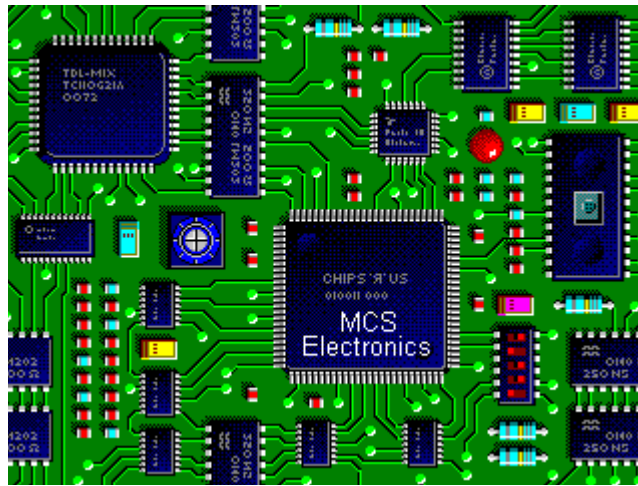




# BASCOM LITE

The Windows 8051 BASIC Compiler



**BASCOM LITE Short guide**  
**© 1995-1999 MCS Electronics**

MCS Electronics may update this documentation without notice.  
Products specification and usage may change accordingly.

MCS Electronics will not be liable for any mis information or errors found in this document.

All software provided with this product package is provided ' AS IS' without any warranty expressed or implied.

MCS Electronics will not be liable for any damages, costs or loss of profits arising from the usage of this product package.

No part of this document may be reproduced or transmitted in any form or by any means, Electronics or mechanical, including photocopying and recording, for any purpose without written permission of MCS Electronics.

Copyright MCS Electronics. All rights reserved.

---

Portions Copyright © 1992-1998 Silicon Studio Ltd.  
<http://www.sistudio.com> a source of PINAPI drivers and programs.

---

## International Distributors

<p><b>Scandinavia</b>                  High Tech Horizon                  Asbogatan 29 C                  S-262 51 Angelholm                  SWEDEN</p> <p>Contact: Christer Johansson                  Phone : +46 431 410088                  Fax : +46 431 410088                  Email: info@hth.com                  WWW: <a href="http://www.hth.com">http://www.hth.com</a></p>	<p><b>Hungary</b>                  CODIX Ltd, Hungary.                  H-1013 Budapest, Atilla u. 1-3                  Hungary</p> <p>Contact: Imre Gaspar                  Phone: +361 356 6330                  Fax: +361 356 4376                  Email: info@codix.hu                  WWW: <a href="http://www.codix.hu">http://www.codix.hu</a></p>
<p><b>Swiss &amp; Germany</b>                  Dr.-Ing. Claus Kühnel                  Consulting &amp; Distribution</p> <p>Contact: Claus Kühnel                  Phone: +41 1 785 02 38                  Fax: +41 1 785 02 75                  Email: ckuehnel@access.ch                  WWW: <a href="http://www.access.ch/ckuehnel">http://www.access.ch/ckuehnel</a></p>	<p><b>Italy</b>                  GRIFO.                  Via dell'Artigiano 8/6                  40016 S.Giorgio di Piano BO                  ITALIA</p> <p>Contact: Salvatore Damino                  Phone +39 (51) 892.052                  Fax +39 (51) 893.661                  E-mail tech@grifo.it                  WWW <a href="http://www.grifo.com">http://www.grifo.com</a> (English)                  WWW <a href="http://www.grifo.it">http://www.grifo.it</a> (Italian)</p>
<p><b>Australia &amp; US</b>                  Dontronics                  PO Box 595                  Tullamarine 3043                  Australia</p> <p>Contact: Don McKenzie                  Email: don@dontronics.com                  WWW: <a href="http://www.dontronics.com">http://www.dontronics.com</a></p>	<p><b>Spain and Portugal</b>                  Multidigital, Lda                  P.O. Box 137                  4435 Rio Tinto                  Portugal</p> <p>Contact: Joaquim Boavida                  Email info@multidigital.com                  WWW <a href="http://www.multidigital.com">http://www.multidigital.com</a>                  Tel : +351 - 2 - 610 22 17                  Fax : + 351 - 2 - 486 21 73</p>
<p><b>Slovenia</b>                  AX Elektronika d.o.o.                  p.p. 5127                  1001 Ljubljana                  SLOVENIA</p> <p>Managing director : Jure Mikeln                  Tel: (+386-61)-14-914-00, -14-914-05                  Fax: (+386-61)-485-688                  Email: jure.mikeln@svet-el.si</p>	<p><b>Czech &amp; Slovak</b>                  LAMIA s.r.o.                  Porici 20 a                  678 01 BLANSKO                  Czech republic                  Contact: Josef Pazderka                  Antonin Straka                  Tel: 00420 506 418726                  Fax: 00420 506 53988</p>
<p><b>Spain</b></p>	<p><b>Korea</b></p>

<p>Ibercomp S.A. C/ del Parc, número 8 (Bajos) E07014 Palma de Mallorca España</p> <p>E-mail <a href="mailto:ibercomp@atlas-iap.es">ibercomp@atlas-iap.es</a> WWW <a href="http://www.ibercomp.es">http://www.ibercomp.es</a> Tel : +34 (9) 71 45 66 42 Fax: +34 (9) 71 45 67 58</p>	<p>SAMPLE Electronics Co. 413-72 kuro-2 kuro Seoul Korea</p> <p>Contact: Junghoon Kim E-mail <a href="mailto:info@sample.co.kr">info@sample.co.kr</a> WWW <a href="http://www.sample.co.kr">http://www.sample.co.kr</a> Tel : 82-2-857-4599 Fax: 82-2-857-6976</p>
<p><b>Croatia</b> Adriatrade d.o.o. Jadranska 47 52470 Umag CROATIA</p> <p>Managing director : Josip Stefek Tel: (+385-52)-752-658 Fax: (+385-52)-752-658 Email: <a href="mailto:adriatrade@ad411.tel.hr">adriatrade@ad411.tel.hr</a></p>	<p><b>UK</b> TECHMAIL SOLUTIONS LTD 14 Dunvegan Road Eltham London SE9 1SA</p> <p>Tel. 0171 343 5242 or 0181 488 9689 Fax.0171 821 6744 Email : <a href="mailto:dogan@dircon.co.uk">dogan@dircon.co.uk</a></p>

Thank you for using BASCOM LITE, the Windows BASIC compiler for the AT89C2051 and other 8051 family members.

I want to thank a few people who helped me with parts of BASCOM LT :

Dr.-Ing. Claus Kühnel for his excellent translation of the documentation into the German language. Check his website : **<http://www.access.ch/ckuehnel>**

Antti from Silicon Studio for the use of his cheap BlowIT 89C2051 programmer design and software. You can check his site at **<http://sistudio.com>**

Jakub Jiricek for the use of his SPI-programmer design and software.

Francois du Plessis for the use of his improved SPI-programmer routines.

This guide will help you to install BASCOM LT and explains the environment briefly.

If you have questions, remarks or suggestions please let us know.

You can contact us by sending an Email to **[mcselec@xs4all.nl](mailto:mcselec@xs4all.nl)**

Our website is on <http://www.xs4all.nl/~mcselec>

**For info on updates : please read the readme.txt file !!!**

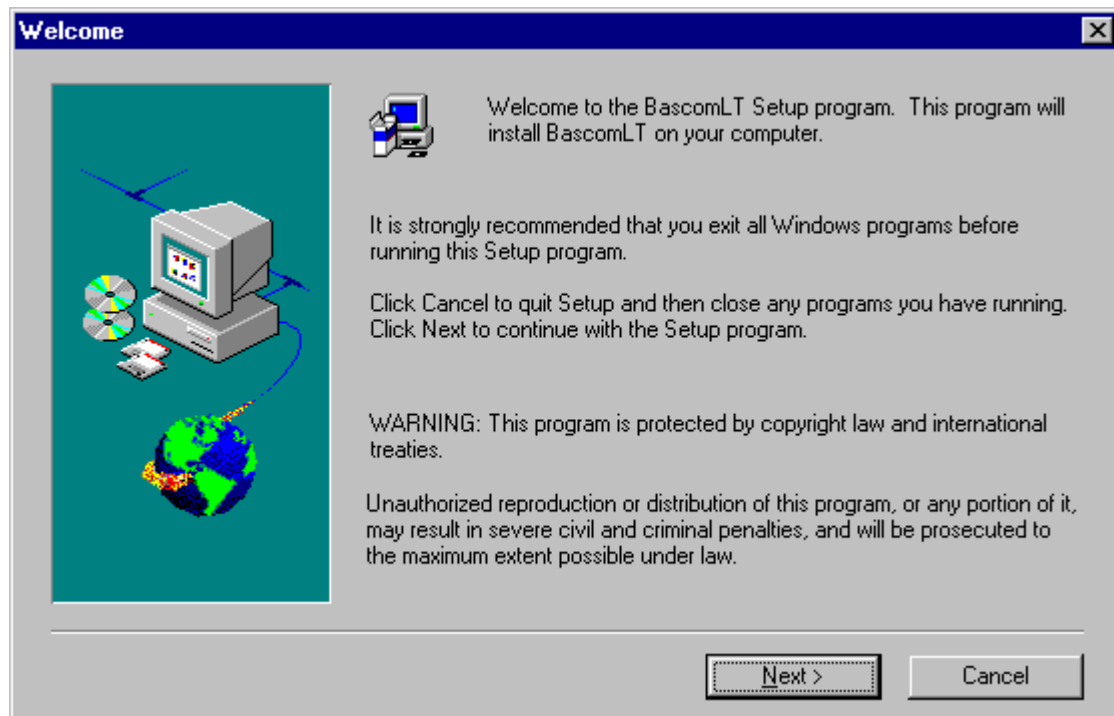
Regards,

Mark Alberts  
MCS Electronics

## Installing BASCOM LT

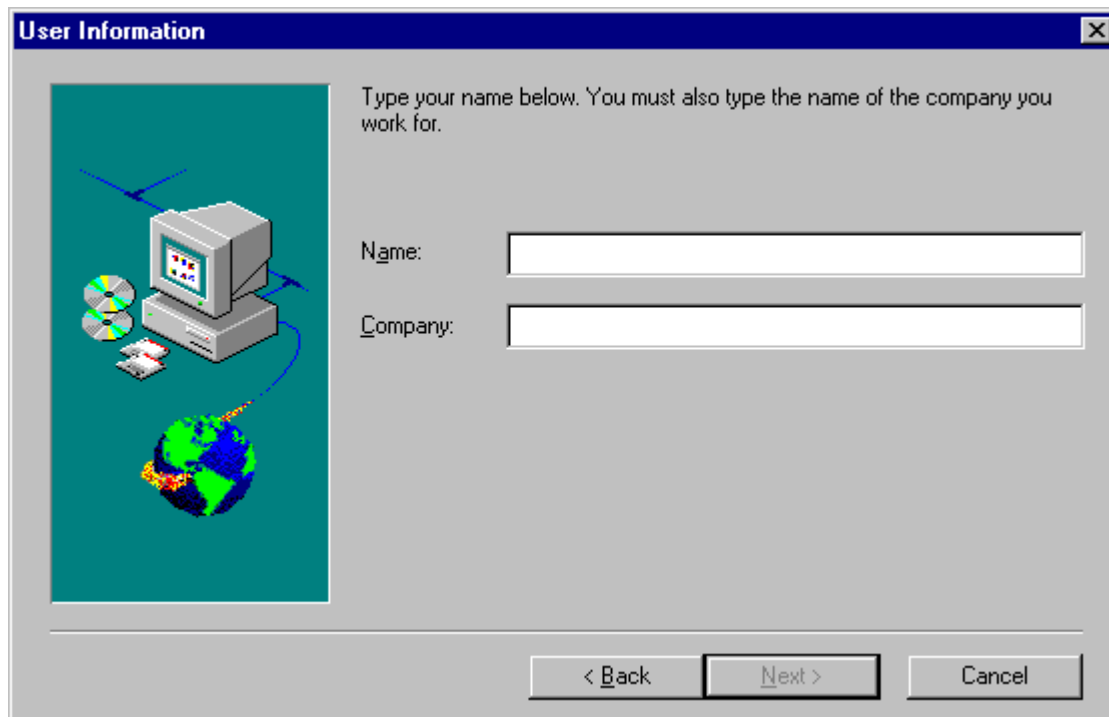
In the program manager from Windows, select **File, Run**, and then type **A:SETUP** .

The following window will appear:



Click on the Next-button to continue installation.

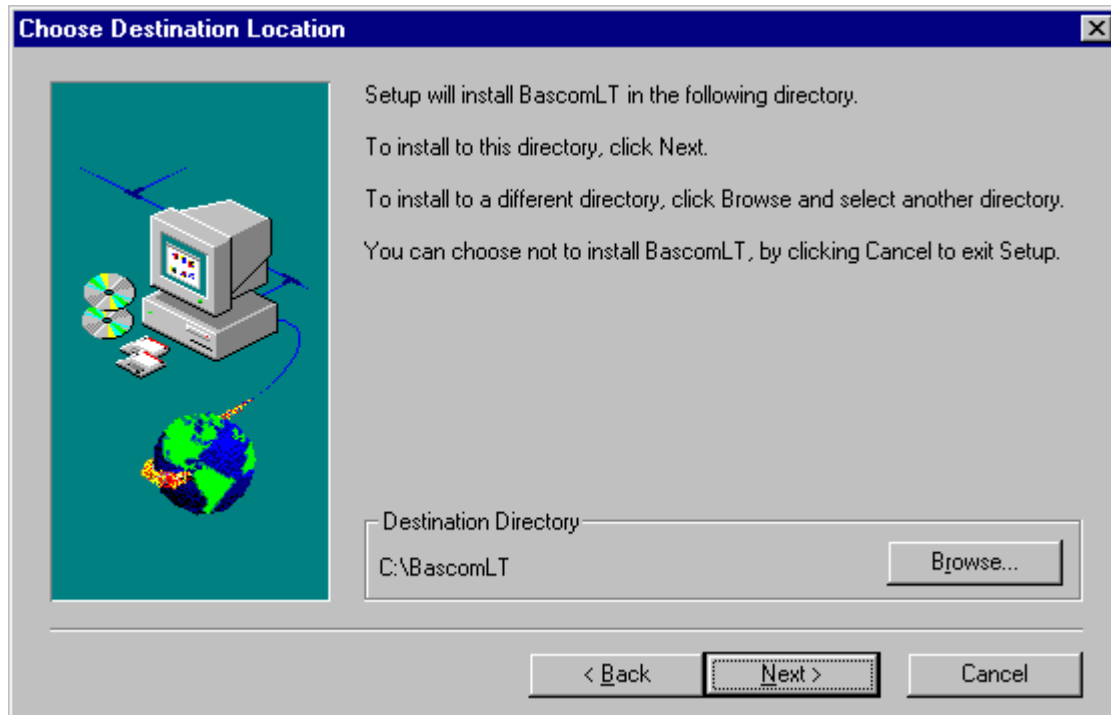
The following window will appear :



The image shows a Windows-style dialog box titled "User Information" with a close button (X) in the top right corner. On the left side, there is a graphic on a teal background depicting a computer monitor, a tower PC, a CD-ROM, and a globe with a blue and green color scheme. To the right of the graphic, the text reads: "Type your name below. You must also type the name of the company you work for." Below this text are two input fields. The first is labeled "Name:" and the second is labeled "Company:". At the bottom of the dialog box, there are three buttons: "< Back", "Next >", and "Cancel".

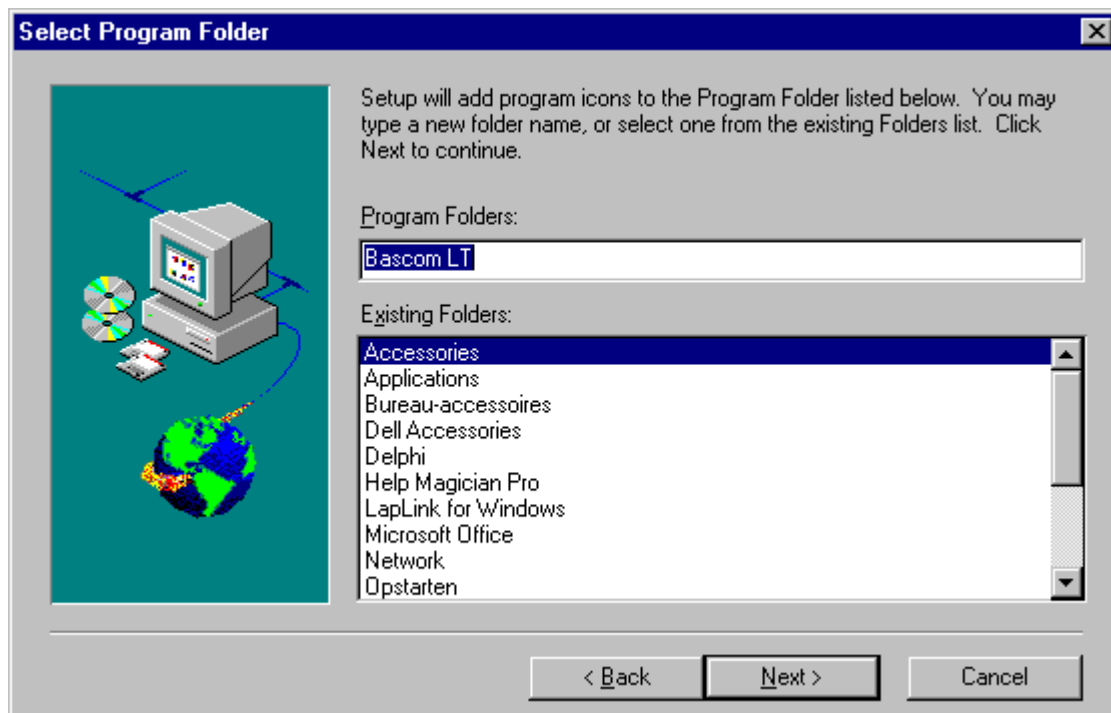
Now fill in your name and company.  
Click the Next-button to continue.

Now you have the change to select the directory in which BASCOM LT will be installed.



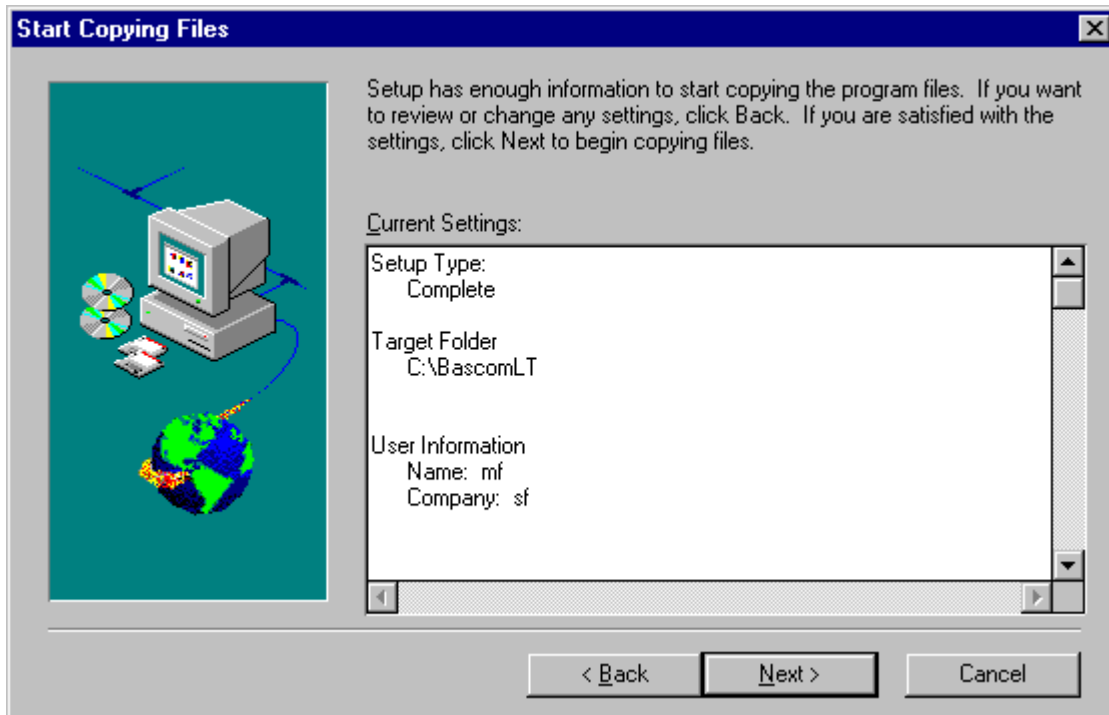
Select the Browse-button to change the directory path.  
By default BASCOM LT will be installed in **C:\BASCOMLT**

After selecting the installation directory click the Next-button.  
Now you will be asked in which program group the BASCOM LT-icon must be placed.  
By default a new program group named BASCOM LT will be made.





After selecting the group click the Next-button to continue. A summary will be showed and you have the change to go back to change your settings. Click the Next-button to install BASCOM LT.



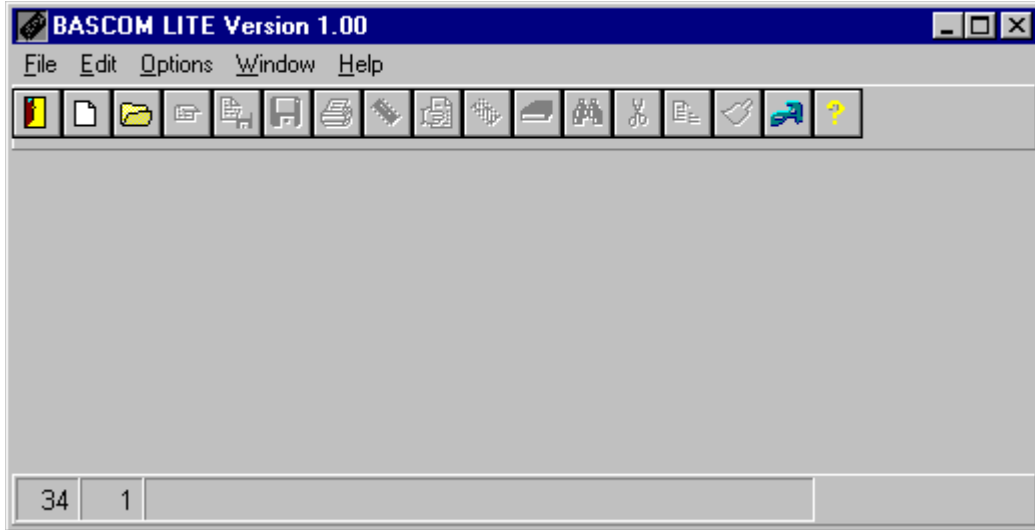
When the installation is completed you must click the Finish-button.

Note that a sub directory named SAMPLES contains all the BASCOM LT sample files.

Depending on the distribution method the CAD files can be found in the CAD subdirectory of can be downloaded from our web site. The file is named **cadfiles.zip**.

## RUNNING BASCOM LT

Double-click the BASCOM LT icon to run BASCOM LT.  
The following window will appear.




The last opened file will be loaded.

The menu options are described on the following pages.

## File New


This option creates a new window in which you write your program. The focus is set to the new window.

File new shortcut : 

## File Open


With this option you can load an existing program from disk. BASCOM LT saves files in standard ASCII format. So if you want to load a file which is made with another editor be sure that it is saved as an ASCII file.

Note that you can specify that BASCOM LT must reformat the file when it opens it with the Options Environment option. This should only be necessary when loading files made with another editor.

File open shortcut : 

## File Close


Closes the current program. When you have made changes to the program or if the file was reformatted you will be asked to save the program first.

File close shortcut : 

## File Save

With this option you can quick save your current program to disk. If the program was created with the File New option you will be asked to name the file first. Use the File Save As option to give the file another name.

Note that the file is saved as an ASCII file.

File save shortcut : 

## File Save As...


With this option you can save your current program to disk.  
You can enter a different name before the program is saved.

Note that the file is saved as an ASCII file.

File save as shortcut : 

## File Print

With this option you can print the current program.  
Note that the current program is the program which has the focus.

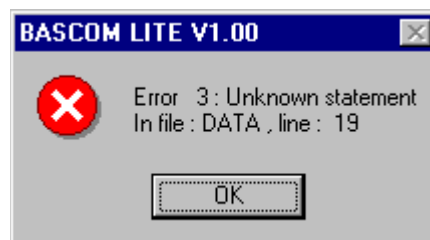
File print shortcut : 

## File Compile

With this option you can compile your current program.  
Your program will be saved automatically before it will be compiled.  
The following files will be created depending on the Option Compiler Settings.

File	
xxx.BIN	Binary file which can be programmed into the uP.
xxx.DBG	Debug file which is needed by the simulator.
xxx.MAP	Map file which is needed by the simulator.
xxx.HEX	Intel hexadecimal file which is needed by some programmers.
xxx.ERR	Error file. (only when errors are found)
xxx.RPT	Report file.

If an error occurs, you will receive an error message. For example :

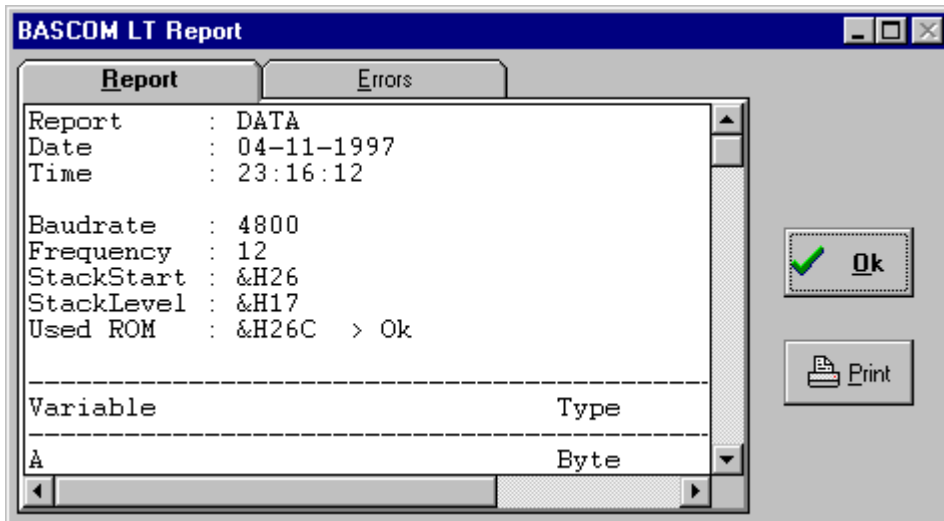


After you have pressed the Ok-button the cursor will be set to the line in which the error occurred.

File compile shortcut : 

## File Show result

Use this option to view the result of the compilation.



See the Options Compiler Output for specifying which files must be created.

The files that can be viewed are report and error.

Click the Print-button to print the selected file.  
Click the Ok-button to return to the editor.

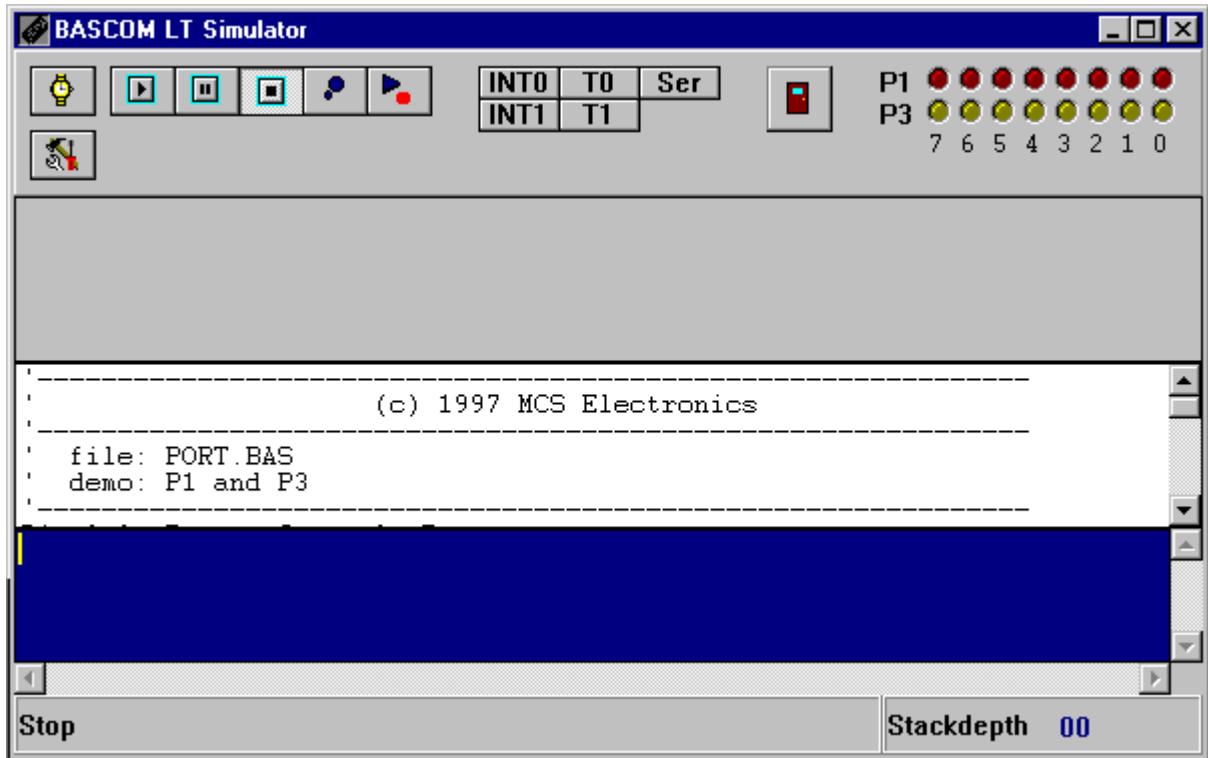
File show result shortcut : 

Information provided in the report:

Info	Description
Date and time	The compilation date and time.
Baudrate and frequency	The baudrate selected for the uP and the used crystal. This info is used for RS232 related statements such as PRINT and INPUT. Note that when you use the \$crystal and \$baud statements the <u>exact</u> baudrate is shown.
Stack start	The starting location of the stack. The space below the stack is used for variables. The stack grows when calls are made by the machine language routines.
Stack level	The number of bytes used from the stack at the deepest level. This value is a rough calculation and the exact amount can be found by running the simulator.
Used ROM	Displays the length of the binary file. If it exceeds 2048 bytes an error will be reported although if you use a processor which can address more memory you won't have to pay attention to this.

## File Simulate

With this option you can simulate your program.  
The simulator window looks like this:



Click the Run-button to start simulation.

The output (if any) is displayed in the lower section.

If input is required then you must set the focus to the lower section before entering the input. This only applies to PRINT and INPUT related statements.

Click the Pause-button to halt execution. Click the run-button to continue.

Click the Stop-button to stop execution.

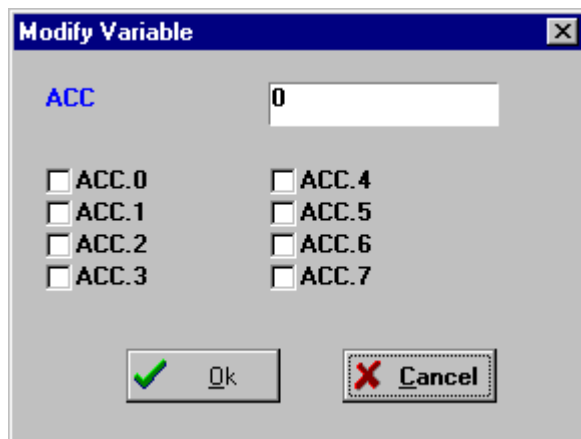
Click the Step-button to step through the program.

Select a line in the source window and click the run to-button to run the program till the selected line. The simulator will pause at that point.

Click the Watch-button to add/remove variables to watch.

With the Environment options you can select to show the internal register too.

To alter the value of a variable double-click on the variable in the variable list. This will bring up the following window :



Note that for bit addressable variables only the value 0 or 1 can be selected. For byte variables the individual bits can be set. Integer and Word variables can be entered as a number only.


Enter the new value and click OK to confirm. The variable will now be adjusted.

The stack depth is the deepest level the stack has reached during execution. If it exceeds 7F (hex) the program will not run correctly in the chip. (chips with 256 bytes of internal RAM don't have this problem)

The ATC89C2051 has 128 bytes of internal RAM space. This space is used by variables you declare and by the stack. You will be warned if you run out of stack space. The solution is to use less variables or use variables of an other type that consumes less memory.

The INT0, INT1, T0, T1 and SET buttons can be clicked to generate an interrupt. Because this is a software emulator no hardware interrupts can be generated. You have to do this yourself by clicking these buttons.



Press the hardware simulation button  to enable the hardware simulation. You need additional hardware to use this feature. An application note can be downloaded that describes the hardware that is needed. The hardware simulator will simulate port 1 and 3. This way you can test your program in circuit without programming the device. At the moment only the status reading and setting of the ports is supported. This means that interrupts are not yet supported in hardware simulation.

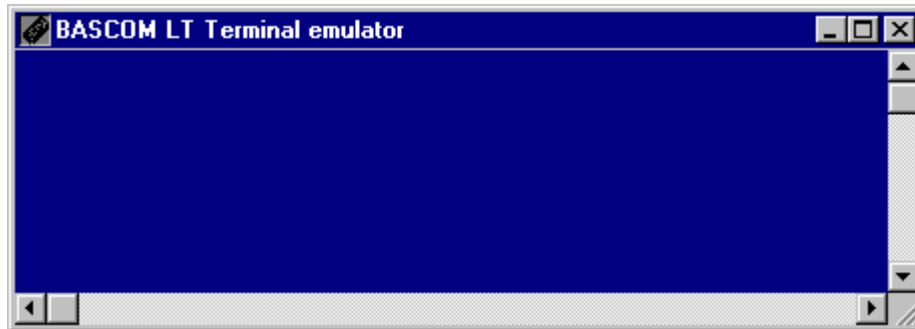
At the upper right position of the screen, LED's are provided to display the status of the ports.

To end simulation click the Exit-button.

File Simulate shortcut : 

## File Terminal Emulator

With this option you can communicate via the RS-232 interface to the microcomputer. The following window will appear :



Information you type and information that the computer board sends, are displayed in the same window.

Note that you must use the same baud rate at both sides. If you compiled your program with the Compiler Settings at 4800 baud, you must set the Communication Settings also to 4800 baud. The setting for the baud rate is also reported in the report file.

### **File Send**

Uploads the current program in HEX format. This option is meant for loading the program into a monitor program.

### **File Escape**

Aborts the upload to the monitor program.

### **File Exit**

Closes terminal emulator.

## File End

With this option you can leave BASCOM LT.

If you have made changes to your program, you can save them upon leaving BASCOM LT.

File exit shortcut : 




## Edit Cut

With this option you can cut selected text into the clipboard.

Edit cut shortcut : 


## Edit Copy

With this option you can copy selected text into the clipboard.

Edit copy shortcut : 

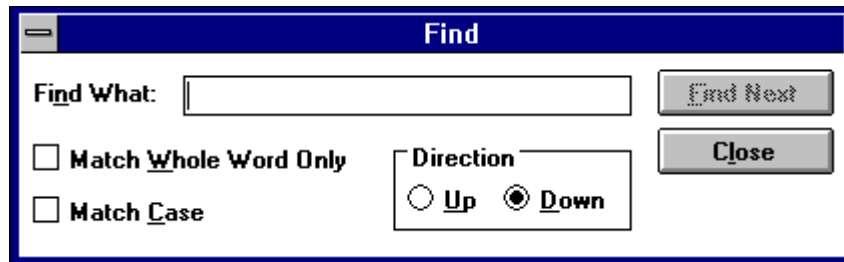
## Edit Paste

With this option you can paste text from the clipboard into the current cursor position.

Edit paste shortcut : 

## Edit Search

With this option you can search for text in your program.  
The standard windows find dialog will appear:



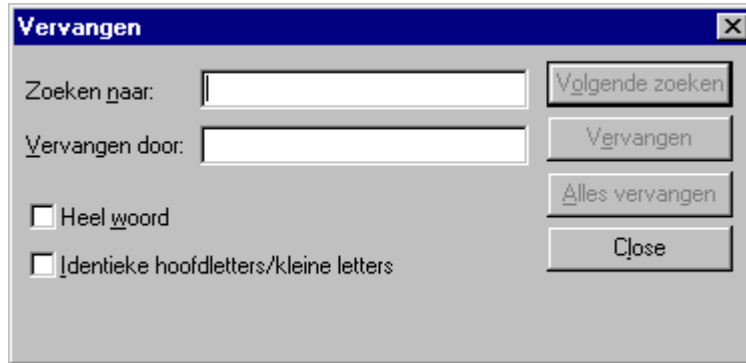
If the text is not found, the following message will appear :



Edit search shortcut : 

## Edit Replace

With this option you can replace text in your program.  
The standard windows replace dialog will appear:



## Edit Find next

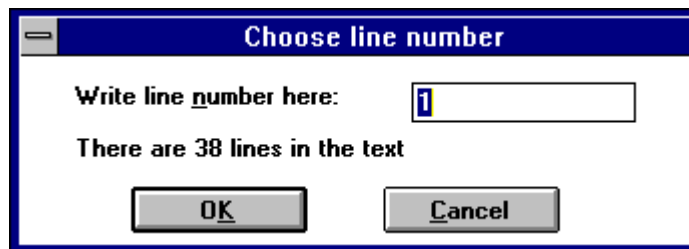
With this option you can search for the next occurrence.  
If no search text was entered previously then the find dialog will be displayed.

Edit find next shortcut : F3

## Edit Goto line

With this option you can goto a line immediately.

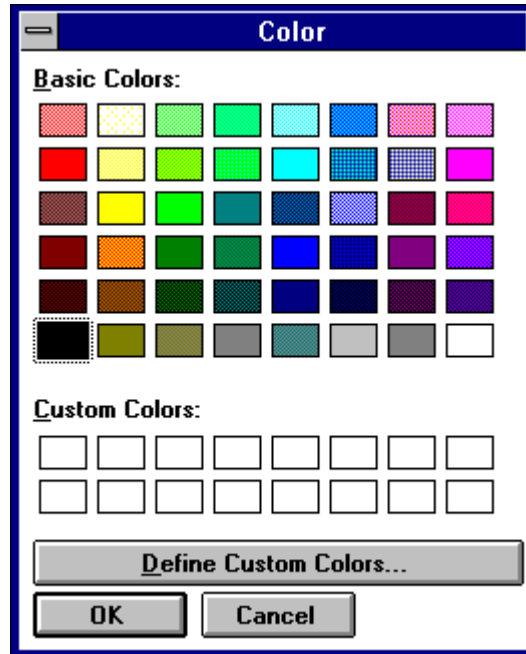
Edit goto line shortcut : CTRL+G



## Edit Background color

With this option you can change the background color of the editor.

The following standard color dialog will appear :

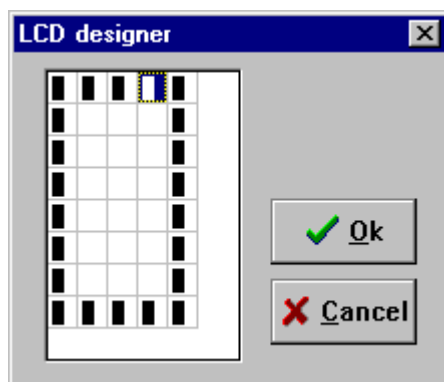


Select a color and Click OK to use that color.

## Edit LCD designer

With this option you can design special characters for LCD-displays.

The following window will appear:



The LCD-matrix has 7x5 points. The bottom row is reserved for the cursor but can be used. You can select a point by double-clicking the left

mousebutton. If a cell was selected it will be deselected. When you are finished you can press the Ok-button : a statement will be inserted in your active program-editor window at the current cursorposition. The statement looks like this :

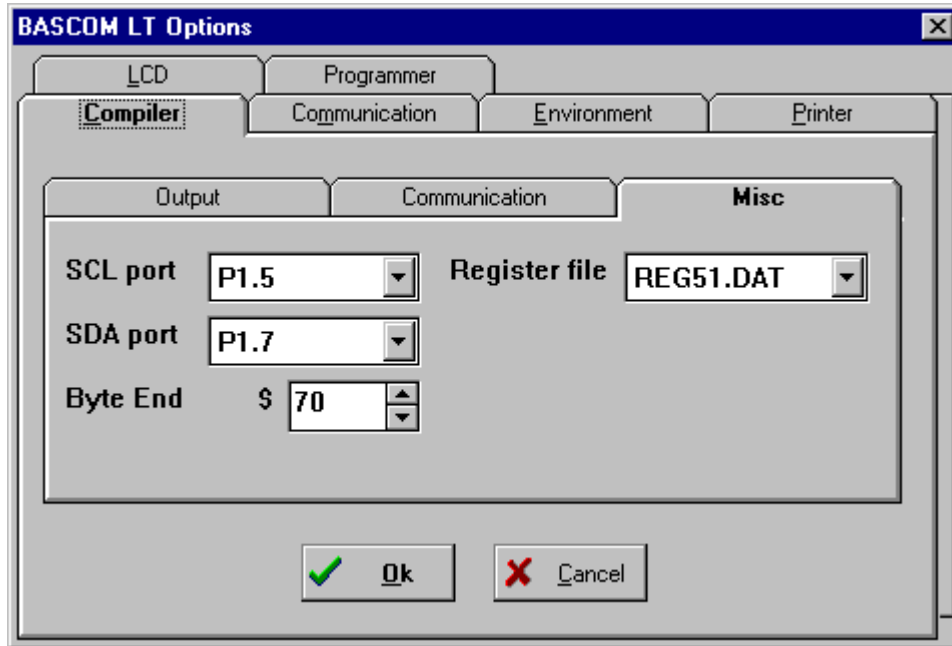
```
Deflcdchar ?,1,2,3,4,5,6,7,8
```

You must replace the ?-sign with a character number ranging from 0-7.

*See Deflcdchar in the helpfile.*

# Options Compiler

With this option you can modify the compiler options.

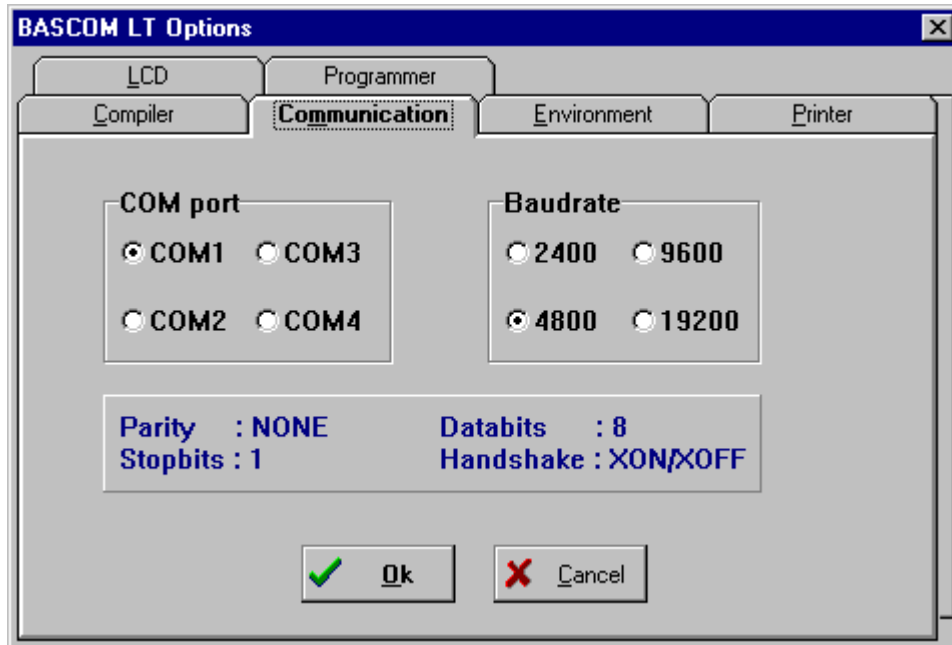


The following options are available:

TAB	OPTION	DESCRIPTION
<b>Output</b>	Binary file	Generates a binary file.
	HEX file	Generates an Intel Hexadecimal file.
	DEBUG file	Generates a debug and map file needed by the simulator.
	Report file	Generates a report file.
	Error file	Generates an error file if an error occurs.
<b>Communication</b>	Baudrate	The baudrate that the RS232 routines like PRINT and INPUT uses to communicate with your computer.
	Frequency	The frequency of the uP crystal on you uP-board.
<b>Misc.</b>	SCL port	The pin that is used for the SCL-line needed by the I2C routines.
	SDA port	The pin that is used for the SDA-line needed by the I2C routines.
	Byte End	The last position in RAM that can be used for internal variables. The stack will start at ByteEnd + 1. When you run out of stack space you must decrease this value, this also means that there is less space for variables.
	Register file	The register file with the SFR addresses that must be used by the compiler. Default is REG51.DAT

## Options Communication

With this option you can modify the communication settings for the terminal emulator. The following window will appear:

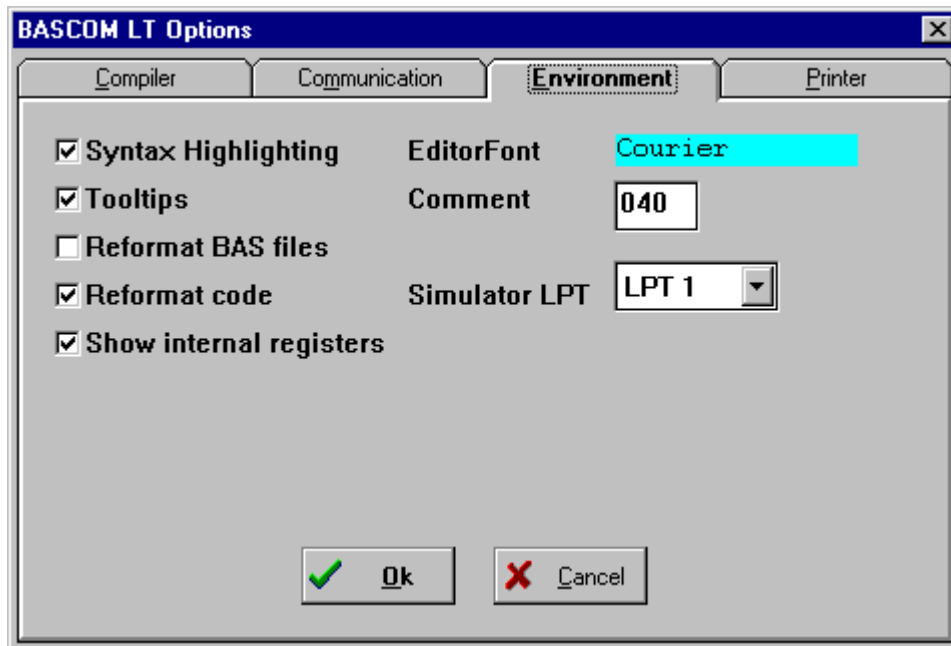


Item	Description
COMPORT	The comport of your PC to use.
BAUDRATE	The baudrate to use.

Note that the baudrate of the terminal emulator and the baudrate setting of the compiler options, must be the same in order to work correctly.

# Options Environment

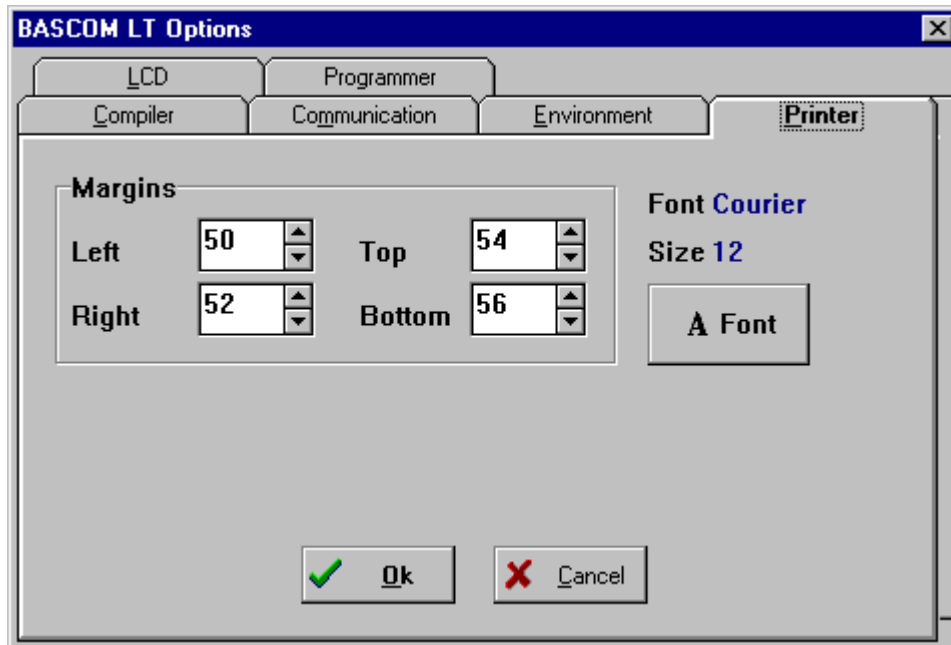
With this option you can modify the environment options.



OPTION	DESCRIPTION
Syntax highlighting	This options highlights BASCOM statements in the editor.
Tooltips	Show tooltips.
Reformat BAS files	Reformat files when loading them into the editor. This is only necessary when you are loading files that where created with another editor. Normally you won't need to set this option.
Reformat code	Reformat code when entered in the editor.
Editor font	Click on this label to select another font for the editor window.
Comment	The position of the comment.
Show internal registers	This option shows the internal registers in the Add Watch windows of the simulator.
Simulator LPT	With this option you can select the printer port that is attached to the hardware simulator.

## Options Printer

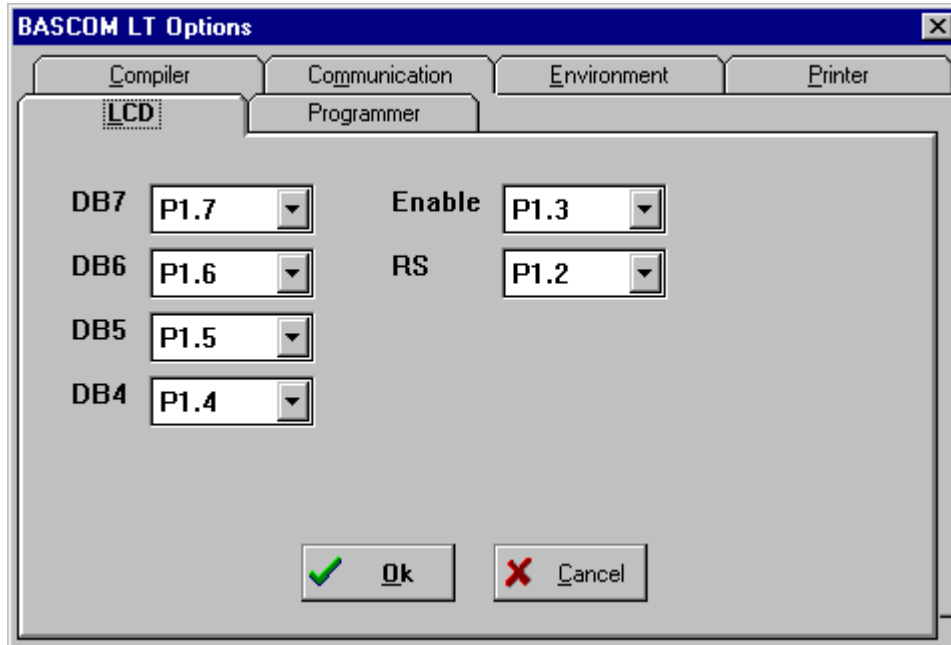
With this option you can modify the printer settings.



OPTION	DESCRIPTION
Left	This left margin.
Right	This right margin.
Top	This top margin.
Bottom	The bottom margin.
Font	Click this button to select the printerfont.

## Options LCD

With this option you can modify the LCD-display settings.



Select the proper port pins for connection to the LCD-display.

In BASCOM LT the 4-bit mode is used to save port pins.

When you want to use an LCD in 8-bit mode you must connect DB0-DB7 of the LCD-display to the data lines D0-D7 of the uP.

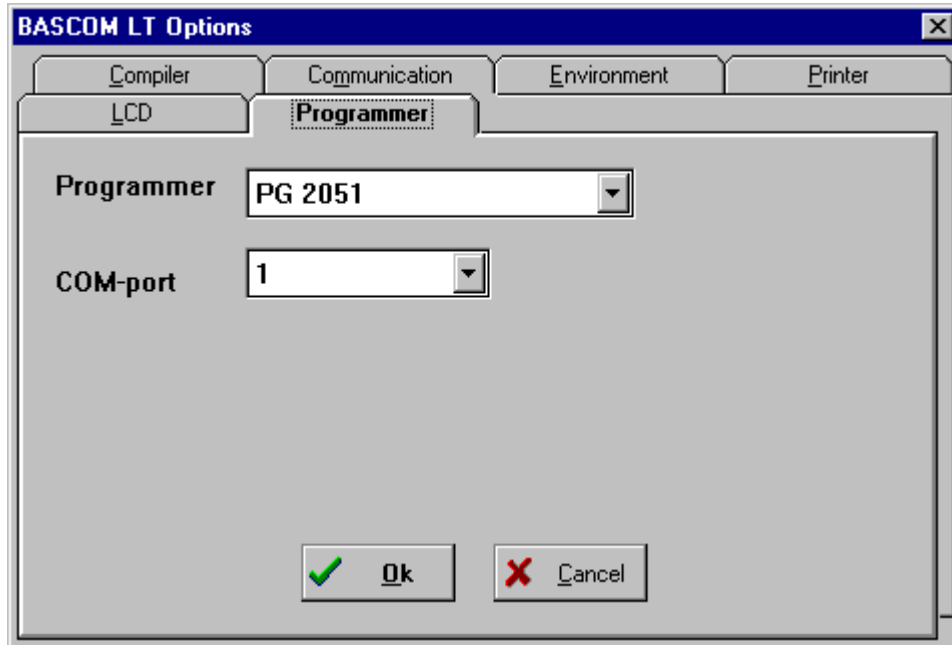
The Enable of the LCD must be connected to an address decoder and the RS line must be connected to the address line A0 of the uP.

In the program you must specify \$LCD = address to use this feature, where address is the address of the address decoder.



## Options Programmer

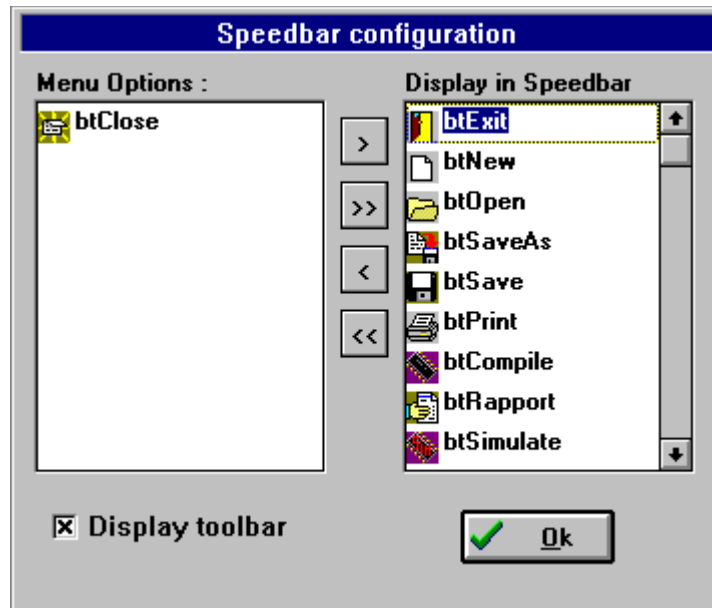
With this option you can modify the programmer settings.



OPTION	DESCRIPTION
Programmer	Choose which programmer you have attached. MCS Flashprogrammer, BlowIT or PG2051.
COM-port	Select the COM-port when you have selected the PG2051 programmer or an other serial programmer. (not available for parallel port programmers)

## Options Toolbar

With this option you can modify the toolbar.



Select the buttons that you want to be displayed in the speedbar with the > and < buttons.

## **Window Cascade**

Cascade all open editor windows.

## **Window Tile**

Tile all open editor windows.

## **Window Arrange icons**

Arrange the icons of the minimized editor windows.

## **Window Minimize all**

Minimize all open editor windows.

## Help About

This option shows an about box as showed below.



Your serialnumber is shown in the about box.

You will need this when you have questions about the product.

The libraryversion is also shown. In this case it is **1.06**.

You can compare it with the one on our website in case you need an update.

Click on **Ok** to return to the editor.

## Help Index

Brings up the BASCOM LT help file.

When you are editing, the current word will be used as a keyword.

## Help on help

Shows help on how to use the Windows help system.

## Editor Keys

Key	Action
<b>LEFT ARROW</b>	One character to the left
<b>RIGHT ARROW</b>	One character to the right
<b>UP ARROW</b>	One line up
<b>DOWN ARROW</b>	One line down
<b>HOME</b>	To the beginning of a line
<b>END</b>	To the end of a line
<b>PAGE UP</b>	Up one window
<b>PAGE DOWN</b>	Down one window
<b>CTRL+LEFT</b>	One word to the left
<b>CTRL+RIGHT</b>	One word to the right
<b>CTRL+HOME</b>	To the start of the text
<b>CTRL+END</b>	To the end of the text
<b>CTRL+ Y</b>	Delete current line
<b>INS</b>	Toggles insert/overstrike mode
<b>F1</b>	Help (context sensitive)
<b>F3</b>	Find next text
<b>CTRL+O</b>	Load File
<b>CTRL+S</b>	Save File
<b>F7</b>	Compile File
<b>CTRL+P</b>	Print File
<b>CTRL+T</b>	Terminal emulator
<b>CTRL+M</b>	File Simulation
<b>CTRL+A</b>	Flash programmer
<b>CTRL+F</b>	Find text
<b>CTRL+P</b>	Compiler Options
<b>CTRL+W</b>	Show result of compilation
<b>CTRL+X</b>	Cut selected text into clipboard
<b>CTRL+INS</b>	Copy selected text into clipboard
<b>SHIFT+INS</b>	Copy text from clipboard into editor
<b>Select text</b>	Hold the SHIFT key down and use the cursor keys to select text. or keep the left mouse key pressed and tag the cursor over the text to select.

Important note to codepage 852 users (Hungary)

When the >-key does not work you can press the **right ALT + :** keys.

## Developing order

- start BASCOM LT;
- open a file or create a new one;
- check the baudrate and frequency settings for the target system;
- compile the file;
- if an error occur fix it and recompile (F7);
- run the simulator and check if you don't run out of stack space;
- check if the file size doesn't exceeds 2048 bytes (AT89C2051);
- program the chip;

### Memory

Every variable uses memory.  
The available memory is 128 bytes.  
From these 128 bytes a maximum of 32 are used for internal registers.  
Some bytes are used by the stack. This depends on the statements you have used.  
Each 8 used bits occupy 1 byte.  
Each byte occupy 1 byte.  
Each integer/word occupies 2 bytes.  
Each Long occupies 4 bytes.

Use bytes when you can. (not allowed for negative values)

If your program use less than 64 bytes you can use an 89C1051 as well. This chip only doesn't have an UART for PRINT and INPUT statements.

You can also use the generated code for other chips as well.  
Other chips have the benefit that much more code can be burned into EPROM.  
The generated code can be larger than 2048 bytes.

## Error codes

The following table list all errors that can occur.

Nr	Error message
1	BASIC sourcefile not found
2	Code does not fit into FLASHROM
3	Unknown statement
4	Extension expected
5	Wrong variable or variable not dimensioned
6	Two parameters expected
7	No more space for BIT
8	No more space for BYTE
9	No more space for INTEGER
10	Wrong type (BIT,BYTE or INTEGER) expected
11	AS expected by DIM
12	, expected
13	Unknown interrupt
14	IF THEN expected
15	FOR, DO or WHILE expected
16	Wrong number of parameters
17	Illegal compare (=,>,<,<>,<=,>=) expected
18	THEN expected
19	TIMER0 or TIMER1 expected
20	DO expected
21	UNTIL expected
22	Illegal mathematical operation
23	FOR expected
24	WHILE expected
25	Variable not dimensioned
26	Source file not found
27	Label not found
100-134	These are internal assembler warnings. Contact MCS Electronics .
135	Too many RAM used
136	Variable already dimensioned
137	Constant must be in range of 1-8
138	Baudrate not supported with selected frequency
139	9 parameters expected
140	COUNTER0 or COUNTER1 expected.
141	= expected.
142	Maximum of 128 ALIAS statement allowed.
143	Duplicate label
144	Value does not fit into byte(byte can store 0-255)

## Hardware

You can attach additional hardware to the ports of the microprocessor.  
The following statements will become available :

**I2CSEND** and **I2CRECEIVE** and other **I2C** related statements.  
**LCD, LCDHEX, DISPLAY** and other related **LCD**-statements.

See the compiler settings for selecting the SDA (data) and SCL(clock) pins.

The LCD-display must be connected as following:

LCD-DISPLAY	PORT	PIN
DB7	P1.7	14
DB6	P1.6	13
DB5	P1.5	12
DB4	P1.4	11
E	P1.3	6

RS	P1.2	4
RW	Ground	5
Vss	Ground	1
Vdd	+5 Volt	2
Vo	0-5 Volt	3

This leaves P1.1 and P1.0 and P3 for other purposes.

You can change these settings from the Options LCD menu.

The LCD-display operates in 4-bit mode.

BASCOM LT supports a lot of statements to control the LCD-display.

For those who want to have more control the example below shows how to.

```
Acc = 5           'load register A with value
Call Lcd_control 'it is a control value to control the display
Acc = 65         'load with new value (letter A)
Call Write_lcd   'write it to the LCD-display
```

Note that `lcd_control` and `write_lcd` are assembler subroutines which can be called from BASCOM LT.

See manufacture details from your LCD display for the correct assignment.



## I2C

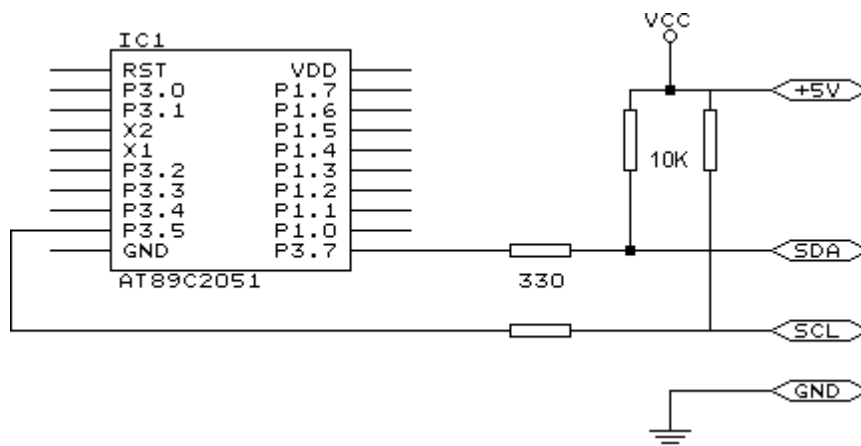
The design below shows how to implement an I2C-bus.

R1 and R2 are 330 ohm resistors.

R3 and R4 are 10 kilo-ohm resistors.

Note that you can select which port pins you want to use for the I2C interface with the compiler settings.

A PCB board with RS-232 interface, I2C interface, EEPROM 2404 and I/O expander PCF8574 is also available. Please inform for prices.



## Hardware related commands

The uP must be connected to a crystal. The frequency of the crystal can range from 0 to 24 Mhz (for AT89Cx051).

The frequency is divided by 12 internally. So with a 12 Mhz crystal the processor is clocked with 1 Mhz.

Because almost each instruction takes 1 clockcycle to execute the processor can handle 1 MIPS.

When RS-232 statements such as INKEY, PRINT and INPUT are used, TIMER1 is connected to the systemclock. So TIMER1 can't be used for other purposes such as ON TIMER1 anymore. When no RS-232 related statements are used you can use TIMER1.

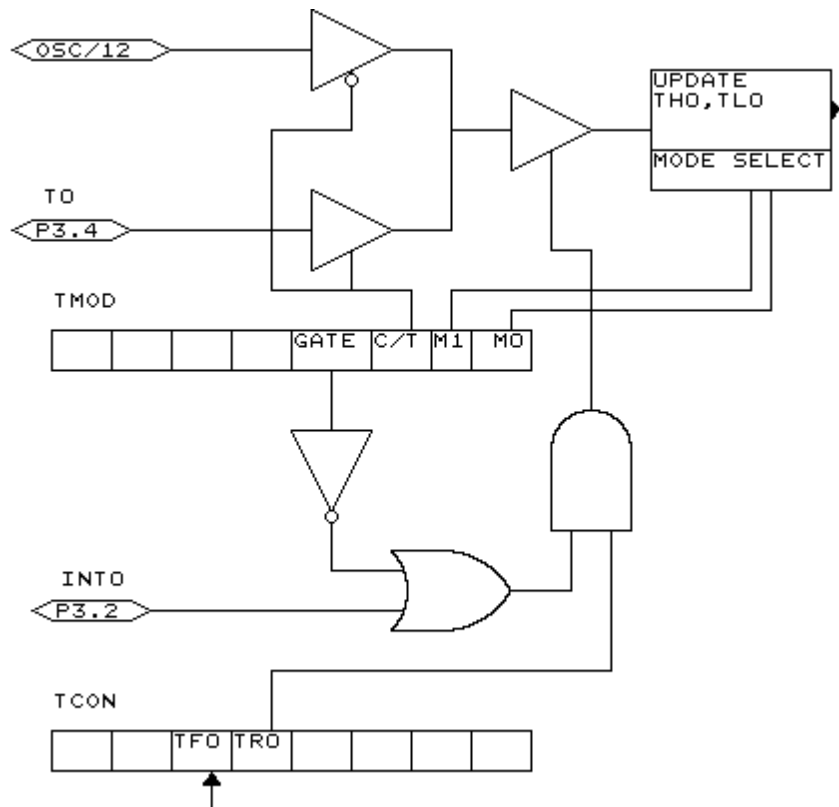
The Baudrate is generated by dividing the systemclock. When a crystal of 11.0592 Mhz is used, the Baudrate can be generated very accurately. Other crystals can be used also but the generated baudrate will never be exactly 2400 or 4800 baud and higher baudrates are almost impossible. Use the \$BAUD = xx and \$CRYSTAL = xx statements to specify your own settings. The exact baudrate will be shown in the report.

## Clock

The clock frequency is the system frequency divided by 12. With a 12 Mhz crystal this means that every microsecond the register is incremented.

## Timers and Counters

The AT89C2051 has two 16-bit timers. Named TIMER0 and TIMER1. Below the internal representation of timer0 is shown. TIMER0 and TIMER1 are almost identical so you can read TIMER1 for TIMER0.



Each counter register has two SFR's associated with it. For TIMER0 the SFR's are TL0 and TH0. TL0 is the lowest byte of TIMER0 and TH0 is the highest byte of TIMER0. These two registers make the timers 16-bit wide.

The timer can operate as a timer or as a counter.

A timer uses the system clock divided by 12 as the source of its input pulses. So it increments periodical.

A counter uses external pulses to increment its count.

The external pulses are received at alternative pin P3.4 for TIMER0 and P3.5 for TIMER1.

The timer/counter can be controlled by the run-bit TR0.

You can stop a timer/counter with the statement STOP TIMER0/COUNTER0.

You can start a timer/counter with the statement START TIMER0/TIMER1.

The timer/counter can also be controlled with the alternative pin P3.2.

This pin is labeled for its alternative INT0-input but it can be used to control the timer. When GATE is reset the timer/counter is enabled. When GATE is set the timer/counter is enabled if INT0 is active(low). (provided that the timer is started)

The timer/counter can operate in four modes:

mode 0 : 13-bit counter.

An interrupt is generated when the counter overflows. So it takes 8192 pulses to generate the next interrupt.

mode 1 : 16-bit counter.

Mode 1 is similar to mode 0. It implements a 16-bit counter. It takes 65536 input pulses to generate the next interrupt.

mode 2 : 8-bit auto reload.

TL0 serves as an 8-bit timer/counter. When the timer/counter overflows the number stored in TH0 is copied into TL0 and the count continues.

An interrupt is generated each time the counter overflows and a reload is performed.

mode 3 : TIMER1 is inactive and holds its count. (TIMER1). For TIMER0 in timer mode two 8-bit timers are available and in counter mode one 8-bit timer is available. See a datasheet for more details.

The timer/counter can be configured with the CONFIG statement.

**CONFIG** TIMER0= COUNTER/TIMER, GATE=INTERNAL/EXTERNAL, MODE=0/3

The first argument is the timer/counter you want to configure.

GATE specifies if external timer control with the INT0 pin is enabled.

MODE specifies the timer/counter mode (0-3).

So CONFIG TIMER0 = COUNTER, GATE = INTERNAL, MODE=2 will configure TIMER0 as a COUNTER with not external gatecontrol , in mode 2 (auto reload)

When the timer/counter is configured the timer/counter is stopped so you must start it afterwards with the START TIMER0 statement.

The ON TIMERx statement can be used to respond to a timer/counter interrupt when the timer overflows.

When the timer/counter is used in mode 2 (auto reload) the reload value can be specified with the **LOAD TIMERx, value** statement.

Because it is an 8-bit register a maximum time of 255 uS can be achieved.

So for a period of 10 uS you must supply a value of (256-10) is 246. To make things easier you can assign the value directly : LOAD TIMERx , 250 will internally be transformed into 256-250=6. This saves you the trouble of calculating the correct value.

The COUNTER0 and COUNTER1 variables hold the values of timer/counter 0 and 1. You can also set the timer/counter contents with the COUNTER0 = value statement.

Port 3 is a unique port because it has alternative functions.

That is you can use it as a port like P3.1 = 1 or SET P3.1 or you can make use of the double function of this port.

<b>Port</b>	<b>Alternative function</b>
P3.0	RxD receive data for RS-232
P3.1	TxD transmit data for RS-232
P3.2	INT0 interrupt 0 input/timer 0 gate control
P3.3	INT1 interrupt 1 input/timer 1 gate control
P3.4	T0 timer 0 input or counter input
P3.5	T1 timer 1 input or counter input
P3.5	-
P3.7	-

When you make use of the PRINT, INPUT and other RS-232 related statements P3.0 and P3.1 are used for the RS-232 interface.

When you make use of the INT0/INT1 interrupts you must connect an interrupt source to the corresponding pins. A switch for example.  
 The INTx interrupt can occur on the falling edge of a signal or when the signal is low.  
 Use the following statements to specify the trigger:

SET TCON.0	falling edge generates interrupt for INT0.
RESET TCON.0	low signal generates interrupt for INT0.
SET TCON.2	falling edge generates interrupt for INT1.
RESET TCON.2	low signal generates interrupt for INT1.

When TCON.x is RESET the interrupts keep on occurring while the input is low.  
 When TCON.x is SET the interrupt only occurs on the falling edge.

To test if a hardware interrupt is generated you can test the TCON.1 and TCON.3 flags. These flags are set by hardware when an external interrupt edge is detected. They are reset by the RETURN statement of the interrupt service routine or subroutine. TCON.1 must be tested for INT0 and TCON.3 must be tested for INT1.

## Internal Registers

You can manipulate the register values directly from BASIC. They are also reserved words. The internal registers are :

### BIT addressable registers

TCON	Timer/counter control
P1	Port 0 latch
SCON	Serial port control
IE	Interrupt enable
P3	Port 3 latch
IP	Interrupt priority control
PSW	Program status word
ACC	Accumulator
B	B register

### BYTE addressable register

SP	Stack pointer
DPL	Data pointer low word
DPH	Data pointer high word
PCON	Power control
TMOD	Timer/counter mode control
TL0	Timer/counter 0 low byte
TL1	Timer/counter 1 low byte
TH0	Timer/counter 0 high byte
TH1	Timer/counter 1 high byte
SBUF	Serial data port
P1	Port 1 latch
P3	Port 3 latch

The registers and their addresses are defined in the REG51.DAT file which is placed in the BASCOMLT application directory.

You can add your own files for uP's that have more registers. These files must have the .DAT extension.

The REG51.DAT file must be selected when you compile for the AT89X051, 8051 or other 8051 compatible uP.

Take care when you are directly manipulating registers!

The ACC and B registers are frequently used by BASCOM LT.

Also the SP register is better to leave untouched. Altering SP will certainly crash your application!

Bit addressable registers can be used with the SET/RESET statements and as bit-variables.

Byte addressable registers can be used as byte variables.

P1 = 40 will place a value of 40 into port 1.

Please note that internal registers are reserved words. This means that they can't be dimensioned as BASCOM LT variables!

So you can't use the statement **DIM B as Byte** because **B** is an internal register.

You can however manipulate the register with the B = value statement.

## Power Up

At power up both ports are high. When you want to read a value from a port it must be high!

So when you have set a port to zero and you want to use it as an input you must first set it to 255.

Individual bits can also be set to use a port both as input/output.

For example : P1 = **&B00001111** , will set a value of 15 to port 1.

P1.0 to P1.3 can be used as inputs because they are set high.

## In line assembly

You can also use inline assembly language with BASCOM LT.

Assembler comment must be preceded by the ;-sign.

**For example :**

*Dim a as byte*

***mov** {a}, #10 ; variables must be enclosed with { }*

*Print a*

Or you can include an assembler file :

**\$INCLUDE** myasm.asm

Note that the file must have the extension .ASM

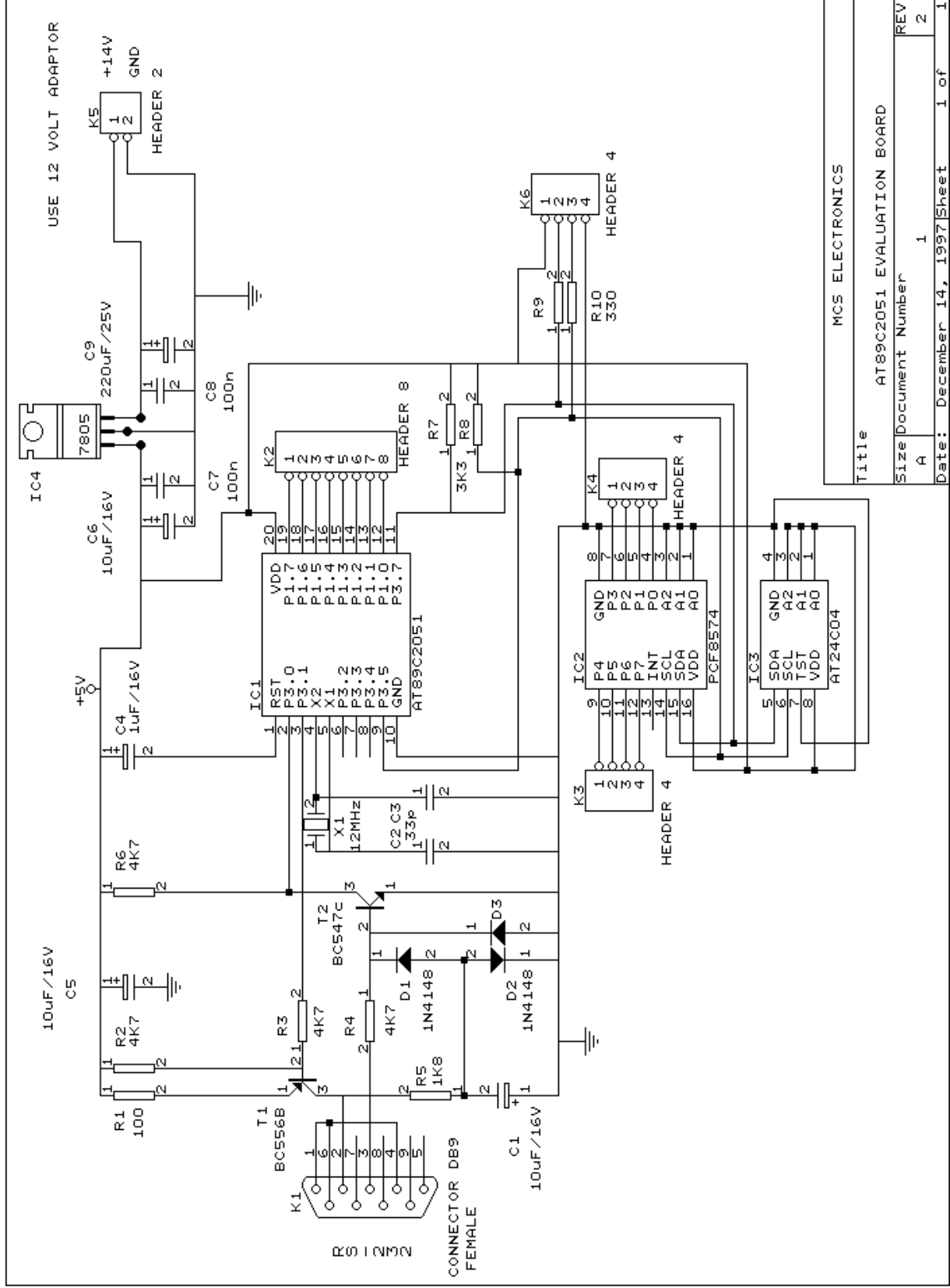
## Initialization

BASCOM LT initializes the processor depending on the used statements.

When you want to handle this by yourself you can specify this by the metastatement

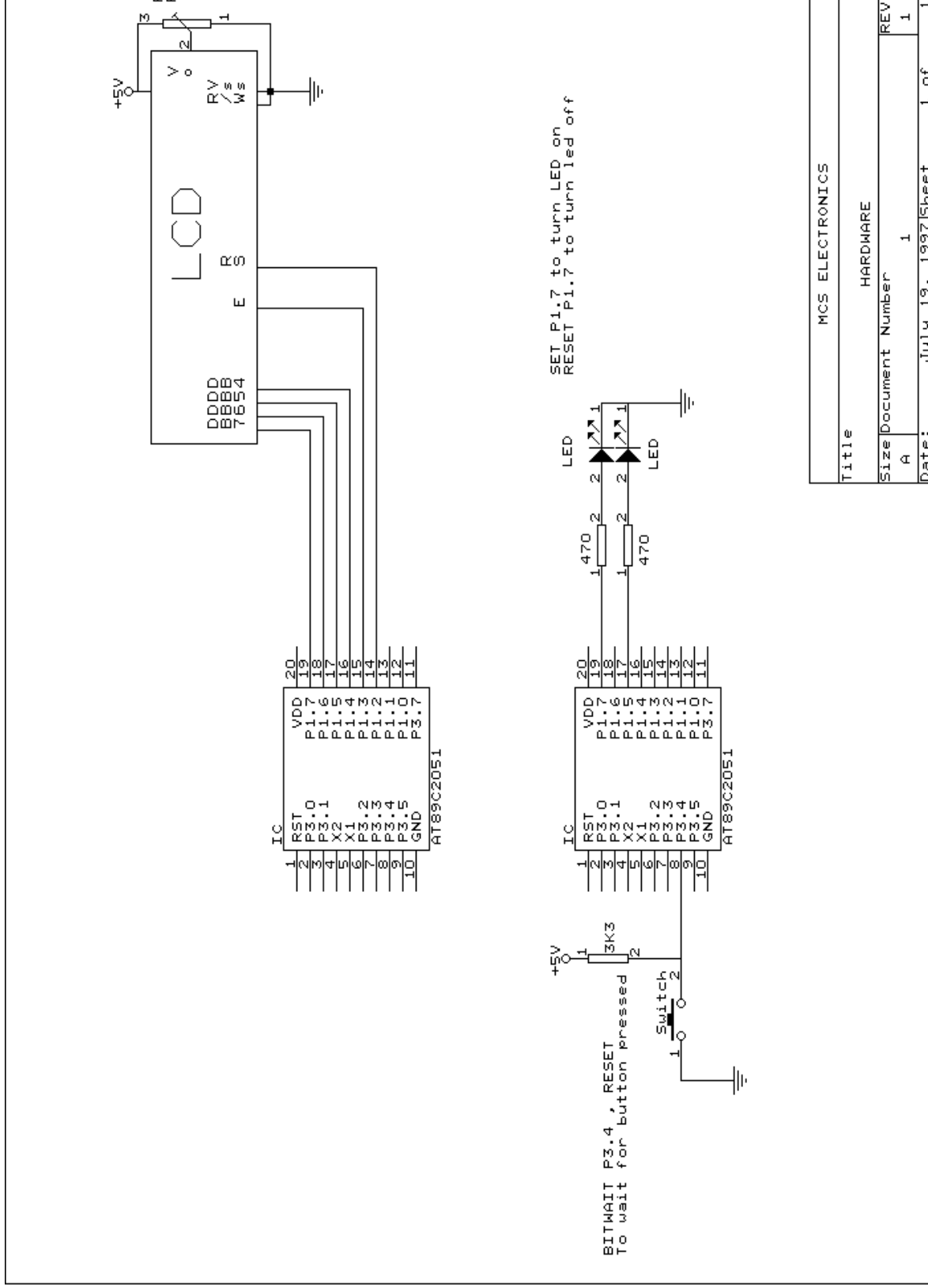
**\$NOINIT.**

The only initialization that is always done is the setting of the stackpointer and the initialization of the LCD-display (if statements are used).



MCS ELECTRONICS	
Title	AT89C2051 EVALUATION BOARD
Size	A
Document Number	1
REV	2
Date:	December 14, 1997
Sheet	1 of 1





**D**

**Developing order** ..... 30

**E**

**Edit Background color** ..... 19  
**Edit Copy** ..... 17  
**Edit Cut** ..... 17  
**Edit Find next** ..... 18  
**Edit Goto line** ..... 18  
**Edit LCD designer** ..... 19  
**Edit Paste** ..... 17  
**Edit Replace** ..... 18  
**Edit Search** ..... 17  
**Editor Keys** ..... 29  
**Error codes** ..... 31

**F**

**File Close** ..... 11  
**File Compile** ..... 12  
**File End** ..... 16  
**File New** ..... 11  
**File Open** ..... 11  
**File Print** ..... 12  
**File Save** ..... 11  
**File Show result** ..... 13  
**File Simulate** ..... 14  
**File Transmit** ..... 16

**H**

**Hardware** ..... 32  
**Hardware related commands** ..... 33  
**Help About** ..... 28  
**Help Index** ..... 28  
**Help on help** ..... 28

**I**

**I2C** ..... 33  
**In line assembly** ..... 39  
**Initialization** ..... 39  
**Installing BASCOM LT** ..... 6  
**Internal Registers** ..... 37

**O**

**Options Communication** ..... 21  
**Options Compiler** ..... 20  
**Options Environment** ..... 22  
**Options Printer** ..... 23; 25  
**Options Programmer** ..... 24  
**Options Toolbar** ..... 26

**P**

**Power Up** ..... 39

S

Save As..... 12

W

Window Arrange icons..... 27  
Window Cascade..... 27  
Window Minimize all..... 27  
Window Tile..... 27