Micromint Modules



www.micromint.com

Getting Started With the Micro64

1.0 Software Installation

1.1 Installing the CodeVisionAVR C Compiler

1. Open the CodeVisionAVR Demo folder on the CD.



2. Click on setup icon and the window similar to the one below should open.



3. Click the Next button and the following window should open.



4. Accept the terms in the license agreement.

5. Click the Next button and the following window should open.

🗆 Pin 1



6. Click the Next button and the following window should open.



7. Click the Next button and the following window should open.

🔂 CodeVisionA	VR C Compiler Evaluation V1.24.5 - InstallShield Wizard 💦 🔀
Setup Type Choose the se	tup type that best suits your needs.
Please select . Typical	a setup type. All program features will be installed. (Requires the most disk space.)
Custom	Choose which program features you want installed and where they will be installed. Recommended for advanced users.
	<pre> < Back Next > Cancel</pre>

8. Click the Next button and the following window should open.



- 1.2 Installing the Micro64's Boot Loader Software
 - 1. Open the Boot Loader folder on the CD.



2. Click on setup icon and the window similar to the one below should open.



3. Wait for the files to finish loading and the following window should appear.

9. Click the Install button and the following window should open.



10. Wait for the program to install then press the Finish button.



4. Click on the OK button to continue with the install and the following window should open.

Micro64/128 ____

🛃 Micro64_Micro128 Bootloader Setup	
Begin the installation by clicking the button below.	
Click this button to install Micro64_Micro128 Bootloader software to the specified destination directory.	
Directory: C:\Program Files\Micro64_Micro128 Bootloader\hange Directory	
Exit Setup	



5. Click on the button and the following window should open.

岩 Micro64_Micro128 Bootloader - Choose Program 🔀
Setup will add items to the group shown in the Program Group box. You can enter a new group name or select one from the Existing Groups list.
Program Group: Micromint Development Tools
Existing Groups:
Accessories MCS Electronics
Microminit Development Tools Microminit Des Programs Microsoft Web Publishing Startup
Continue

6. Click on the Continue button and files should start loading into your computer. After the files are done loading the following window should open.

Micro64_Micro128 Bootloader Setup
Micro64_Micro128 Bootloader Setup was completed successfully.
ОК

7. Click OK to finish the installation.

2.0 Compiling and running HelloWorld.c

When the compiler is opened for the first time the GUI (Graphical User Interface) should look like the following image.

🗱 CodeVisionAVR
File Edit Project Tools Settings Windows Help
<u> </u>
Navigator Code Templates C · ·
CodeVisionAVR No Project Other Files
Messages
Insert

Please follow the following steps to setup the compiler for a Micro64/128.

1. Click on "File|New" menu option or click the toolbar button and the following window will be displayed.

🕂 Create Ne	w File 🛛 🔀
File Type	
	oK]
⊂ <u>P</u> roject	🗙 <u>C</u> ancel

2. Select "Project" and press "OK" and the following window will be displayed.

Confirm	n 🗵
2	You are about to create a new project. Do you want to use the CodeWizardAVR?
	<u>Yes</u> <u>N</u> o

3. For simplicity we will not use the CodeWizardAVR so click "No" and the following window should open.

Create New Pro	oject	? 🛛
Save in:	🔁 EXAMPLES 💽 🗢 🖻 📅 📰 •	
My Recent Documents Desktop My Documents My Computer	ADC8535 AVR134 C_ASM D51920 D51990 EEPROM KEYPAD LCDCHAR LCDDEMO LED MAX1241 MULTFLE SPI THERM75 THERM75	
My Network Places	File name: Image: Save as type: Project files (*, pri)	Save Cancel

4. Type in a file name of your choice and click "Save". For demonstration purposes HelloWorld was chosen for the File name. The following screen should appear.

😽 Configure Project HelloWorld.prj	
Files C Compiler After Make	
C:\cvavreval\EXAMPLES\HelloWorld.prj	- ≓i <u>A</u> dd
	t Remove
	TI Edit File Name
	↔ Movelin
	- Mous Down
	▲ Move Down
1	
🔽 <u>D</u> K 🕺 <u>C</u> and	cel ? <u>H</u> elp

5. Click on the "C Compiler" tab and the window will look similar to the following.

Configure Project Hellow	Vorld. prj	×
Code Generation Messages Chip: ATtiny13 Clock: 4.000000 MHz Memory Model: Tiny	Globally #define Paths SRAM Data Stack Size: 16 bytes Heap Size: 0 bytes Internal SRAM size: 64 bytes External SRAM size: 0 bytes	
Optimize for: Size (s)printf Features: int, width (s)scanf Features: int, width	External SRAM Wait State Code Generation Bit Variables size: 16 Promote char to int Image: Char is unsigned Image: State Bit State Image: State Bit State Image: State Bit State Image: State Bit State File Output Format(s): COF RDM HEX EEP	
	<u>V</u> _DK X Cancel ? Help	

- Click the arrow on select Atmega64 for Micro64 or Atmega128 for Micro128.
- Click next to the first digit in the following Clock: 4.000000 MHz and change it to 11.0592.
- 8. After you complete the changes the window should look like the following for the Micro64/128.



- 9. All other parameters do not need to be changed in order to compile a simple program. Please refer to the CodeVisionAVR's help section for further details on the other parameters. Click on the "Files" tab to continue.
- 10. The next step would be to add a source file. To add a

source file click the <u>A</u>dd button and the following window will appear.

Add File To Pro	oject			? 🛛
Look in:	C EXAMPLES	•	- 🗈 💣 🎫	
My Recent Documents Desktop My Documents	AUC8535 AVR134 C_ASM D51820 EEPROM KEYPAD LCDCHAR LCDCHAR LCDCHAR MULTFILE SPI THERM75 THERM75	D HelloWorld.c		
My Network Places	File name:		•	Open
	Files of type:	C Compiler files (*.c)	_	Cancel

11. Find and Select "HelloWorld.c" and click "Open". HelloWorld.C can be found on the CD or on Micromint's Micro64/128 Datasheet & Application Notes webpage. http://www.micromint.com/app_notes/micro64_128. htm The following window should appear.

©2004, Micromint, Inc. ATmega64 and AVR are trademarks of Atmel, CodeVisionAVR is trademark of HP InfoTech, BASCOM-AVR is trademark of MCS Electronics

	—
HelloWorld.c	Add 👔
	±1 <u>R</u> emove
	T. Edit File Nam
	↔ Move <u>U</u> p

- 12. Click the "OK" button and then you will be ready to make the hex file.
- 13. To make the hex file for the program click on "Project|Make" menu option or click the 🔀 tool

bar button. The following window should open.



14. Click "OK". Now it is time to program the Micro64. using the Boot Loader software. If you already installed the Boot Loader software in section 1 then click on "START|Programs| Micromint Development Tools | Micro64 Micro128

REV 1.0 May 11, 2005

Bootloader" and the following window will open. The boot loader uses USART1 to download programs to the Micro64/128. If you are using the development board then please refer to section 6.31 of the Micro64/128 datasheet to set-up the jumpers properly.

Micro64/Micro128 Bootlo	ader - V 1.0	
ile Default All Settings Info	_	
Micro64 AVIT [®] Based Controller	Connection Com Port Selection:	Micro128 AVR®Based Controller
Status: Idle		
Application Code		Province
Application code (*		DIOWSE
EEPROM Code		Browse
	Download	
		,
Info		E <u>x</u> it

15. Click the Browse button next to "Application Code" to select the hex file that was created by the compiler. A similar window like the following window will open.

Select File to D	ownload		? 🗙
Look in:	C EXAMPLES	▼ ← 1 → 1	
My Recent Documents Desktop My Documents	ADC8535 AVR134 C_ASM D51820 EEPROM LCDCHAR LCDCHAR LCDCHAR MULTFILE SPI	C THERM75 THERMLCD I HelloWorld	
My Network Places	File name: Files of type:	Image: Weight of the second	en ncel

16. Find and select the file "HelloWorld.hex" and click "Open".

<u>D</u>ownload

- 17. Click on the _____ button to send the file through COM1 to the Micro64.
 NOTE: The Boot Loader software uses the COM Port when it is sending a file to the Micro64. If the software says it is unable to open the COM port there can be a few reasons why.
 - 1. The COM port selected is already in use. Close the application using the COM port or select another one.
 - 2. The Micro64/128 is constantly sending information to the closed COM port. Hold the RESET button or disconnect the power the Micro64/128 then click on the Download button.
- Press and release the RESET button on the Micro64 development board or cycle the power to the module. After the program is done downloading the file the following message box will open.

Micro64/Micro128 Bootloader - V 1.0 🔀				
Flash File successfully downloaded.				
ок				

19. Click "OK" and open HyperTerminal with the following settings.

COM1 Properties	? 🛛
Port Settings	
Bits per second:	9600
Data bits:	8
Parity:	None
Stop bits:	1
Flow control:	None
	Restore Defaults
0	IK Cancel Apply

20. Press and release the RESET button or cycle the power. Wait for a moment and you should see "Hello World" constantly displaying on the screen. That is all there is to getting a program up and running.

Micro64/128 _____

3.0 HelloWorld.c Listing

/*************************************	
<pre>#include <mega64.h> // Comment this line out for Micro128 // For Micro128 make sure that you goto "Project Configure", to the // "C Compiler" tab and change the chip to ATmega128 //#include <mega128.h> // Uncomment this line for Micro128 #include <stdio.h> // Standard library</stdio.h></mega128.h></mega64.h></pre>	I/O
<pre>// Declare your global variables here #define RXB8 1 #define TXB8 0 #define UPE 2 #define OVR 3 #define FE 4 #define UDRE 5 #define RXC 7 #define TXC 6</pre>	
<pre>#define FRAMING_ERROR (1<<fe) #define="" (1<<ovr)="" (1<<rxc)="" (1<<txc)="" (1<<udre)="" (1<<upe)="" ***********************************<="" com="0" com;="" data_overrun="" data_register_empty="" if="" int="" it="1" parity_error="" rx_complete="" td="" then="" tx_compete="" usart0="" usart1="" use=""><td></td></fe)></pre>	
/* inform the compiler that an alternate version of the getchar function will be used for USART1 */	
#define _ALTERNATE_GETCHAR_	
/* now define the new getchar function */ char getchar(void) { /* write your code here */ char status,data; switch(COM) {	
<pre>case 0: while (1) { while (((status=UCSR0A) & RX_COMPLETE)==0); data=UDR0; if ((status & (FRAMING_ERROR PARITY_ERROR DATA_OVERRUN))==0) return data; };</pre>	

```
case 1:
      while (1)
          {
         while (((status=UCSR1A) & RX COMPLETE)==0);
         data=UDR1;
         if ((status & (FRAMING_ERROR | PARITY_ERROR | DATA_OVERRUN))==0)
         return data;
         };
  };
}
/* inform the compiler that an alternate version
 of the putchar function will be used */
#define _ALTERNATE_PUTCHAR_
/* now define the new putchar function */
void putchar(char c)
/* write your code here */
  switch(COM)
   ł
    case 0:
      while ((UCSR0A & DATA REGISTER EMPTY)==0);
      UDR0=c;
      break;
    case 1:
      while ((UCSR1A & DATA REGISTER EMPTY)==0);
      UDR1=c:
      while ((UCSR1A & TX_COMPETE)==0);
  };
//**********
                 void main(void)
// Declare your local variables here
// Set up USART1's Baud rate at 9600 bps with a 11.0592 MHz Crystal
UCSR1A=0x00;
                // RX EN, TX EN
               // RX EN, TX EN
UCSR1B=0x18;
               // 8N1
UCSR1C=0x06;
UBRR1H=0x00; // Baud rate high - 9600
UBRR1L=0x47; // Baud rate low
COM = 1;
                      // Use USART1
DDRD.6 = 0;
                                                                                             // Make
PORTD.6 an output
PORTD.6 = 1;
                                                                                             // Enable the
RS485 control line
while (1)
   {
   printf("Hello World\r\n");
   };
}
```