED-signals

After the setting of the voltage, the LEDs signal the actual modus.

The pre-installed "easyToWeb"-Software defines following functions for the LED1 to LED4.

- **LED1 - LED4:** are activated in the start phase of the "easyToWeb"-software and go on for a few milliseconds. These go off, when the main loop is reached. (File: "etw_mc.c"; "http_main.c").
- **LED1 (yellow):** on-and-off-flashing of the LED in the loop of the main-Routine (File: "http_main.c")
- **LED2 (red):** is switched with the access to the implemented HTTP-server (File: "easyweb.c") in the function " easyweb_HTTPServer()".
- **LED3 (red):** this LED can be used for personal measurements
- **LED4 (red):**
PortD.7 is used for the DS2430 and is not available for personal uses, otherwise as 1Wire-Bus;

The LANLED and LINKLED are controlled from Ethernetcontroller and signal the function modus of the CS8900A.

- **LANLED (green):** signals sending, receiving and the identification of collisions
- **LINKLED (green):** is activated with each reception of a valid 10Base-T Link-Puls.

Messages of Terminal-Program

- Display of the actual software version with date and time
- Display of the CodeVisionAVR-version, which has been used to compile the software;
- Microcontroller-type and quartz-frequency;
- The actual reset-source;
- Information on Ethernetcontroller CS8900A;
- Display of the actual datas of DS2430A (1Wire-Bus);
- Display MAC-adress;
- Display the IP-, GATEWAY- and SUBMASK addresses;
- Display the options of easyToWeb-Software;


```

CodeVisionAVR - eTW_120_16Mhz_http_debug.prj - [Terminal]
File Edit Project Tools Settings windows Help
Disconnect Hex Code Send Bx File Tx File Help Clear
CodeVisionAVR
Project: eTW_120_16Mhz
Notes
  http_main.c
  device.c
  cs8900.c
  buffer.c
  measure.c
  ethernet.c
  esp.c
  scnp.c
  p.c
  tcp.c
  tcp_debug.c
  cs8900_debug.c
  web_debug.c
  server.c
  http_server.c
  Mf.c
  eth_mc.c
  Other Files
  project.h
project: CS8900-HTTP-Server Version 1.01
easyToWeb: Sep  6 2004/15:15:17/Version 2.04
ATmega128 (L):16.000MHz
CodeVisionAVR-Version:1243
AVR-RESET: External,
CS8900A-Rev D (0x0e630009); 5.0Volt
1-WIRE-DEVICE:
(0)BIT07..00 - Family Code : 0x14 - DS2430A
(0)BIT56..08 - Serial Number: 0x00000179710b
(0)BIT63..57 - CRC : 0xc7
MAC-Address from DS2430-PROM:
MAC : 00:50:c2:21:10:bd
IP, SUBMASK and GATEWAY from EEPROM:
IP : 192.168.0.2
SUBMASK: 255.255.255.0
GATEWAY: 192.168.0.100
---- easyToWeb-Software ----
DEVICE_CS8900
CS8900_MEMORY
ETHERNET_ENABLE
ARP_ENABLE
IP_ENABLE
ICMP_ENABLE
TCP_ENABLE
HTTP_ENABLE
HTTP-Server OK
COM1: 115200,8N1 No handsh. ASCII TTY Echo on
Messages
Insert

```

Figure 6: Messages in the terminal program

The "easyToWeb-AVR" module is ready with all correct informations, see figure 6.

Debug-Modi

Die easyToWeb-Software erlaubt die Kompilierung mit verschiedenen Debug-Modi. Die Aktivierung bzw. Deaktivierung der *Debug-Modi* erfolgt in der Datei "project.h" durch einfaches Auskommentieren der entsprechenden Macros, siehe "eTW-Software_debug_commands_V10.pdf".

Debug-modes

The software to "easyToWeb-AVR" -module allows the compilation with different debug modes. The activation respectively desactivation of the *Debug-Mode* takes place in the file "project.h" The functions of the software debug interface are available with the macros "**DEBUG**" and "**WEB_DEBUG**".

For an optimal exploitation of the resources and a maximal execution speed, you can disable the macros "**DEBUG**", "**WEB_DEBUG**", "**WEB_DEBUG_CS8900**" and so on.

Software debug-interface

In the as-delivered conditions, the software is compiled with debug-settings. Therefore are many possibilities for the parameterization of the software **without** new compilation available to the user. More informations are displayed during the runtime. Figure 7 shows an extract at the access to the stored webpage (File: "websiteside.h") with a browser. You have to write "debug tcp" inside of a terminal.

```

CodeVisionAVR - e:\TW_128_16MHz_http_debug.prj - [Terminal]
Project: e:\TW_128_16MHz
Notes
  http_main.c
  device.c
  csi900.c
  buffer.c
  measure.c
  ethernet.c
  asp.c
  scmp.c
  p.c
  tcp.c
  tcp_debug.c
  csi900_debug.c
  web_debug.c
  server.c
  http_server.c
  Mf.c
  shv_mc.c
Other Files
  project.h

TCP(0,0) State-M: LISTEN
TCP(0,1) Open: 80
TCP(0,1) Read: SYN, - SEQ:1018756943,ACK:0,data:0Byte
TCP(0,1) Option: 8Byte 02 04 05 b4 01 01 04 02
TCP(0,1) Write-Ctrl: SYN,ACK, - SEQ:119128397,ACK:1018756944
TCP(0,1) State-P: SYN_REC'D
TCP(0,2) Read: ACK, - SEQ:1018756944,ACK:119128398,data:0Byte
TCP(0,2) SOCK_CONNECTED
TCP(0,2) State-P: ESTABLISHED
TCP(0,3) Read: PSH,ACK, - SEQ:1018756944,ACK:119128398,data:468Byte
TCP(0,3) Write-Ctrl: ACK, - SEQ:119128398,ACK:1018757412
TCP(0,3) Write: Data, ACK, PSH - SEQ:119128398,ACK:1018757412,data:1000Byte
TCP(0,4) Read: ACK, - SEQ:1018757412,ACK:119129398,data:0Byte
TCP(0,4) Write: Data, ACK, PSH - SEQ:119129398,ACK:1018757412,data:1000Byte
TCP(0,5) Read: ACK, - SEQ:1018757412,ACK:119130398,data:0Byte
TCP(0,5) Write: Data, ACK, PSH - SEQ:119130398,ACK:1018757412,data:1000Byte
TCP(0,6) Read: ACK, - SEQ:1018757412,ACK:119131398,data:0Byte
TCP(0,6) Write: Data, ACK, PSH - SEQ:119131398,ACK:1018757412,data:258Byte
TCP(0,7) Read: ACK, - SEQ:1018757412,ACK:119131656,data:0Byte
TCP(0,7) Write-Ctrl: FIN,ACK, - SEQ:119131656,ACK:1018757412
TCP(0,7) State-M: FIN_WAIT_1
TCP(0,8) Read: ACK, - SEQ:1018757412,ACK:119131657,data:0Byte
TCP(0,8) State-P: FIN_WAIT_2
TCP(0,9) Read: FIN,ACK, - SEQ:1018757412,ACK:119131657,data:0Byte
TCP(0,9) Write-Ctrl: ACK, - SEQ:119131657,ACK:1018757413
TCP(0,9) State-P: TIME_WAIT
TCP(0,9) FIN Timeout (State: TIME_WAIT)
TCP(0,9) SOCK_DISCONNECT
TCP(0,9) State-M: CLOSED
TCP(0,9) Socket Error: tcp_timeout
TCP(0,x) Passive Reopen: 80
TCP(0,9) State-M: LISTEN
COM1: 115200,8N1 No handsh. ASCII TTY Echo on
Messages
Insert

```

Figure 7: Runtime messages during HTTP-access of internal webside

Terminal-Commands

After setting the "Terminal Settings" (see figure 5), the terminal is available for commands-input. Each command that is described in „overview on terminal-commands“ are to be concluded with the **enter-key**. Each following pressure of the **enter-key** repeats the last command. Already typed-in signs can be deleted with the **backspace-key**.

CAUTION: The delete takes place at the display of the terminal and in the microcontroller.

REMARKS: All commands are defined in small letters (see file "web_debug.c"). They are generally case sensitive and need at times additional parameters, see following overview. Commands that aren't fulfilled are noticed with an error-message.

Overview of terminal-commands

- 1.) **info**
 - description:* displays the start-screen - displays all important informations
 - syntax:* info
 - short form:* -
 - indication:* -
 - example:* info

- 2.) **ip**
 - description:* inputs a new IP address
 - syntax:* ip <xxx.xxx.xxx.xxx>
 - short form:* -
 - indication:* -
 - example:* ip 192.168.0.2

- 3.) **submask**
description: inputs a new SUBMASK address
syntax: submask <xxx.xxx.xxx.xxx>
short form: submask
indication: -
example: submask 255.255.254.0
- 4.) **gateway**
description: inputs a new GATEWAY address
syntax: gateway <xxx.xxx.xxx.xxx>
short form: gateway
indication: -
example: gateway 192.168.0.200
- 5.) **reset**
description: software reset of hardware (ATmega128/32)
syntax: reset
short form: -
indication: -
example: reset
- 6.) **debug**
description: activates / deactivates the display of debug informations
syntax: debug <parameter>
short form: -
indication: -
example: debug tcp

REMARK: Some commands have special abbreviations. In case you define proper commands, you must take care not to use the same names or abbreviations.

Settings for the Ethernet-interface

The good functioning of the "easyToWeb-AVR" webserver requires some settings. The terminal-interface is used for this aim, as described above.

REMARK: The "easyToWeb-AVR" webserver supports the BootP (dynamic IP configuration). A fixed IP-address can be given.

The terminal-commands **ip** <>, **submask** <> and **gateway** <> are used for the entry, see example figure 8.

The screenshot shows a CodeVisionAVR terminal window with a project named 'eTW_120_16Mhz'. The terminal output displays the following configuration details:

```

?? : ip 192.168.0.2
old IP: 192.168.0.2
new IP: 192.168.0.2
old eeprom IP: 192.168.0.2
new eeprom IP: 192.168.0.2
?? : submask 255.255.255.0
old SUBMASK: 255.255.255.0
new SUBMASK: 255.255.255.0
old eeprom SUBMASK: 255.255.255.0
new eeprom SUBMASK: 255.255.255.0
?? : gateway 192.168.0.100
old GATEWAY: 192.168.0.100
new GATEWAY: 192.168.0.100
old eeprom GATEWAY: 192.168.0.100
new eeprom GATEWAY: 192.168.0.100

```

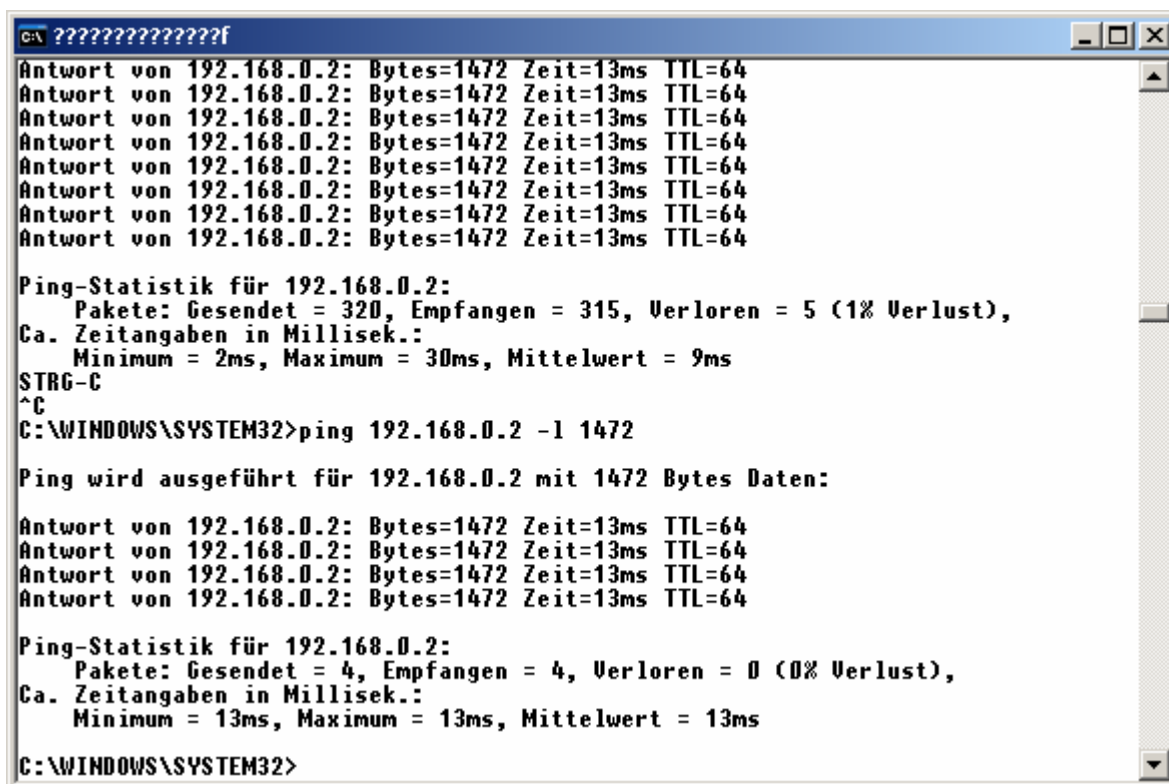
The terminal window also shows a file explorer on the left with a project structure including files like 'http_main.c', 'device.c', 'cpu800.c', 'buffer.c', 'measure.c', 'ethernet.c', 'app.c', 'scnp.c', 'p.c', 'top.c', 'top_debug.c', 'cpu800_debug.c', 'web_debug.c', 'server.c', 'http_server.c', 'MII.c', 'eth_mc.c', and 'project.h'. The terminal status bar at the bottom indicates 'COM1: 115200,8N1' and 'No handsh.'.

Figure 8: Entry of the new internet-addresses

CAUTION: The "easyToWeb-AVR" webserver has a 10Mbit T-Base connector. A 10Mbit Hub or Switch with 10Mbit input or a 10Mbit network-card (the actual and also cheaper network-cards supports an 10/100Mbit-networks) must therefore be available.

Tools to start the Ethernet-interface

Different tools are available for the steps of putting into operation or debugging. The first test is easy to realise with the Ping-command in a DOS-window (ICMP Internet-Protocoll).



```

C:\ ??????????????????
Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64
Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64
Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64
Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64
Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64
Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64
Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64

Ping-Statistik für 192.168.0.2:
    Pakete: Gesendet = 320, Empfangen = 315, Verloren = 5 (1% Verlust),
Ca. Zeitangaben in Millisek.:
    Minimum = 2ms, Maximum = 30ms, Mittelwert = 9ms
STRG-C
^C
C:\WINDOWS\SYSTEM32>ping 192.168.0.2 -l 1472

Ping wird ausgeführt für 192.168.0.2 mit 1472 Bytes Daten:

Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64
Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64
Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64
Antwort von 192.168.0.2: Bytes=1472 Zeit=13ms TTL=64

Ping-Statistik für 192.168.0.2:
    Pakete: Gesendet = 4, Empfangen = 4, Verloren = 0 (0% Verlust),
Ca. Zeitangaben in Millisek.:
    Minimum = 13ms, Maximum = 13ms, Mittelwert = 13ms

C:\WINDOWS\SYSTEM32>

```

Figure 9: Ping-command from the DOS-window

REMARK: The activities of the "easyToWeb-AVR" webserver can be very well followed over the Terminal-program, see figure 7.

With the good functioning of the PING-command, the first read-in of the internal webpage (websiteside.h) should be without error, figure 10.

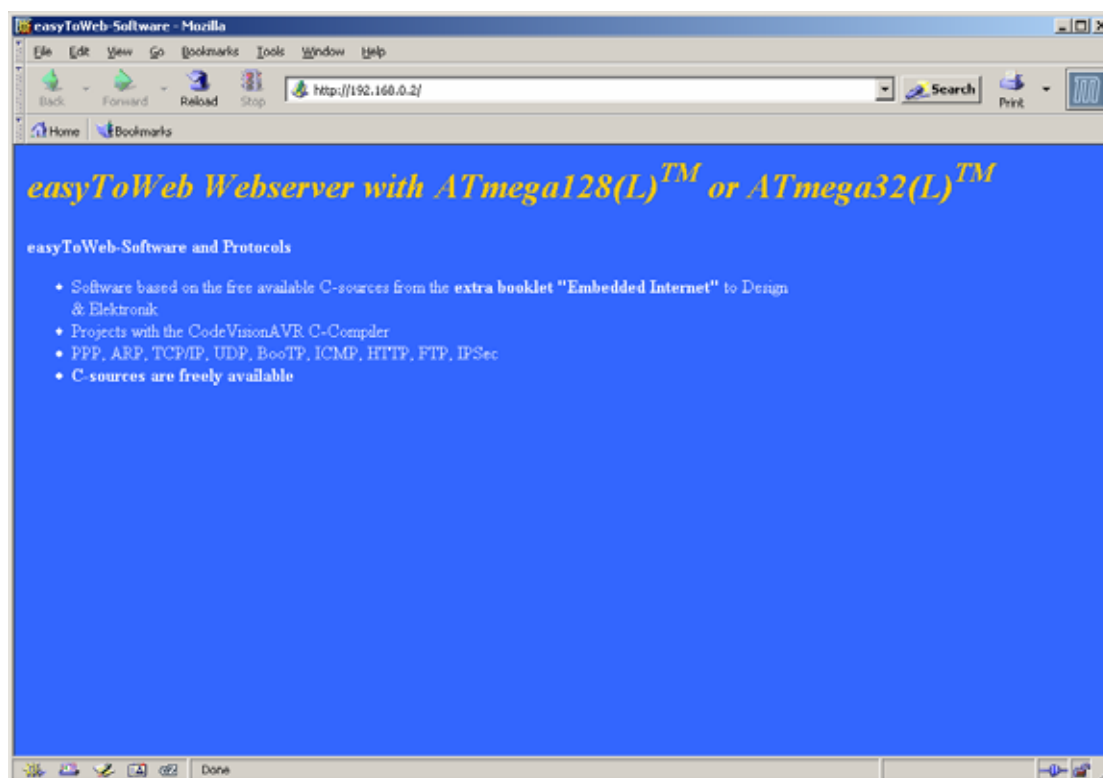


Figure 10: The stored webpage (websiteside.h) in the Mozilla-browser

REMARK: We recommend "Mozilla" as from 1.7, "Firefox" as from 1.5 or "Internet Explorer" as from 5.5 to display the webpages.

The debugging, directly after the modification of the source-code, can easily be realised with the free tool "Ethereal". The version 0.99.0. is in the CD-ROM under the directory <ethernet_tools>. The most actual versions for almost every operating systems are to be found under the address "www.ethereal.com".

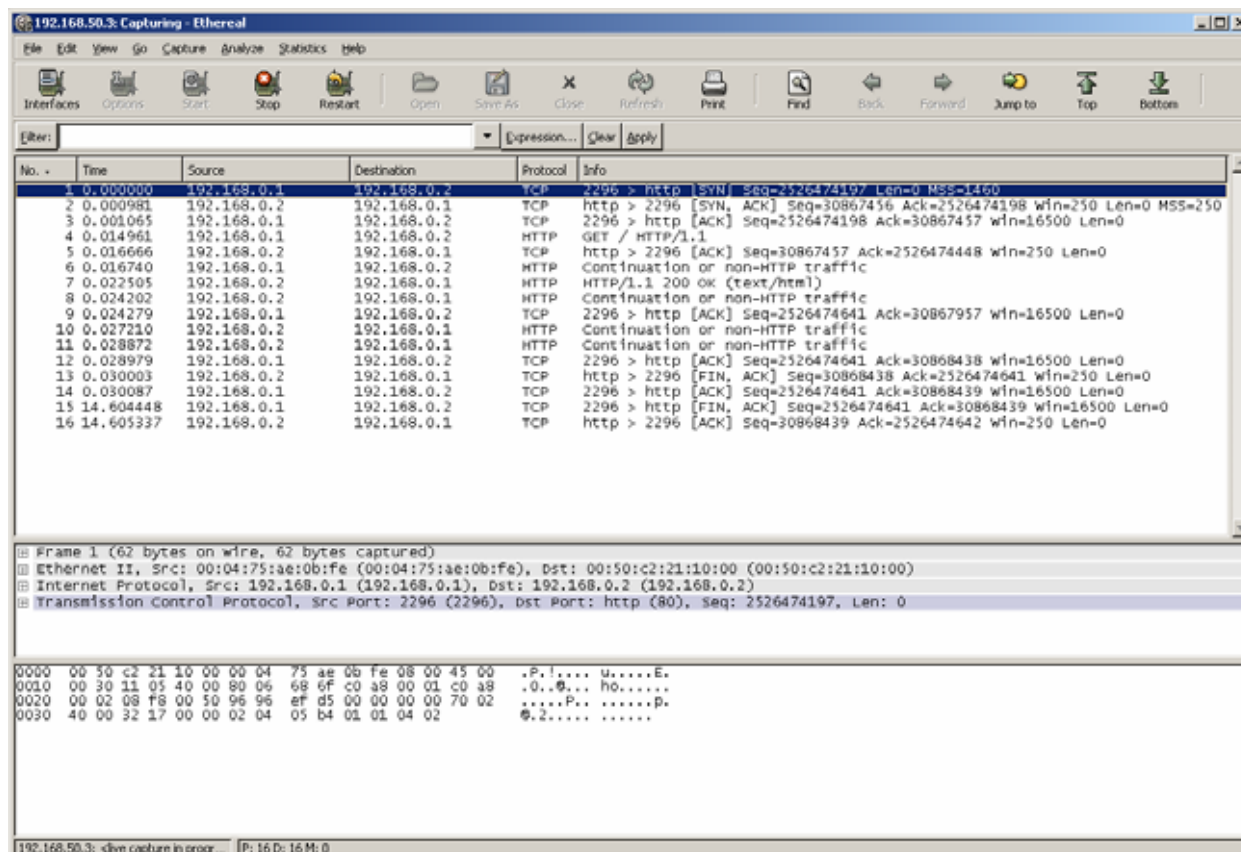


Figure 11: Ethereal with a TCP/IP-connection (Source-IP: 192.168.0.2; Dest-IP: 192.168.0.1)

easyToWeb-Software and CodeVisionAVR

The complete source-code for the Embedded Webserver have been stored under the directory <easyToWeb-Software Version 2.xx>. Beside of the C-code, the directory also contains the projects for the CodeVisionAVR.

The project-file <eTW_128_16Mhz_http_debug> in CodeVisionAVR must be opened for a new compilation of the c-sources, figure 12. The project "eTW_128_16Mhz_http_lib" permits the compilation with the libraries.

CAUTION: The booklet on the ATmega128 should be read for a better function.

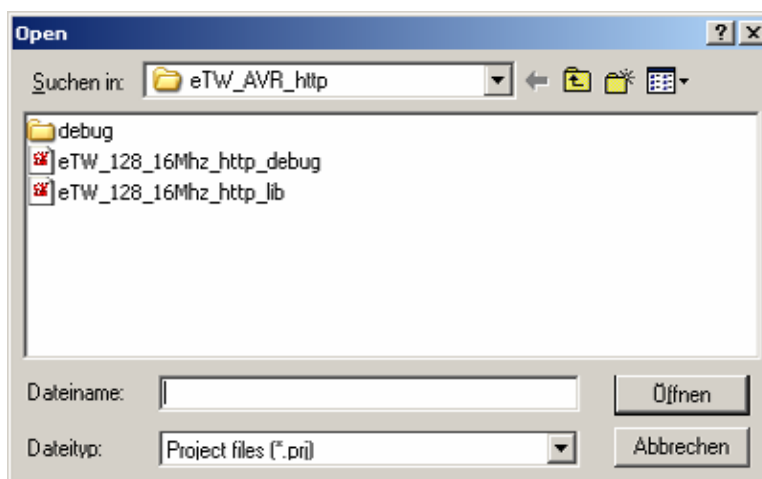


Figure 12: <File><Open> Dialogbox of CodeVisionAVR

All relevant settings for the compilation of the c-sources are loaded after the opening of the project. The necessary files belong to it, figure 13.

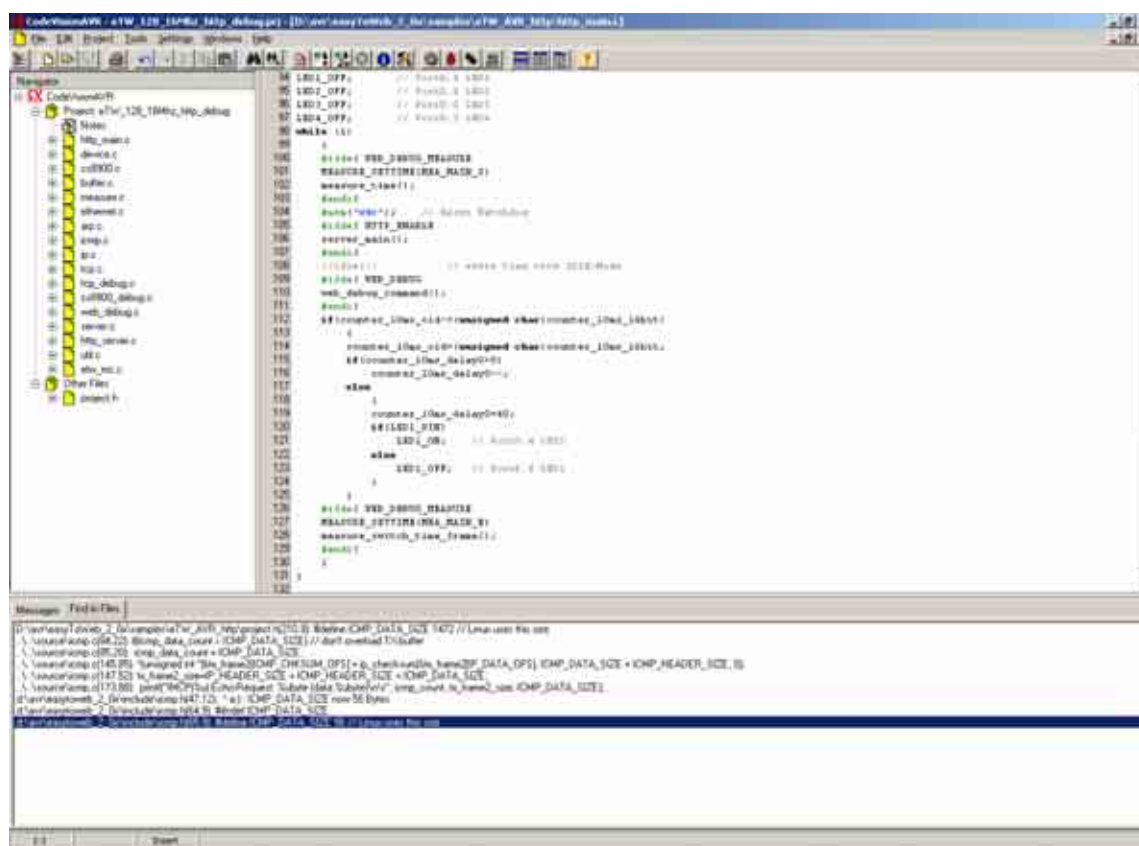


Figure 13: C-source-files of the easyToWeb-Software

Parameter of C-Compiler and Programmer-Interface

The most important project-parameters are to be found under the menu <Project><Configure>. The single settings are detailly decrbed in the booklet or online-help of the CodeVisionAVR-IDE. The most relevant parameters for the easyToWeb-project are revealed in figure 14.

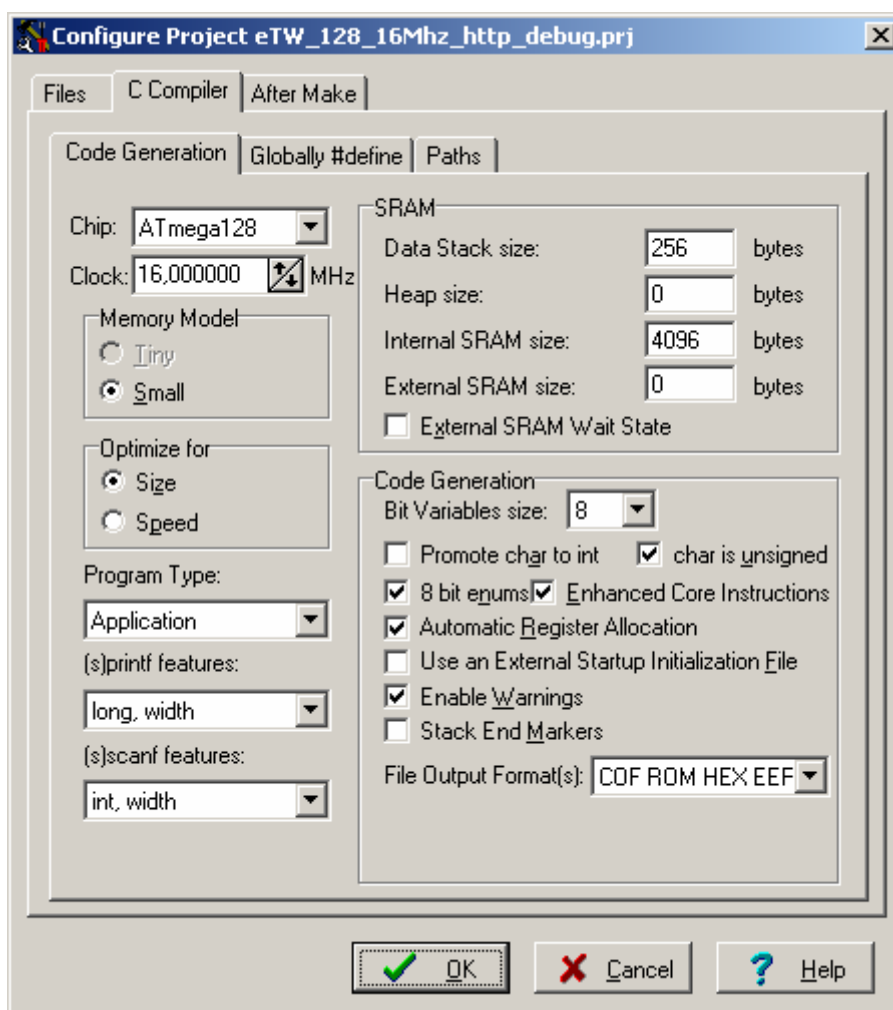


Figure 14: Settings for the C-compiler

Changes should be cautiously performed under professional help for e.g. CodeVisionAVR online-help.

The settings to program the microcontroller are shown in figure 15. The settings of the "Fuse Bits" are particularly relevant for a certain functioning of the "easyToWeb-AVR" module. Proper experiments should occur only under the support of the data sheet of the ATmega128.

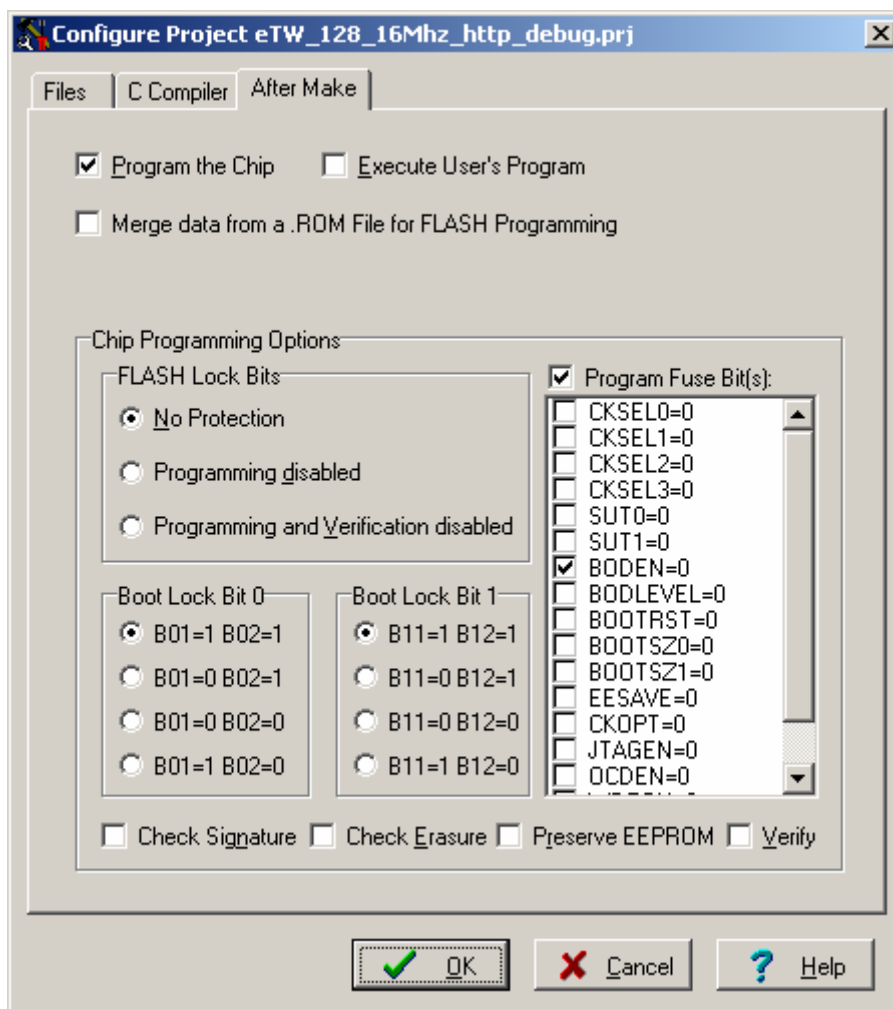


Figure 15: Microcontroller-settings of C-Compiler