Silicon Storage Technology, Inc.

Boot-Strap Loader Software Example

PC: EasyIAP11F Software
MCU: v1.1F Firmware

User's Guide

S74002-00-001 8/02

TABLE OF CONTENTS

1.0 INTRODUCTION

The EasyIAP Boot-Strap Loader (BSL) software example[1] enables SST customers to download/upload their application software into/from the FlashFlex51 flash memory via In-Application Programming (IAP) while the system is running. The software package is composed of a windows-based software called EasyIAP11F for the PC and BSL v1.1F firmware for the SST FlashFlex51 family of MCUs.

EasyIAP11F software is capable of the following:

- determining the most suitable baud rate over an external crystal frequency range of 1 to 40 MHz
- providing self-detection of the serial connection
- performing upgrades to newer firmware versions
- setting and changing password to protect the privacy of flash contents
- modifying internal flash memory contents at the byte-level

BSL v1.1F firmware is capable of the following:

- performing file downloads and uploads from either internal or external memory
- automatically detecting/establishing the RS232 serial communication baud-rate with the host PC
- storing/verifying user password to prevent unauthorized code/data fetches from the MCU flash memory
- performing IAP operations in support of flash memory reads, writes, verifies and erases
- supporting BSL firmware upgrades

The companion SST89C54/58 or SST89x564RD/SST89x554RC MCU data sheet and the earlier version (v1.1E) of MCU firmware source code should be reviewed in conjunction with this document for a complete understanding of the software application. Data sheets and sample source code for v1.1E can be downloaded from the SST web site (www.SST.com).

There are seven appendices at the end of this User's Guide. Appendix A provides the file name convention for the MCU firmware; Appendix B lists available PC, MCU and external memory files; Appendix C contains a list of switch combinations for the BSL Demo Board v2.0; Appendix D furnishes the schematic for the BSL Demo Board v2.0; Appendix E provides the SoftICE User's Guide for the SST89C5x, SST89x554RC, and SST89x564RD MCUs; Appendix F provides the MCU resources used by BSL v1.1F; and Appendix G shows the entry stage from the user code into BSL routines.

1. The Boot-Strap Loader Software Example is for the user's reference and convenience only. SST does not guarantee the functionality or the usefulness of the example bootstrap loader.

2.0 FIRMWARE UPGRADES

2.1 Automatic Upgrade to BSL v1.1F from BSL v1.1E

During factory test, the SST89C54/58 chips are pre-programmed with BSL v1.1E into block 1 flash memory, and are re-mapped to the 1KB range. Users need to upgrade this earlier version of firmware to current BSL v1.1F by following the instructions of "Detect Chip/RS-232 Configuration" in Internal Memory Mode, covered in Section 4.2.2 of this guide.

2.2 BSL Demo Kit v1.0 - Reprogram SST39SF010A MPF to Work with BSL v1.1E

For users of the SST BSL Demo Kit Version 1.0 (P/N SST89CK77BSL) that have to program the MPF chip in the socket of U3, use the following procedure:

1. Remove the SST39SF010A MPF from the U3 socket.
2. Use a universal programmer to program the file of F51EBLK5.HEX into the 0000h location.

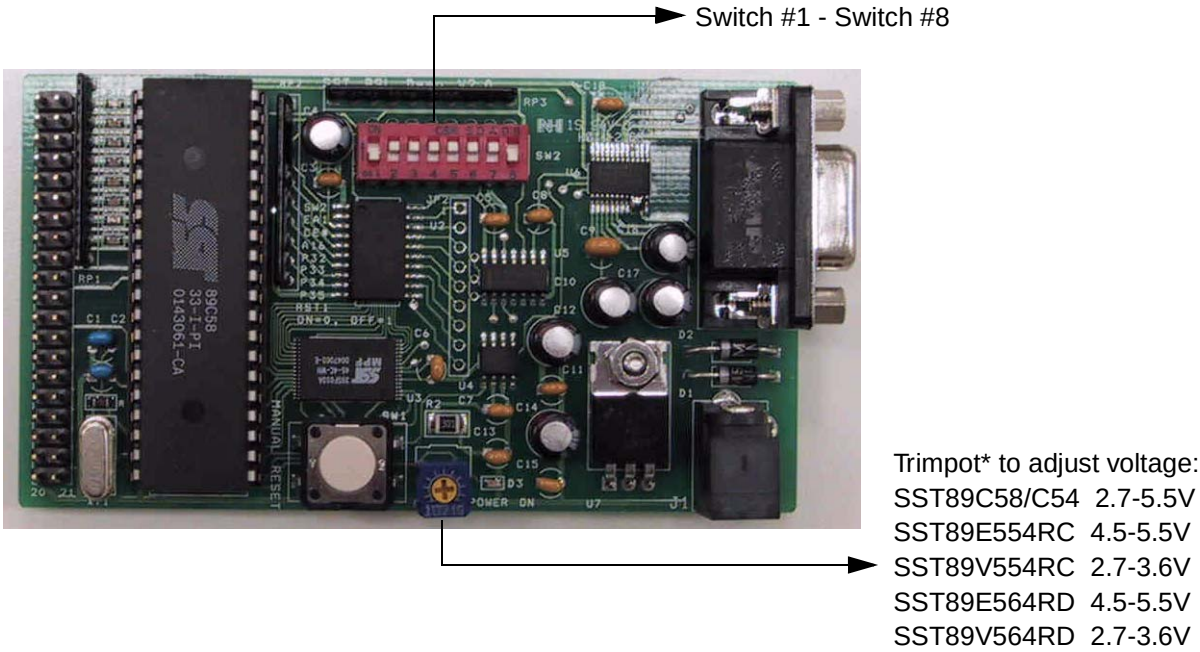
The file F51EBLK5.HEX can be downloaded from the SST website (www.SST.com).

3.0 BSL DEMO BOARD v2.0

BSL Demo Board v2.0 is a newly designed board, which has many new features compared to the old board (v1.0), including:

- 2.7V to 5.5V operation
- On-board and off-board Reset and EA pin control
- 128 KByte of external program memory
- External flash memory upgrade capability
- MCU signal access
- User-defined 4-input switches

With Switch#1 and #8 turned off, PC software can control EA and RST signals, and the reset button on the demo board will not work. Sw#1 controls EA1, Sw#8 controls RST1.



Customer can Read, Write, Sector-Erase, or Chip-Erase external flash SST39SF010A with Sw#2 turned on. Details are given in Section 4.2.5.

*Factory set to 3.3-3.4V. To increase the voltage, turn trimpot clockwise. Output voltage range (via trimpot setting) is between 1.2-5.5V.

3.1 Serial Cable

A standard RS-232 DTE-DTC cable is required to connect the Host-PC to the BSL Demo Board (or other development platform). The female side of the cable connects to the PC (Data Terminal Equipment, DTE) and the male side connects to the BSL Demo Board or other development platform (Data Communication Equipment, DCE). The serial cable connections are:

PC DB-9 plug (any COM port)			Development Platform	
RxD	Pin 2	to	TxD	Pin 2
TxD	Pin 3	to	RxD	Pin 3
DTR	Pin 4	to	DTR	Pin 4
GND	Pin 5	to	GND	Pin 5
RTS	Pin 7	to	RTS	Pin 7

No hardware handshake line is required to invoke communication between the Host-PC and the development platform as the firmware contains a transmission protocol to ensure fault-free data transmission between the PC and the development platform.

3.1.1 RTS and DTR signals conflict between BSL Demo Board v2.0 and Keil µVision2 Monitor 51

RTS and DTR signals set by the Keil software keeps the BSL Demo Board v2.0 in a "Reset" status all the time. In order to block the RTS and DTR signals of the serial communication cable from reaching the board, please use the SoftICE adapter between the BSL Demo Board v2.0 and the serial cable.

4.0 WINDOWS SOFTWARE DESCRIPTION

The flowcharts in Figures 4-1 and 4-2 provide an operational overview of the SST FlashFlex51 software for the SST89C5x and SST89x564RD/SST89x554RC MCUs respectively. The memory mode includes two different types – Internal Memory Mode and External Memory Mode.

In External Memory Mode (the EA# pin of the target MCU is tied to a logic "low"), the IAP commands are executed from the external memory device to program block 0 and 1 of the MCU internal flash memory. For operations of Chip erase, Memory re-mapping and Set security-level, the user needs to switch to internal memory mode.

Hence, in Internal Memory Mode (the EA# pin is tied to a logic "high"), the IAP commands are executed from block 1 of on-chip flash memory.

Descriptions of the user interface are provided in Section 4.2.

4.1 Installing Windows Software

The BSL package includes a PC executable program and the MCU binary/Intel Hex code. The PC executable, EasyIAP11F.EXE, is a Windows-based application and runs directly under Window 95/98/NT/2000/Me/XP operating systems. Two additional MFC library files provided in this package, MFC42.DLL and MSVCRT.DLL, are usually located in the Windows System or System32 folder. The user needs to copy these two library files into the same folder as SSTFlashFlex51.EXE only if they don't exist or are not the latest revision codes.

The MCU binary/Intel Hex code can work with external crystal frequency range from 1 MHz through 33 MHz (40 MHz for SST89E554RC and SST89E564RD MCUs), the PC pre-settings for serial communication are: 38.4K/ 19.2K/9600/4800/2400/1200/600 baud, 8 data bits, 1 stop bit and no parity.

The BSL code, residing in block 1 flash, can be installed in three different ways:

1. by the factory,
2. by the user with SST Boot-Strap Loader (BSL) Demo Kit, or
3. by the user with a universal programmer that supports the SST microcontroller being programmed.

Please visit the SST website for the information on the SST BSL Demo Kit and a list of programmer vendors that support the FlashFlex51 family. To understand the fundamentals of IAP operations, the user can download the earlier version (BSL v1.1E) BSL source code, which is written in 8051-family assembly language.

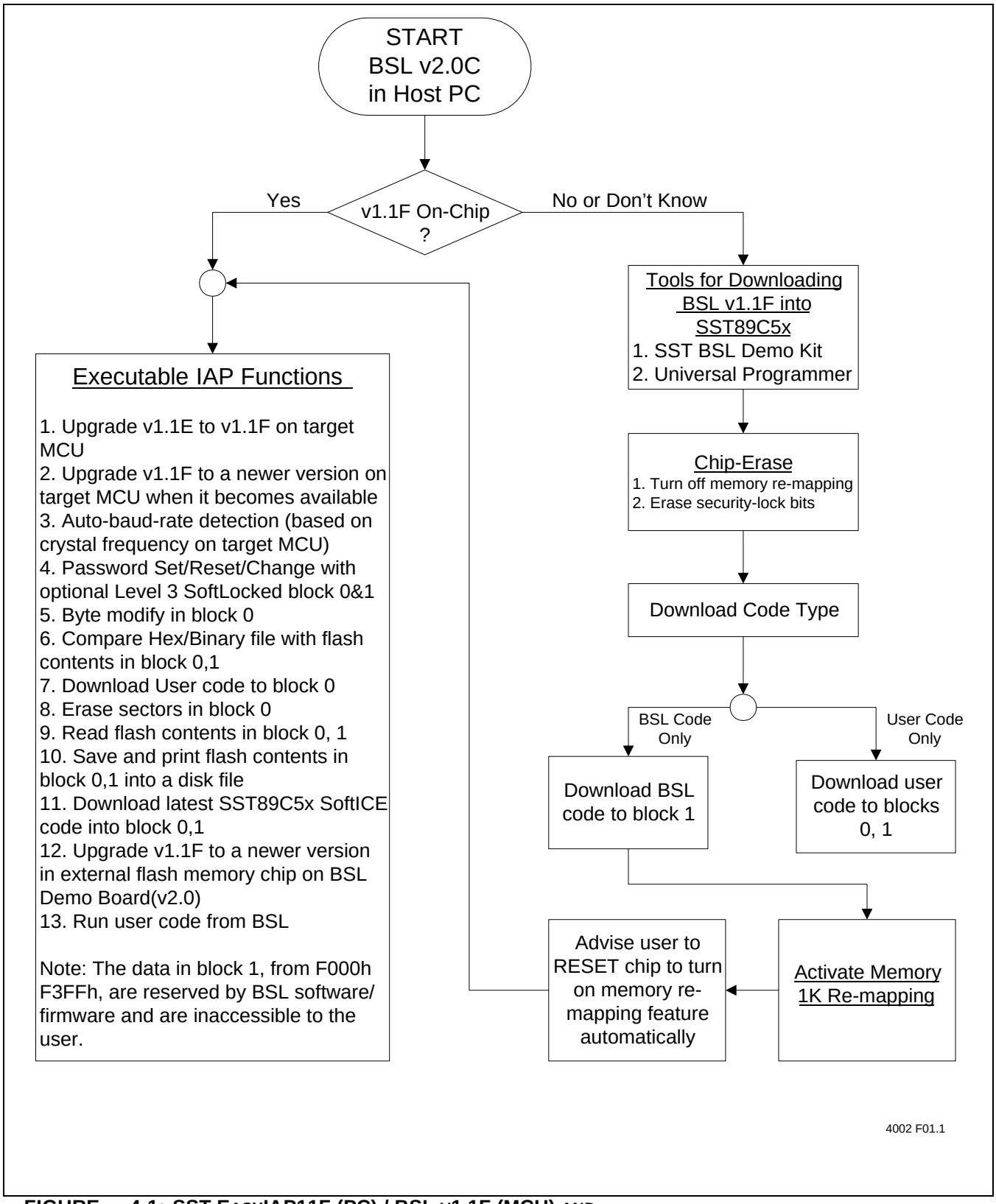


FIGURE 4-1: SST EASYIAP11F (PC) / BSL V1.1F (MCU) AND USAGE OF MEMORY RE-MAPPING IN THE SST89C5X MCU

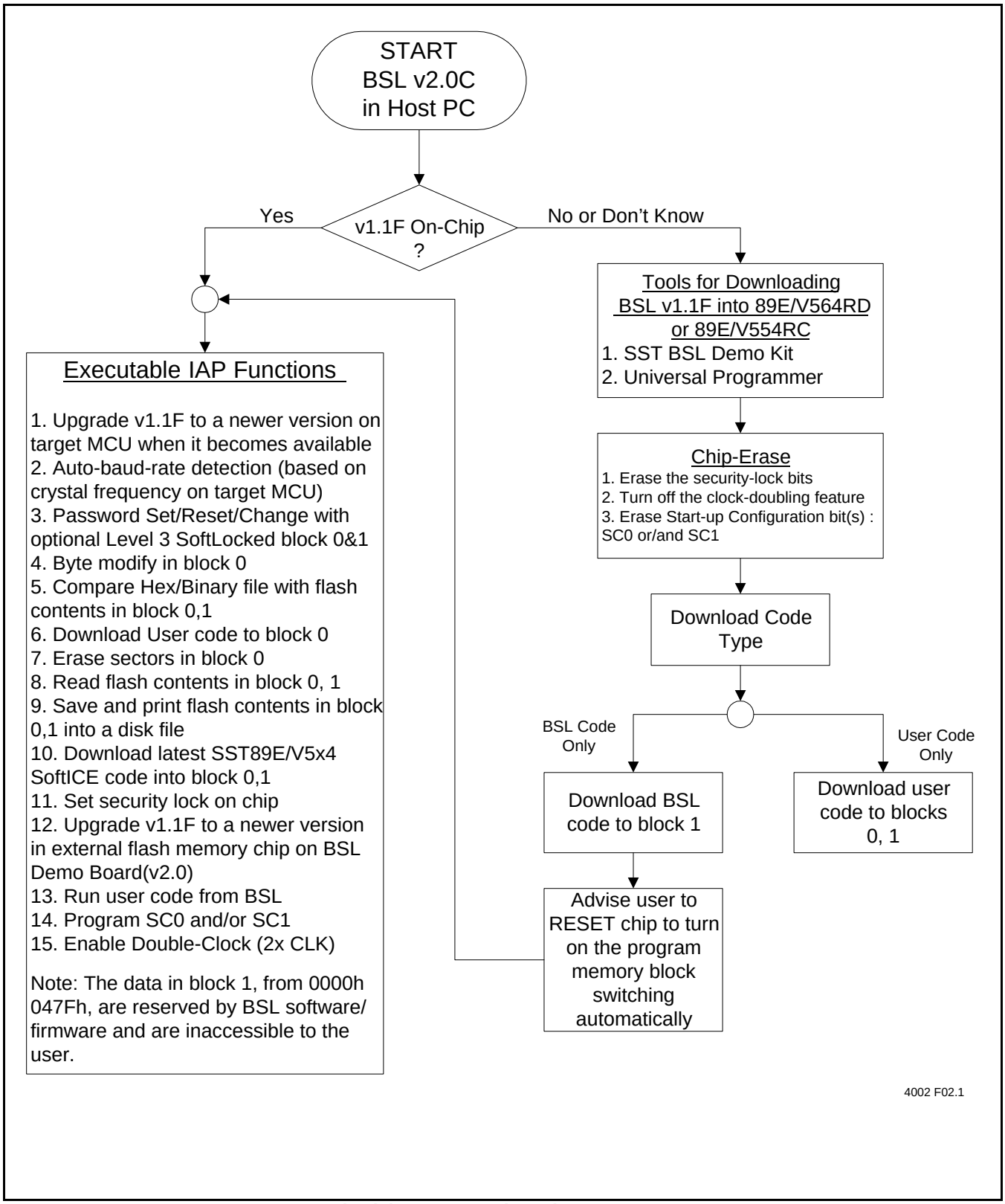
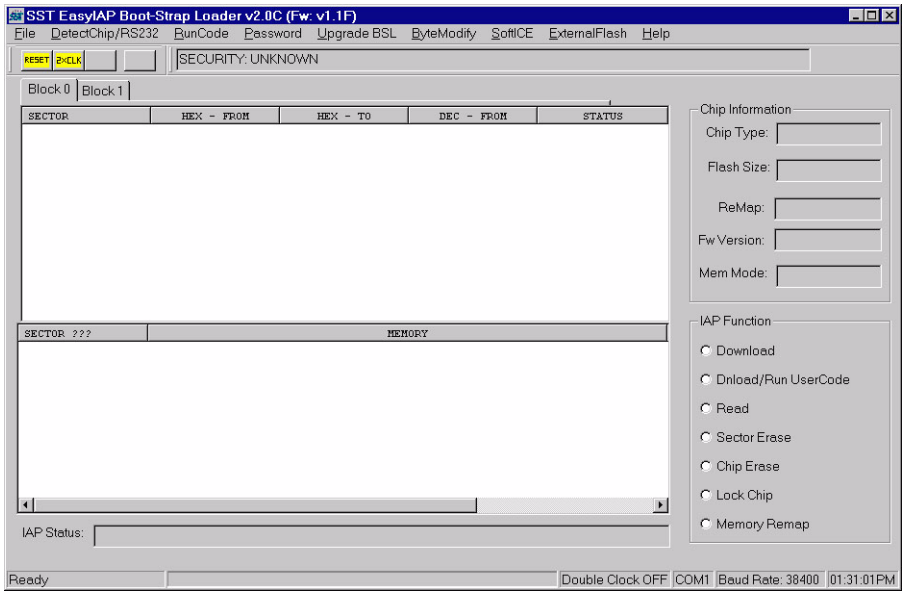


FIGURE 4-2: SST EASYIAP11F (PC) / BSL V1.1F (MCU) AND USAGE OF PROGRAM MEMORY BLOCK SWITCHING IN THE SST89E/V564RD OR THE SST89E/V554RC MCU

4.2 User Interface

4.2.1 Entry Menu

In the Entry Menu, all system parameters, such as COM port, Baud Rate, External Crystal Frequency, Memory Mode and Chip Type are automatically initialized to their default values.

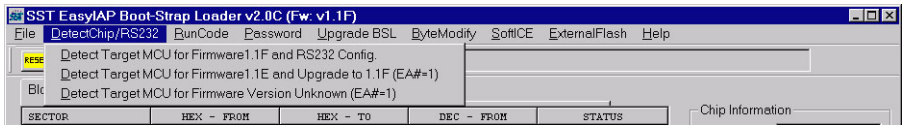


4.2.2 Detect Chip/RS-232 Configuration

Click on Help and About Detect Chip to view a brief description of this option.

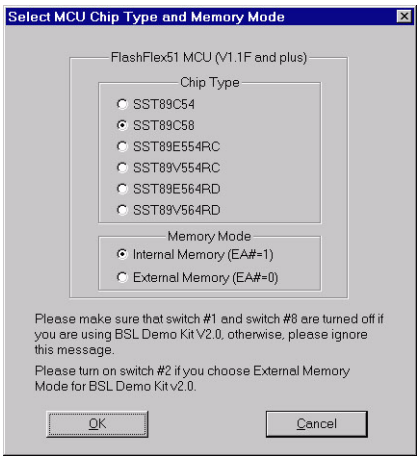
EasyIAP11F only detects firmware version 1.1E and later.

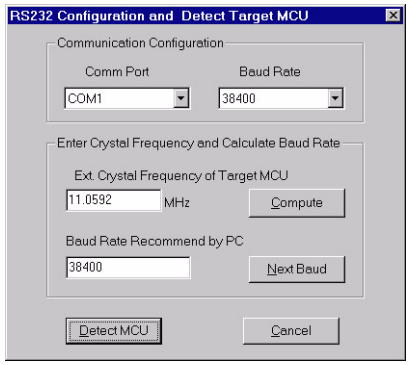
To begin using EasyIAP11F, click on DetectChip/RS232.



4.2.2.1 Detect Target MCU for Firmware 1.1F and RS232 Configuration

By clicking this submenu option, it will display the following windows. The user will have to choose the correct Chip Type and Memory Mode.

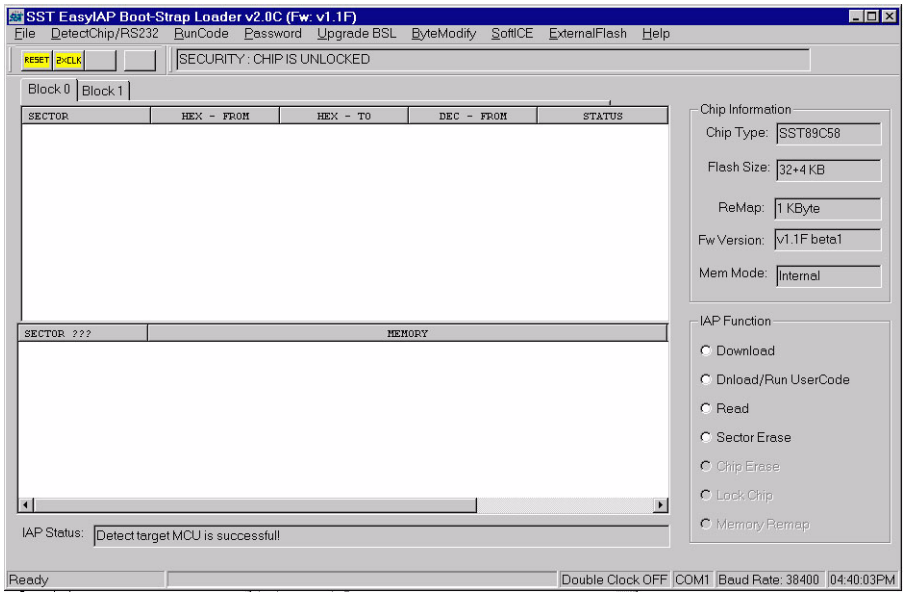




Also displayed are the default values of the COM Port, Baud Rate and Crystal Frequency. Users may enter the crystal frequency of the target MCU and let the program calculate the best baud rate by pressing the Compute button. The Next Baud button will calculate the 2nd, 3rd, 4th… best baud rate choices. Click the Detect MCU button once you have finished the COM port and Baud Rate configurations. Then click OK and reset the target MCU when a dialog box is shown.

Users of the new BSL Demo Board v2.0 do NOT need to manually push the reset button since it is done automatically by the PC.

Once the chip is detected successfully, the following window appears:



The RS-232 configuration is saved into a text file, SstBslComDft.txt, in the root directory of C drive. This saved configuration becomes the future default.

4.2.2.2 Detect Target MCU for Firmware 1.1E and Upgrade to 1.1F (EA# =1)

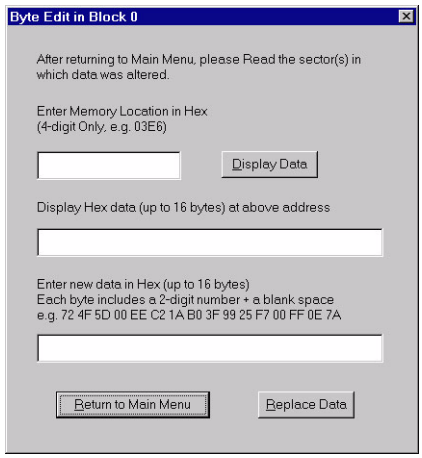
By clicking this submenu option, the software will detect for firmware 1.1E in internal memory mode. If detected, it will give the user the option to upgrade to 1.1F automatically without destroying user code in block 0. By deciding not to upgrade to 1.1F from 1.1E, the user must use the previous version of the software. EasyIAP11F does not support firmware 1.1E on any IAP function. This function can only be used under Internal Memory Mode.

4.2.2.3 Detect Target MCU for Firmware Version Unknown (EA# =1)

By clicking this submenu option, the software will detect for firmware 1.1F first, and if that detection fails, it will detect for 1.1E. If 1.1E has been detected, the user has the option to upgrade to 1.1F automatically without destroying user code in block 0. However, if the user decides not to upgrade to 1.1F from 1.1E, then the user has to close this application and use the former version of software. EasyIAP11F does not support firmware 1.1E on any IAP function. This function can only be used under Internal Memory Mode.

4.2.3 Byte Modify

Click on Help and About Byte Modify to view a brief description of this option.



This option allows the user to modify the flash content in block 0. The user needs to enter a 4-digit memory location (in hexadecimal format), then click on the Display Data button to display up to sixteen bytes of data starting from the address user entered. The modified data is entered into the lowest window. Click on Replace Data button to complete the data modification.

4.2.4 Chip Erase

This option is only allowed in External Memory Mode. Chip-Erase will erase all the contents of both internal memory blocks.

4.2.5 External Flash Update

With the new demo board design (v2.0), downloading code to external flash, reading from external flash, erasing data from external flash, as well as running code at external flash are possible. **Please check your demo kit version. If it is v2.0 or greater, then the external flash can be updated; otherwise this feature is not applicable if you are using demo board v1.0.**

SST39SF010A Flash Memory has 32 sectors and each sector is 4KB for a total of 128KB of memory. It is divided it into low 64K and high 64K. External BSL1.1F firmware is normally downloaded to 0x0000 of the low 64K. So when the user chooses External Memory Mode, external BSL1.1F will run from the first sector of the low 64K of external flash. The user can download their code to the high 64k of external flash and keep the low 64K of the BSL1.1F firmware unchanged.

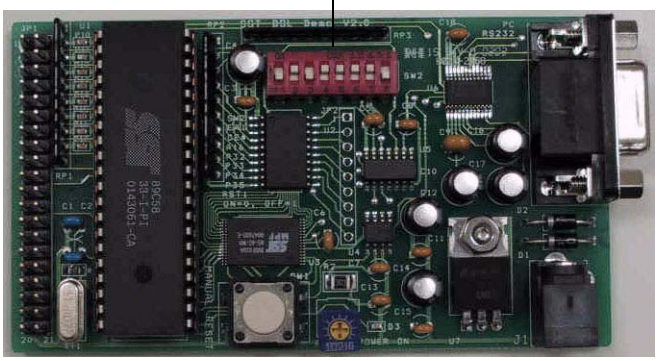
The first sector of low 64K, where external BSL resides, is not allowed to be erased. The user may check the SST website from time to time and download the latest version of SSTEasyIAP. External BSL will be updated by choosing "Upgrade External BSL Firmware to Newer Version" under the "Upgrade BSL" menu.

Please note that external flash is not accessible if block 0 of SST89C5x is hard locked.

To access the low 64K of external flash memory, position the toggle switches as indicated below.

To access low 64K:
Switch #1 to OFF
Switch #2 to ON
Switch #3 to ON
Switch #8 to OFF

To access the high 64K of external flash memory, position the toggle switches as indicated below.



To access high 64K:
Switch #1 to OFF
Switch #2 to ON
Switch #3 to OFF
Switch #8 to OFF

4.2.6 File Compare

To compare an Intel hex or a binary file with the contents in internal (block 0/1) or external (low/high 64K) flash memory, the user needs to click on the File menu option, then open the Compare option. Enter or select a filename, select the starting address (in Range list box), then click OK. The result of the comparison is shown in the dialog box (labeled as IAP Status) – the text of "File Compare OK!" is for a matched comparison or the text of "Unmatched data at memory address xxxxh: xxh (MCU) vs. xxh (File)" is for an unmatched comparison.

4.2.7 Download

User code can be downloaded into block 0 and block 1 of MCU internal flash and high 64K of external flash. To start the file downloading, click on the Download button under IAP Function, select the appropriate File Name, e.g. BINCTR.HEX, and Starting Sector (e.g. 0000H), then click on OK.

Prior to downloading, the sectors in flash memory, which match the code size, are erased completely. Consequently, the program warns the user and asks whether the download is to proceed or not. Click on Yes to proceed or No to quit. To search for the file, the user can click on the "..." button located at the right end of File Name box.

4.2.8 Download/Run User Code

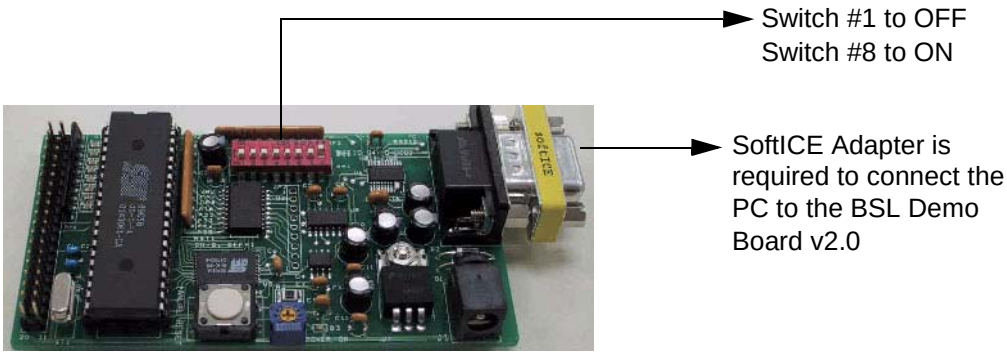
This function combines Download and Run User Code into one. The Download/Run-user-code command automatically runs user code after it downloads user code flash memory.

4.2.9 Download SoftICE

SoftICE will use the first 1K in block 1 and last 4K in block 0 (for SST89C5x), or first 4K in block 1 and last 1K in block 0 (for SST89x554RC and SST89x564RD). Please note that downloading SoftICE will erase the BSL code in block 1. Once SoftICE has been downloaded into the MCU, users are able to use it to debug their code.

The following is a brief description on setting up SST SoftICE:

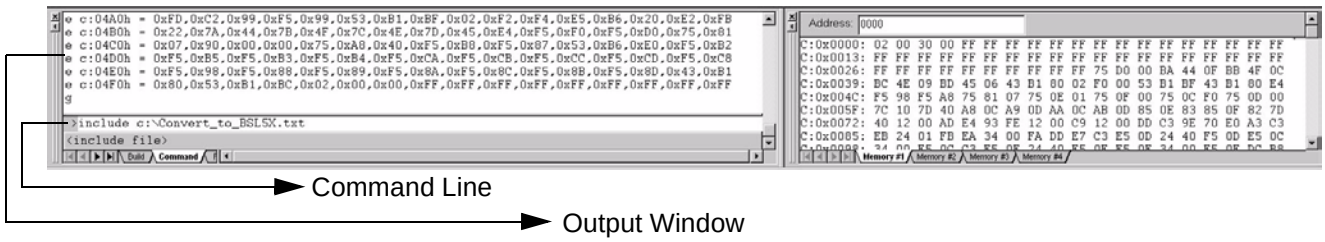
1. Use SoftICE with SST BSL Demo Board v1.0: switch JP2 jumper to select internal mode and reset the board.
2. Use SoftICE with SST BSL Demo Board v2.0: as shown in the figure below, position switch#1 to OFF and switch#8 to ON. Use the SST SoftICE adapter to connect between the BSL Demo Board v2.0 and DB9 cable to the PC.



Users can also convert back to the BSL from SoftICE through the Keil Monitor-51. In the command line (click Output Window in View Menu) in the Keil GUI, type "include c:\Convert_to_BSL5x.txt" for SST89C5x, "include c:\Convert_to_BSLx564.txt" for SST89x564RD, provided the script file is saved under root directory of C drive. Refer to the picture below for an example of entering on the command line.

Next press Enter. SST SoftICE will load Patch_E.hex to block 0 and run Patch_E.hex to load back the BSL firmware. A "g" appears in the bottom of the Output Window above the command line to indicate that loading Patch_E.hex to block 0 is completed. Wait around 2s before stopping your debug session, and close the Keil GUI. Now the user has converted back to the BSL from SoftICE, but the remaining code in block 0 needs to be cleared by the user.

Please refer to Appendix E for detailed information on how to use SST SoftICE.



4.2.10 Lock Chip

Most lock levels are only allowed in External Memory Mode for the C54/C58, except for SoftLock block 0 and block 1 (PROG-SB2) which could be done after a password is set/reset. But all level locks are allowable in Internal Memory Mode for the 89E/V554RC and 89E/V564RD. EasyIAP11F will not allow user to choose Level 4 hard lock block 0 and block 1 for any type of MCU. Only a Chip-Erase can remove any level of security lock.

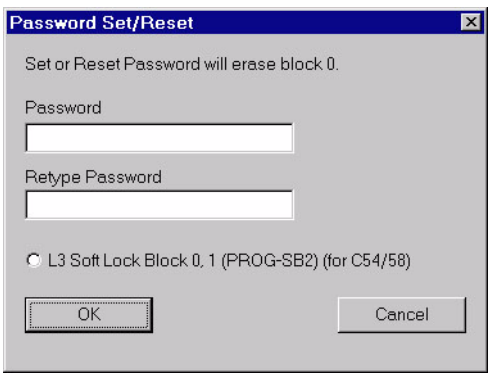
4.2.11 Memory Remap

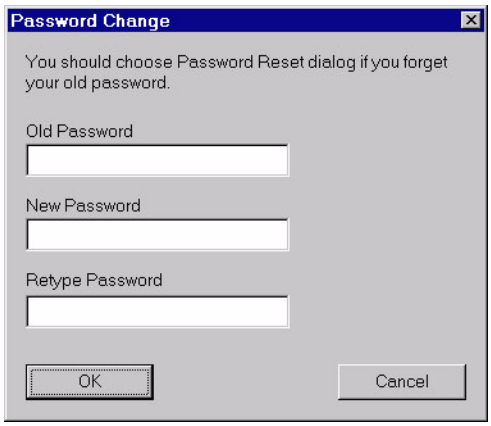
The user can choose to remap to 1K, 2K and 4K for SST89C5x. For the SST89x564RD, user can program SC0, and for the SST89x554RC, user can choose to program SC0 and/or SC1.

4.2.12 Password Set/Reset/Change

The Password feature protects user code from being read by SSTEasyIAP software, while security lock protects code from being read by a universal programmer.

Click on Help and About Password to view a brief description of this option.





Password only works in internal memory mode.

Choosing Password Set/Reset will always erase block 0 entirely. This will protect the user's code from being exposed to an unauthorized user. The User has the option to Soft Lock blocks 0 and 1 after a password is set/reset, this option is for SST89C54/58 only. For best protection, it is highly recommended that the user combine both password and security-lock features in their application code.

If you upgrade firmware from 1.1E to 1.1F, or you are using 1.1F for the first time, or you have not set a password previously, then you have no password set yet. So when you are requested to provide password in order to use IAP functions, you just leave the password field blank, and click on OK. When users utilize a 3rd party programmer to load BSL1.1F firmware into the MCU, they should choose to fill unimportant bytes with "00" or "FF", so that the sector which is used to store the password will be filled with "00" or "FF". This allows the user who has not set a password to do so using IAP. BSL firmware will think a password has been set if the password sector has something other than all "00" or all "FF."

Firmware 1.1F and later will not allow user to use other software to read from or write to the MCU once a non-blank password has been set.

Changing password will not destroy user's code in block 0.

4.2.13 Print Memory Contents

The Print option from File allows the user to send memory data from internal or external flash memory to a printer.

4.2.14 Read

This function reads the code from block 0, block 1 or external flash memory, then displays the contents in hex and ASCII format. The procedure is: Click on the Read button, select the starting address and range of sectors, then click on OK. Sector status can be any one of four conditions – All Zeros, Blank, Not Blank or Unknown. The unread sectors correspond to the Unknown status.

4.2.15 Reset Target MCU/Clear Chip Info

Clicking on the yellow RESET toolbar button erases the Chip Information on the screen and sends a RTS and DTR signal to the target MCU through the RS232 serial port.

The RTS/DTR signal provides a single level with standard RS232 signal amplitude (from -12 V to +12V). The duration of RTS signal is about 40 milliseconds before it raises from -12V to +12V. On the target MCU board, the user can optionally transform this RTS signal into a TTL/CMOS signal and use it to reset the target MCU. RTS/DTR signal sent by PC will not be able to control the BSL Demo Board v1.0. On the BSL Demo Board v2.0 with Switch#1 & #8 turned to the OFF position, RTS and DTR signals sent by PC will control the RST and EA# signals of the demo board directly.

4.2.16 Run User Code

This function resets all SFRs in MCU chip to their default values, and then executes user code at address 0000h in block 0 internal flash memory, or 0000h/10000h in low/high 64K of external flash memory.

4.2.17 Save Data into a File (Upload)

To save the contents of block 0/1 of internal flash or low/high 64K of external flash into a binary/text file, the user needs to click on File menu option, then open the Save option. Next the user enters a filename, chooses the type of file (binary or text file), selects the starting address and number of sectors (in Range list box), and then clicks OK. Click OK when the message "Save data has completed" appears.

4.2.18 Sector Erase

This option enables the user to select the region of internal/external flash memory to be erased. The user enters the starting address and number of sectors (in Range list box) to be erased, then clicks OK. Click OK when the message "Sector erase completed!" appears.

4.2.19 Update BSL Code in SST89C5x, SST89x554RC and SST89x564RD MCU

This feature works in internal memory mode only. It will only upgrade firmware v1.1F or later to a newer version. Upgrading BSL will not destroy user's code in block 0.

Users should download the latest version of BSL firmware from SST's website as the source code for updating, then save it to local hard drive.

4.2.20 Double Clock

The crystal frequency of the target board will be doubled by clicking this button, and applies only to the SST89x554RC and SST89x564RD MCUs. Afer Double Clock is set, the user can enter twice the original crystal frequency to gain the higher baud rate (maximum 38400 bps) during chip detection.

4.3 Self-detection of Serial Link

The software can detect whether the serial link is alive or not in about ten seconds. After either a disconnection of the serial link or an interruption of DC power, the software issues a warning message and clears the chip information on the screen.

5.0 MCU DEMO SOFTWARE

Three demo software programs have been supplied with the SSTFlashFlex51.EXE, and the intent here is to provide the user with some understanding of their basic functionality. Any one of the three can be downloaded from the PC to the user platform or the BSL Demo Board, and executed there. For a visible presence on the demo board/platform, each of the demo routines manipulates the board LEDs in some manner. The source (.A51) and download (.HEX) files for each of the three demo programs can be downloaded from SST's website.

The file names are:

1. TWOBALL.A51/.HEX
2. BINCTR.A51/.HEX
3. PENDB.A51/.HEX

5.1 TWOBALL

The "twoball" routine corresponds to a two-ball bouncing ball sequence, that is, the two most significant LEDs will light up and proceed to shift up one LED bit position at a specific time interval. When the lighted pair reaches the two least significant bit positions, then they will begin to shift down in the same manner. The up-down sequence will be continuous.

5.2 BINCTR

The "binctr" routine shows a binary counting sequence on the LEDs, which are changing at a specific time interval.

5.3 PENDB

The "pendb" software routine causes the LEDs to behave similar to a pendulum. Like the previous routines discussed, the LEDs are changing at a specific time interval visible to the user.

For technical support via email, please contact the SST Technical Support Hot Line: support@sst.com. In the future, check the SST website (www.SST.com) under FlashFlex51 Microcontrollers for information on the available downloadable versions of the BSL code.

APPENDIX A. FILE NAME CONVENTION

The BSL file name convention has been adopted to accommodate improved software versions, additional chip types and frequencies, memory mode and custom boot-strap loaders for the FlashFlex51 MCU family. The file name format is:

File Name:	F51xBLyz.ext	
where	x = E	External-memory-resident code, e.g. SST39SF010A MPF, in External Memory Mode
	y =	Downloader Version
	x = P	MS-DOS, PC-resident code
	y =	BSL Version Number
		A = Version 1.0
		B = Version 1.1/ 2.0
		C = Version 1.2
	x = M	MCU-resident code in Internal Memory Mode
	y =	Chip Type and Frequency
		E = C58 11.0592 MHz
		F = C58 12 MHz
		G = C54 11.0592 MHz
		H = C54 12 MHz
		I = designate to v1.1F
	x = C	Custom MCU-resident code
	y =	Customer ID
		A = Infronex C58 7.3728 MHz
		B = (next customer)
	z =	Revision No.
		0 = Original Release
		1 = First Revision

APPENDIX B. LIST OF FILES

The BSL package now consists of three parts:

- For the Windows 95/98/NT/2000/Me/XP-resident software, an executable file (SSTFlashFlex51.EXE) is supplied.
- For the MCU-resident code, an Intel hex file (.HEX) and a binary file (.BIN) are furnished.
- For the external memory-resident code, an Intel hex file (.HEX) is furnished.

Table B-1 lists the files that can be downloaded from the SST web site.

TABLE B-1: LIST OF EASYIAP11F (PC) / BSL V1.1F (MCU) / V1.1F (EXTERNAL MEMORY CHIP) FILES

Chip Type	Ext. Crystal Frequency	Baud Rate	PC Files	MCU/Ext. Memory Files
SST89C54/C58 MCU	1 – 33 MHz (5V) 1 – 12 MHz (2.7V)	38.4K/19.2K/9.6K/ 4.8K/2.4K/1.2K/600	SSTEasyIAP11F.exe	F51MBLI5.HEX[1] F51MBLI5.BIN[1]
SST89E/V554RC MCU	1 – 33 MHz (V554RC) 1 – 40 MHz (E554RC)	38.4K/19.2K/9.6K/ 4.8K/2.4K/1.2K/600	SSTEasyIAP11F.exe	F51MBLL5.BIN[1]
SST89E/V564RD MCU	1 – 33 MHz (V564RD) 1 – 40 MHz (E564RD)	38.4K/19.2K/9.6K/ 4.8K/2.4K/1.2K/600	SSTEasyIAP11F.exe	F51MBLL5.BIN[2]
SST39SF010A MPF or equivalent	1 – 40 MHz	38.4K/19.2K/9.6K/ 4.8K/2.4K/1.2K/600	SSTEasyIAP11F.exe	F51EBLK5.HEX

TB-1.1 4002

1. Binary file should be downloaded into block 1 and starts at address F000h for SST89C5x and E000h for SST89x554RC. Hex file needs to be downloaded into block 0 and starts at address 0000h, both should be downloaded in external memory mode.
2. Both binary file and hex file should be downloaded into block 1 and starts at address 0000h using external memory mode.

APPENDIX C. BSL DEMO BOARD V2.0 SWITCH FUNCTION

TABLE C-1: THE FUNCTION OF INDIVIDUAL SWITCH ON NEW BSL DEMO BOARD (V2.0)

Position	Function		
1	EA1	OFF=1 ON=0	See Table C-2 and Table C-3 below
2	CE#	OFF=1	Disable External Flash SST39SF010A
		ON=0	Enable on-board SST39SF010A
3	A16	OFF=1	Select upper 64K of SST39SF010A
		ON=0	Select lower 64K of SST39SF010A
4	P3.2	OFF=1	User definable switch's function
5	P3.3	ON=0	
6	P3.4		
7	P3.5		
8	RST1	OFF	RST is controlled by PC software
		ON	RST is controlled by MAX706 with Manual Reset button

TC-1.0 4002

TABLE C-2: NEW BSL DEMO BOARD IS RUNNING STANDALONE WITHOUT PC'S CONTROL AT ALL:
NOTE: Disconnect cable between PC and Board v2.0

RST1 must be ON, MCUs RST is controlled by on board MAX706 with Manual Reset button			
EA1	**A16**	**CE#**	**Effects**
0	0	0	Run external lower 64K of SST39SF010A
0	1	0	Run external upper 64K of SST39SF010A
1	0	0	Run internally, lower 64K as data memory
1	1	0	Run internally, upper 64K as data memory
0	0	1	Illegal combination of settings
0	1	1	Illegal combination of settings
1	0	1	Run internally, no data memory, SST39SF010A is disabled
1	1	1	Run internally, no data memory, SST39SF010A is disabled

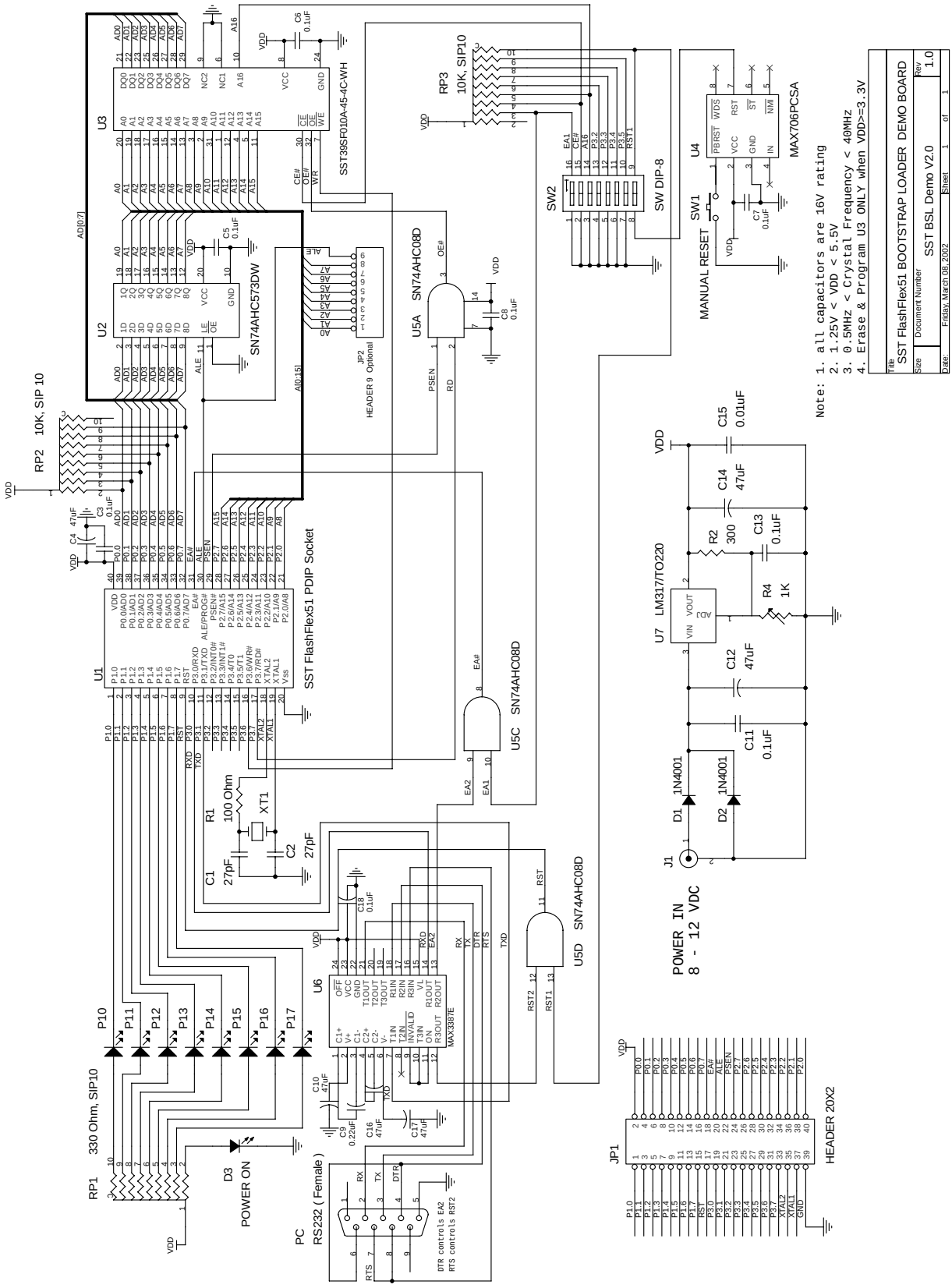
TC-2.0 4002

TABLE C-3: NEW BSL DEMO BOARD IS RUNNING UNDER CONTROL OF PC SOFTWARE:

Both RST1 and EA1 must be set OFF, MCUs RST and EA# is controlled by PC software				
RST2				
1	Reset MCU, don't care EA2, A16 or CE#.			
0	MCU is running, mode is determined by following chart:			
	EA2	**A16**	**CE#**	**Effects**
	0	0	0	Run external lower 64K of SST39SF010A
	0	1	0	Run external upper 64K of SST39SF010A
	1	0	0	Run internally, lower 64K as data memory
	1	1	0	Run internally, upper 64K as data memory
	0	0	1	Illegal combination of settings
	0	1	1	Illegal combination of settings
	1	0	1	Run internally, no data memory, SST39SF010A is disabled
	1	1	1	Run internally, no data memory, SST39SF010A is disabled

TC-3.0 4002

APPENDIX D. BSL DEMO BOARD V2.0 SCHEMATIC



APPENDIX E. SST89C5X/SST89X554RC/SST89X564RD SOFTICE USER'S GUIDE

SST89C5x / SST89x554RC / SST89x564RD SoftICE User's Guide

OVERVIEW

Introduction

SoftICE stands for Software In-Circuit Emulator. It is an in-circuit development tool for SST customers to debug their applications in SST89C5x, SST89x554RC, and SST89x564RD MCUs.

The purpose of this document is to provide a hands-on reference guide for users of the SoftICE software utility tool. It also lists SoftICE features and shows the users how to install SoftICE in their system.

Scope

The scope of this document is limited to discussing SoftICE features, target hardware requirements, installing and using SoftICE, and user code restrictions.

References

TABLE 1: FILE AND SOFTWARE REFERENCES

Title	Location
1. KEIL 8051/251 Evaluation Kit	
Setup software	CD-ROM\Keil\setup.exe
Getting Started with µVision 2	CD-ROM\Keil\Keil\C51\Hlp\Gs51.pdf
2. IAP demo program	
demoIAP_5x	CD-ROM\SST89C5x\SoftICE\demoIAP_5x
demoIAP_554	CD-ROM\SST89x554\SoftICE\demoIAP_554
demoIAP_564	CD-ROM\SST89x564\SoftICE\demoIAP_564
3. SST Boot-Strap Loader Software Example User's Guide	CD-ROM\User's Guides\Boot-Strap Loader Software Example User's Guide
4. SST89C5x SoftICE	
SoftICE54.hex	CD-ROM\SST89C5x\SoftICE\ SoftICE54.hex
SoftICE58.hex	CD-ROM\SST89C5x\SoftICE\ SoftICE58.hex
5. SST89x554RC SoftICE	
SoftICE554.hex	CD-ROM\SST89x554\SoftICE\ SoftICE554.hex
6. SST89x564RD SoftICE	
SoftICE564.hex	CD-ROM\SST89x564\SoftICE\ SoftICE564.hex

T1.0 4002

Software/Documentation Updates

For the latest SST software and documentation updates, visit the SST web site at www.SST.com or www.Super-Flash.com.

For the latest Keil software and documentation updates, visit Keil's web site at www.keil.com.

SOFTICE FEATURES

SST89C5x/SST89x554RC/SST89x564RD SoftICE communicates with the KEIL µVision2 Debugger using one of the PC's COM ports. It helps debug the target programs in real-time, thus provides engineers using SST89C5x / SST89x554RC / SST89x564RD with an in-circuit development tool, which is simple, effective and easy to use. Small and compact though, SoftICE has most of the features of sophisticated hardware emulators. The features supported by SoftICE together with KEIL µVision2 Debugger are as follows:

- Download Intel HEX files.
- Source code debugging supporting both assembler and C51 high level language.
- Disassemble the code area into 8051 mnemonics.
- In-line Assembler.
- STEP
- STEP OVER
- Set/Remove Break Point (up to 10 fixed and 1 temporary Break Point)
- GO to Break Point/Cursor
- Read/Write Data memory
- Read/Write Code memory
- Read/Write SFRs
- Read/Write Ports
- SST-specific In-Application Programming (IAP)

TARGET HARDWARE REQUIREMENTS

SST89C5x/SST89x554RC/SST89x564RD SoftICE requires the following hardware components:

- Serial interface with timer 2 as baud rate generator.
- 8 byte stack space
- 5 KByte code memory is used by SoftICE firmware.

For SST89C58: 1 KByte in Block1, from F000h to F3FFh.
4 KByte in Block0, from 7000h to 7FFFh.

For SST89C54: 1 KByte in Block1, from F000h to F3FFh.
4 KByte in Block0, from 3000h to 3FFFh.

For SST89x554RC: 4 KByte in Block1, from 0000h to 0FFFh.
1 KByte in Block0, from 7C00h to 7FFFh.

For SST89x564RD: 4 KByte in Block1, from 0000h to 0FFFh.
1 KByte in Block0, from FC00h to FFFFh.

All other on-chip resources can be used by the application.

A typical connection between SoftICE and KEIL µVision2 Debugger is shown in Figure 1.

Make sure the device is in Internal Mode, that is, EA# (pin 31) is at a logic "high".

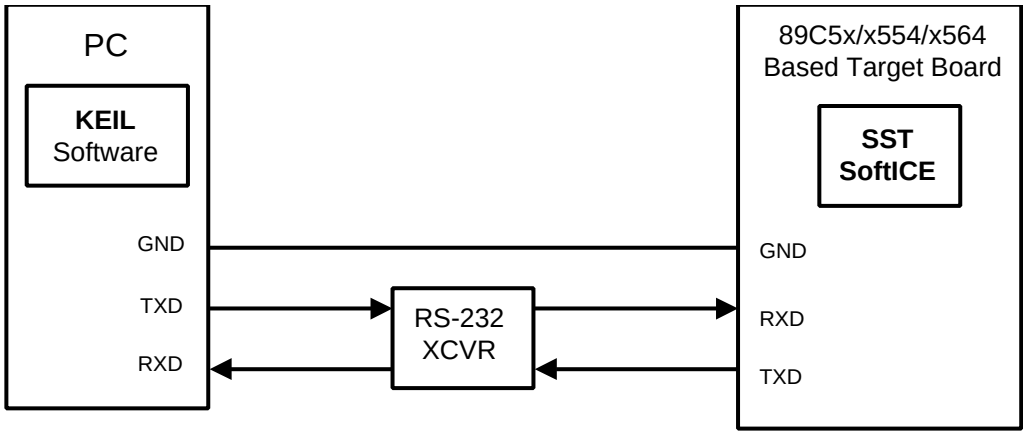


FIGURE 1: HARDWARE CONNECTION

The connection between SST BSL Demo Board v2.0 and PC is shown in Figure 2. Use the SoftICE adapter between the cable and the demo board.

Make sure switches 1 and 2 are OFF and switch 8 is ON.

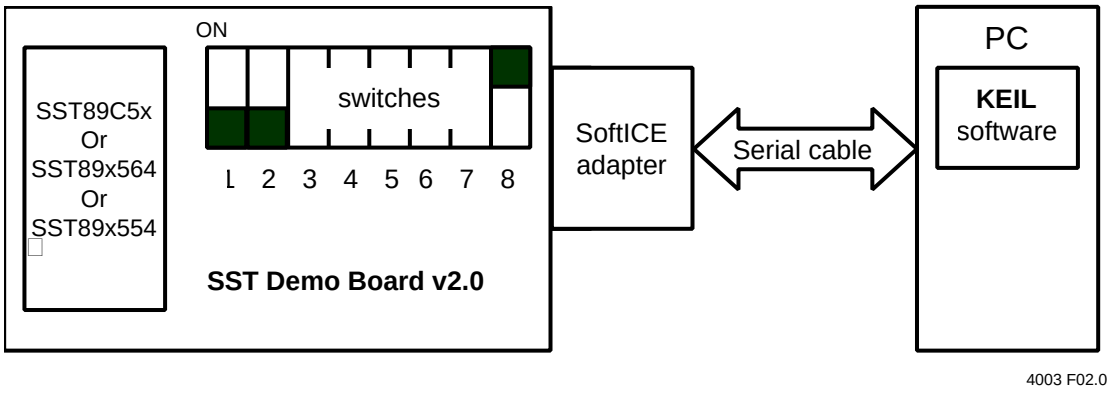


FIGURE 2: CONNECT SST DEMO BOARD AND PC

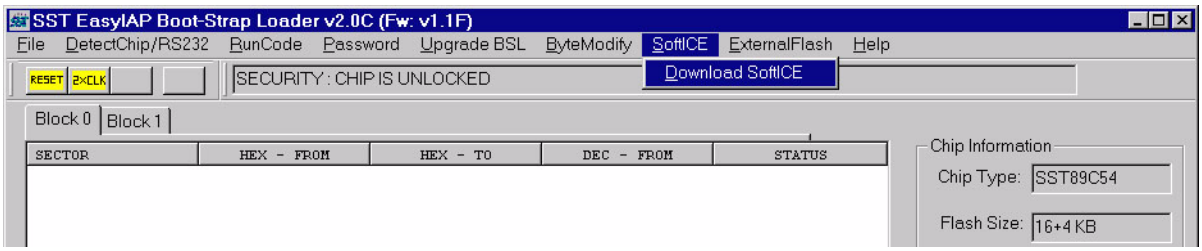
INSTALLING SOFTICE

Install SoftICE from SST Boot-Strap Loader

SoftICE firmware can be loaded into MCU directly from the SST Boot-Strap Loader.

After the device type is detected in Internal Memory Mode, SoftICE firmware can be downloaded by clicking on "Download SoftICE" in the SoftICE menu. SST Boot-Strap Loader will download the specific SoftICE for the device detected. Be aware that once the SoftICE firmware is downloaded, the SST Boot-Strap Loader located in block 1 will be replaced by SoftICE firmware.

For detailed information on how to use the SST Boot-Strap Loader, please refer to the *SST Boot-Strap Loader Software Example User's Guide* (Reference 3).



Install SoftICE by programmer

SoftICE firmware can also be loaded into the MCU by a 3rd party programmer.

- For SST89C54: Load SoftICE54.hex (Reference 4) to SST89C54, and remap 1k.
- For SST89C58: Load SoftICE58.hex (Reference 4) to SST89C58, and remap 1k.
- For SST89x554RC: Load SoftICE554.hex (Reference 5) to SST89x554RC, block 1.
  Do not program SC0 (Prog-SC0) or SC1 (Prog-SC1) bits, so that after reset,
  the MCU will always start to execute SoftICE.
- For SST89x564RD: Load SoftICE564.hex (Reference 6) to SST89x564RD, block 1.
  Do not program SC0 (Prog-SC0) bit, so that after reset, the MCU will always start to execute SoftICE.

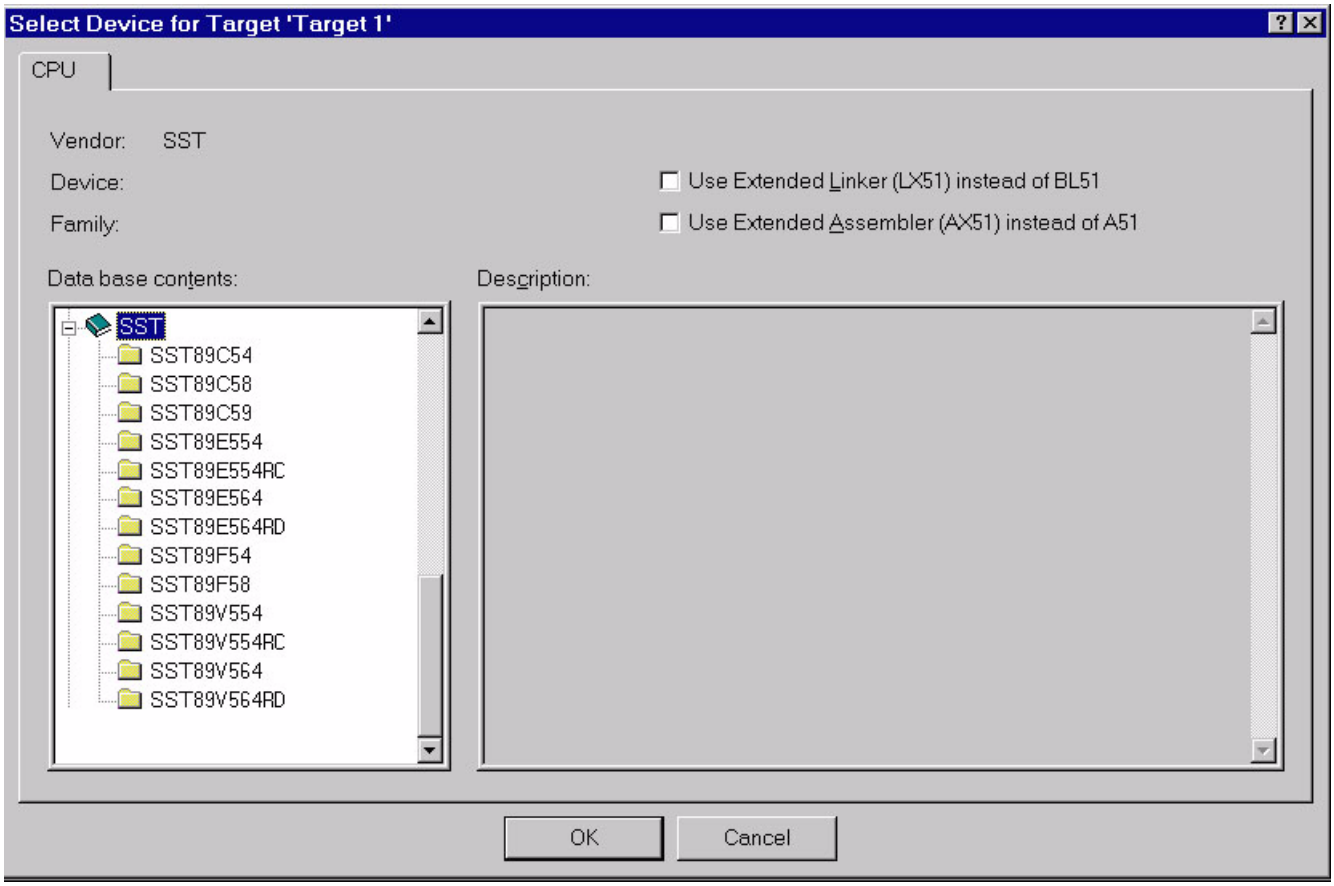
After SoftICE firmware is loaded into MCU and Keil software installed (Reference 1, setup software), connect the target board with PC as shown in Figure 1. When power-on or reset, the SoftICE is ready to communicate with Keil software to debug the user program. Note that when using Keil software demo version, user code size is limited to 2 KByte.

HOW TO USE SOFTICE

Step 1. Create project and compile user program

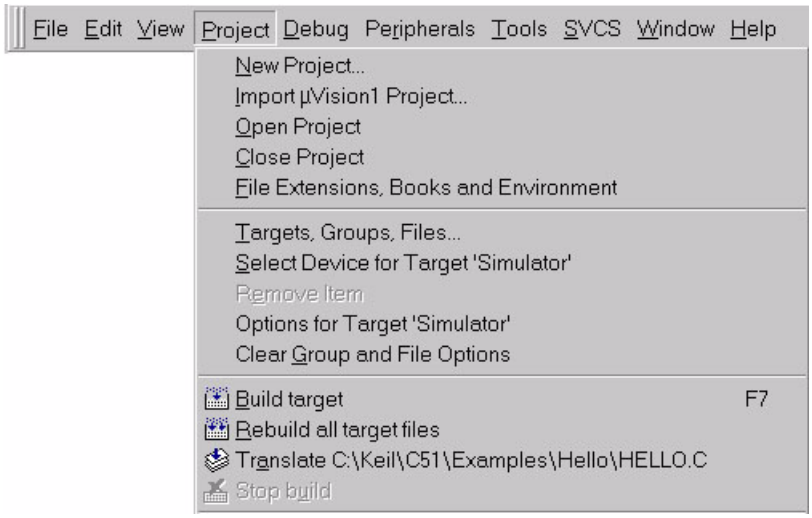
Create project and compile user program. For detailed instructions on how to create a project and compile a user program, please refer to *Getting Started with µVision2* (Reference 1).

Select one of the SST MCUs (e.g.SST89C54, SST89C58, SST89x554RC or SST89x564RD) as the target device.

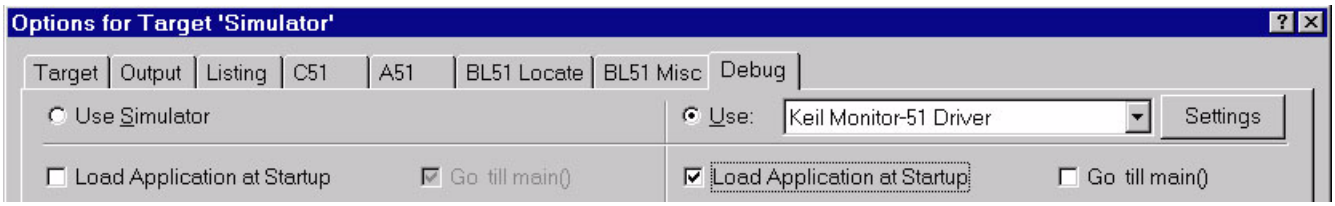


Step 2. Configurations

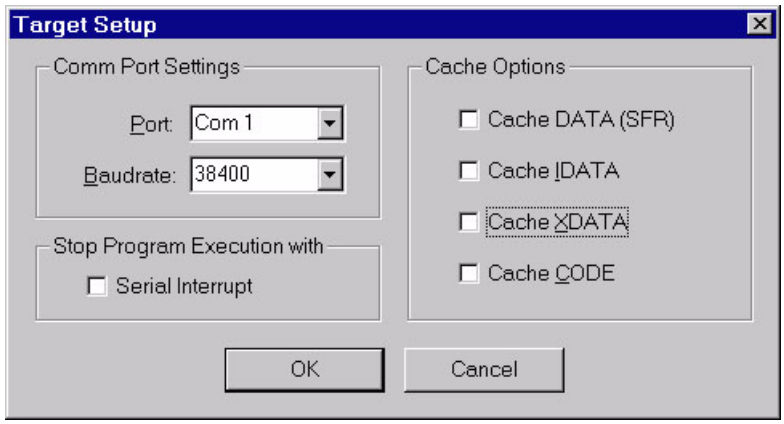
Select Options for Target 'Target 1' From KEIL user interface.



In the Options for Target 'target 1' window, select Use Keil Monitor-51 Driver in Debug tag. Note that the user's code area will be erased each time when the target board is reset. So Load Application at Startup should be selected also, if the user program needs to be loaded into the target board at startup.



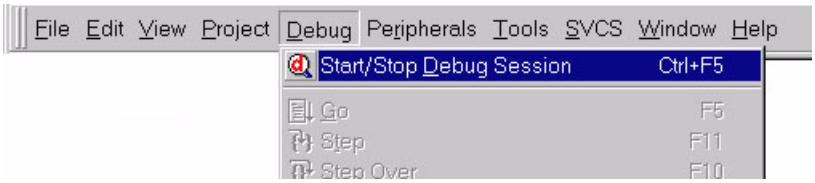
Click the Settings button, and the Target Setup window pops up for configuration. Select the PC serial interface (COM1-COM4) and the baud rate for the PC to be able to communicate with the target board. Don't select Cache Options if real time displaying of the memory window is needed. Note that if Stop Program Execution with Serial Interrupt is selected, SoftICE will modify three bytes at the serial port interrupt vector location c:0023h. Be sure that the user program does not use these code locations (c:0023-0025h).



The above setup can also be modified during the user code debugging process. The Configuration dialog box can be accessed by selecting Target Settings, from the Peripherals menu.

Step 3. Start Debugging Session

Start the debugging session by clicking the Start/Stop Debug Session in the Debug menu.



For detailed instructions on how to use KEIL µVision2 Debugger, please refer to *Getting Started with µVision2* (Reference 1).

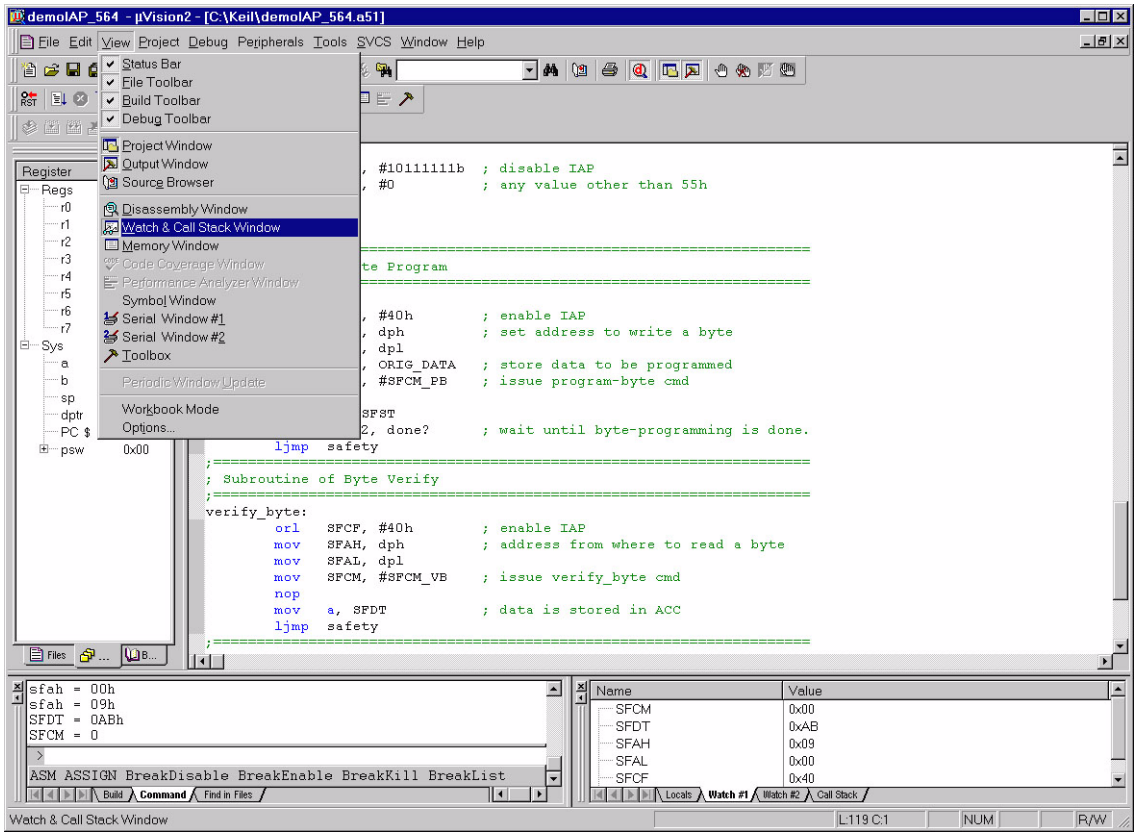
Tips for Debugging SST-specific In-Application Programming

Read/Write SST-specific SFRs

To display SST-specific SFRs, just type in the SFR name in the Watch Window. The Watch Window can be activated by clicking Watch & Call Stack Window from the View menu, and the current value of the SST-special SFRs will appear.

SST-specific SFRs can be modified through Output Window (Command Window in earlier version of Keil) in the View menu. See the screen image below.

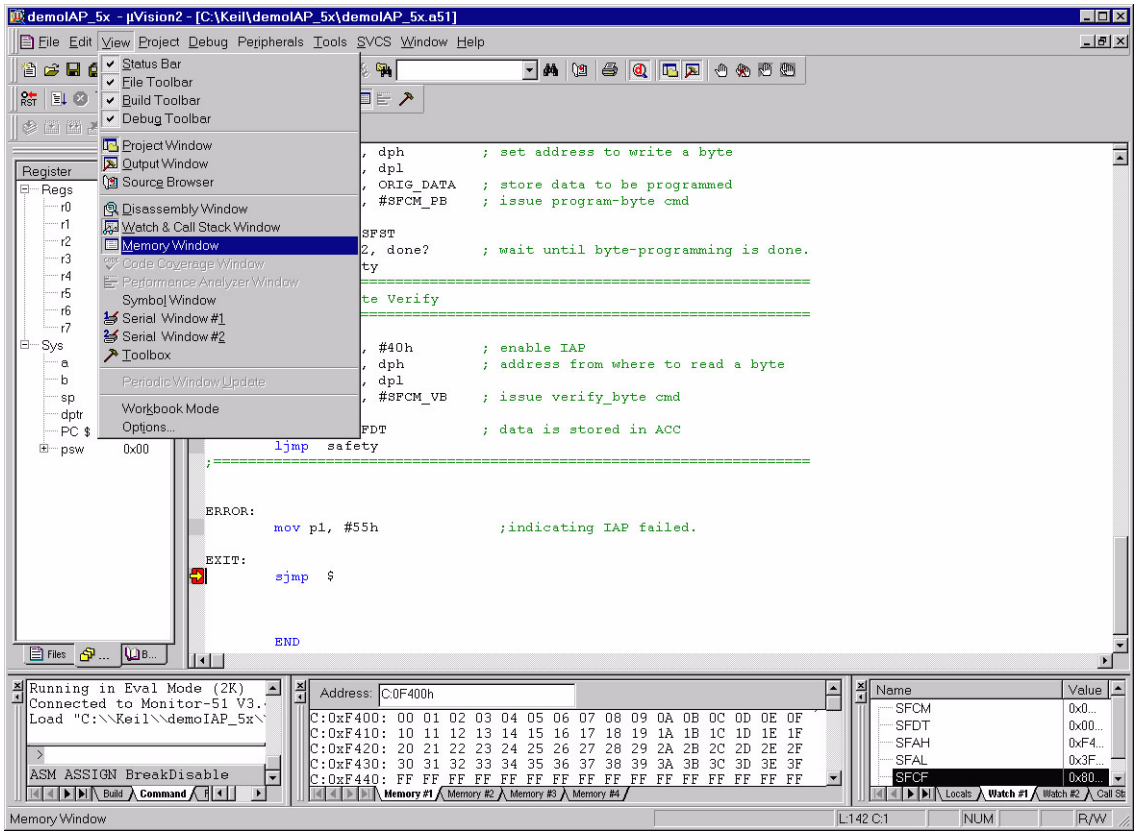
Note: The SST-specific SFRs should be defined properly in the user program.



Monitor the IAP procedure.

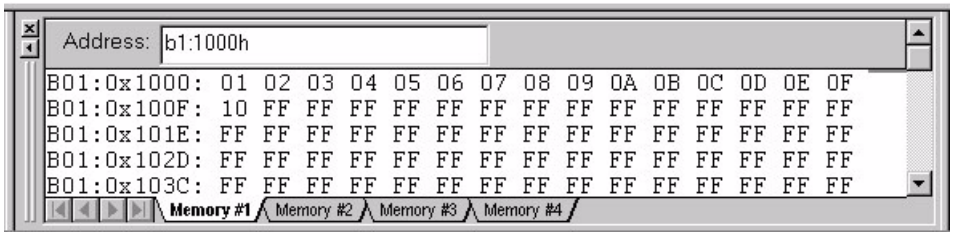
To view the effects of an IAP command procedure executing in block 0 on flash memory in block 1, open the Memory Window by clicking the Memory Window in View menu. Type the target address of IAP in the Memory Window. The result will be shown in the Memory Window immediately.

Note: Make sure Cache Options in the Target Setup window (See page 31) are not checked for real time display.



For SST89C5x SoftICE, the flash memory available for IAP is starting from C:0F400h.

For SST89x564RD SoftICE, C: (or B0:) is used to display code memory in block 0. B1: is used to display code memory in block 1. And the code memory available for IAP is starting from B1: 1000h.



For SST89E554 / SST89V554RC SoftICE, both C: and B0/B1: can be used to display code memory. Either B0: or C:0000h – C:7FFFh can be used to display code memory in block 0. Either B0: or C:0E000h – C:0FFFFh can be used to display code memory in block 1. The code memory available for IAP is starting from b1:1000h or C:0F000h.

Note: If there is problem writing "00h" or "0FFh" to the Memory Window, please try the following technique:

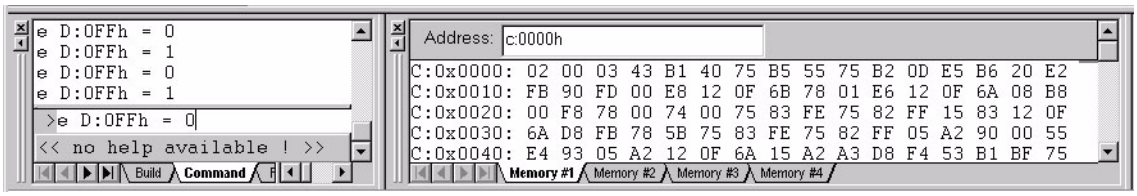
- **To Read/Write to code memory in block 1:**

Type in e D:0FFh = 1 in Output window (select Output window in View menu). Type in the starting address of the code memory to be displayed in the Memory Window. Any modifications to block 1 can now be made through the Memory Window.

- **To Read/Write to code memory in block 0:**

Type in e D:0FFh = 0 in Output window. Type in the starting address of the code memory to be displayed in the Memory Window. Any modifications to block 0 can now be made through the Memory Window.

The default setting is select code memory in block 0.



IAP DEMO PROGRAM

There are two IAP Demo programs to demonstrate Sector-Erase, Byte-Program, and Byte-Verify programming routines using SST-specific IAP. The programs are clearly-structured and well-documented for user understanding.

- DemoIAP_5X, is for SST89C5x devices. (See Reference 2)
- DemoIAP_564 is for SST89E564RD / SST89V564RD devices. (See Reference 2)
- DemoIAP_554 is for SST89E554RC / SST89V554RC devices. (See Reference 2)

These sample programs are for user reference only. SST does not guarantee the functionality or the usefulness of this sample code.

SOFTICE USER CODE RESTRICTIONS

- KEIL MON51.DLL does not support the following KEIL debugger features:

  Memory Map/Performance analyzer/Call stack analyzer/Code coverage.

  Trace is not supported by SoftICE because of its tremendous memory usage.
- For SST89C5x, the VIS bit (SFCF.7) should always be 1. For SST89x564RD/554RC, neither SC0 or SC1 should be programmed by user program.
- SST-specific WatchDog Timer and Soft Reset are not supported.
- User code should be within 0000h-6FFFh for SST89C58, 0000h-2FFFh for SST89C54, 0000h-FBFFh in block 0 for SST89E/V564RD, and 0000h-7BFFh in block 0 for SST89x554RC.
- No breakpoint should be set on a one-byte instruction, if a label (jump target) is right after this instruction (within two bytes).
- Timer 2 can be used as a timer/counter only during a GO command. TR2 (T2CON.2, start timer 2) can only be set to 1 after the point where GO is issued, and should be cleared before the GO command reaches a break point.
- When selecting the "Stop Program Execution with Serial Interrupt" option, SoftICE will modify three bytes beginning at the serial port interrupt vector location 0023h. Make sure that the user program does not use these code locations. The continuous hand shaking signals between PC and MCU will make the execution of the user program slower during a GO. And Timer 2 can not be used as a timer/counter even during a GO command. This function should not be selected if a real-time GO is desired.
- User cannot Step Into an interrupt service routine (ISR) from the main program. When Stepping ISR is needed, user can set a break point at the beginning of the ISR, then Step through the remainder of the ISR code after the reaching the break point.

APPENDIX F. RESOURCES USED BY BSL V1.1F

The user is required to save the contents of resources listed in this appendix before branching to BSL code, and then restoring their original values upon returning from the BSL code.

SST89C54/C58

BSL v1.1F restores reset values into SFRs, except the SFCF=80h to set the visibility bit and remap=0 KB, before executing user's code starting at 0000h of block 0.

1. Internal Flash Memory

   The code stores in secondary block of flash memory occupies the lower 16 sectors (64 bytes per sector) or address range from F000h—F3FFh.

2. Internal RAM

   Register R0—R7 in register bank 0 and internal RAM 08—0Dh.

3. Special Function Registers (SFR)

   There are 34 SFRs in FlashFlex51 SFR Memory Map of SST89C54/C58 MCU. The BSL v1.1F uses the following 27 SFRs: SP, DPL, DPH, WDTD, TMOD, TL0, TL1, TH0, TH1, SCON, SBUF, IE, SFCF, SFCM, SFAL, SFAH, SFDT, SFST, WDTC, T2CON, RCAP2L, RCAP2H, TL2, TH2, PSW, ACC, and B.

SST89E/V564RD or SST89E/V554RC

BSL v1.1F restores reset values into SFRs, except the SFCF=02h to set SWR=1, before executing user's code starting at 0000h of block 0.

1. Internal Flash Memory

   The code stores in secondary block of flash memory occupies the lower 9 sectors (128 bytes per sector) or address range from 0000h—047Fh.

2. Internal RAM

   Register R0—R7 in register bank 0 and internal RAM 08—0Fh.

3. Special Function Registers (SFR)

   There are 65 SFRs in FlashFlex51 SFR Memory Map of SST89E/V564RD or SST89E/V554RC MCU. BSL v1.1F uses the following 24 SFRs: SP, DPL, DPH, WDTD, AUXR, SCON, SBUF, IE, SFCF, SFCM, SFAL, SFAH, SFDT, SFST, WDTC, T2MOD, T2CON, RCAP2L, RCAP2H, TL2, TH2, PSW, ACC, and B.

APPENDIX G. ENTRY INTO BSL V1.1F

Figure G-1, "MCU Firmware Architecture of the Boot-Strap Loader v1.1F" shows the entry stage from the user's code or follows a power-on reset operation on the MCU target board.

SST89C54/C58

Entry into Internal Memory Mode is possible only if the following MCU requirements are met: (1) BSL code must reside in block 1 of MCU flash memory, (2) MCU must be either unlocked or soft-locked both block 0 and 1, and (3) MCU must be set to 1K/2K/4K re-map. Once the chip requirements are met, BSL routines can be entered by any of the following ways:

1. Power-cycle the system, which provides a power-on reset.
2. With power on the system, apply an external reset signal to MCU.
3. Issue a branch instruction from user code, for example:

```
ORL SFCF, #80h          ; make block 1 flash visible
LJMP 0F000h             ; enter BSL code
```

SST89E/V554RC or SST89E/V564RD

Entry into Internal Memory Mode is possible only if the following MCU requirements are met: (1) BSL code must reside in block 1 of MCU flash memory, and (2) Start-up Configuration bits SC0/SC1 must be in an un-programmed (erased) state. From the factory, SC0/SC1 bits are in an erased state. Once the chip requirements are met, BSL routines can be entered by any of the following ways:

1. Power-cycle the system, which provides a power-on reset.
2. With power-on the system, apply an external reset signal to MCU.
3. Issue a bank switch instruction from the address higher or equal to 2000h in block 0, for example:

```
ORG 2000H
ANL SFCF, #00h          ; bank switch: switch to block 1
LJMP 0000H              ; enter BSL code
```

How to by-pass BSL1.1F

To Remove BSL v1.1F Code: Execute a Chip-Erase to erase all BSL1.1F code and any user code or data on the chip. Re-program the user code into Block 0 and program SC0 in order to start from the user code after each MCU reset.

To Retain BSL v1.1F Code: To keep BSL1.1F in Block 1 unchanged, program SC0 to directly start user code after each MCU reset. User is still able to enter into BSL1.1F by means of item 3 in the previous section.

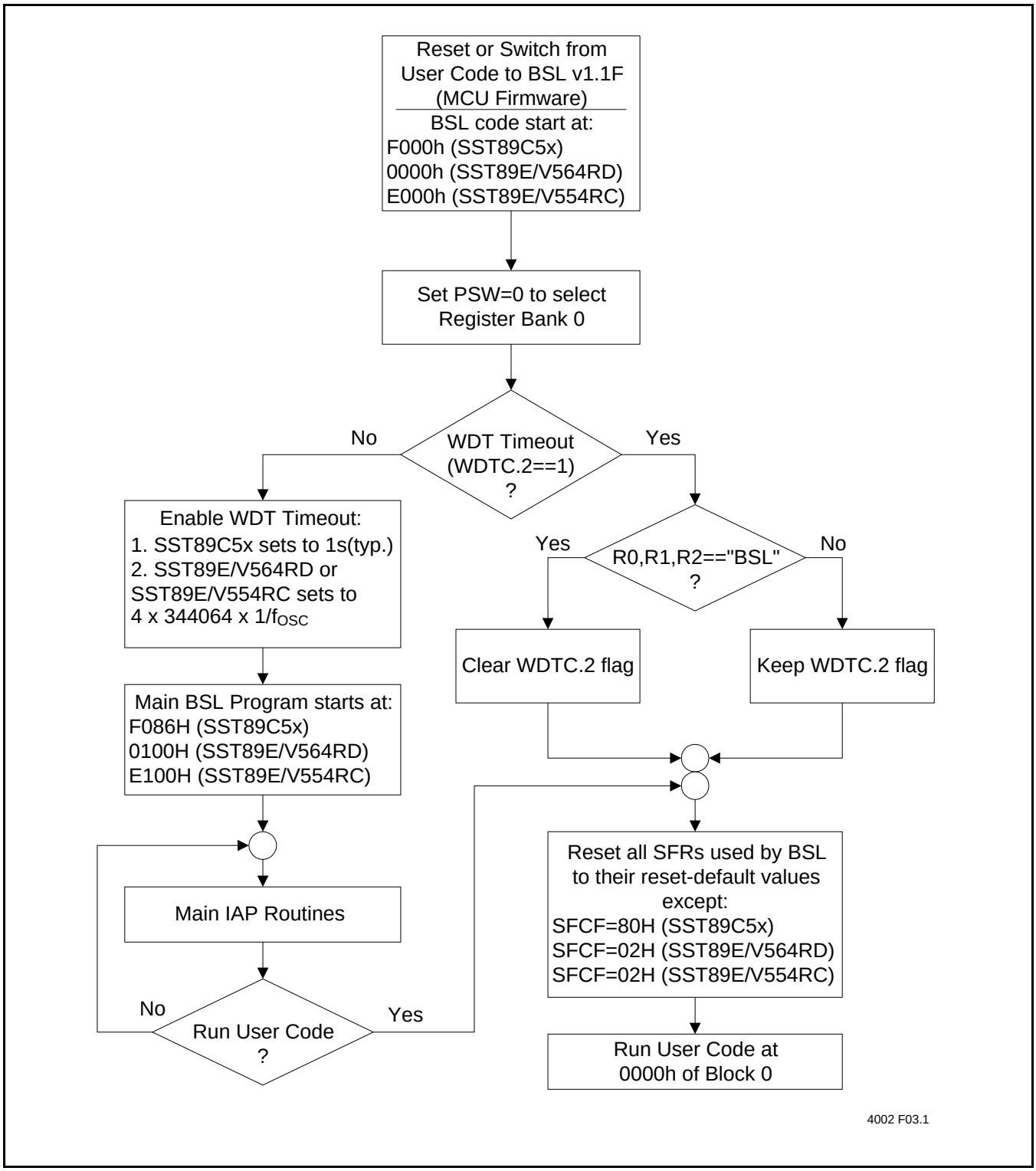


FIGURE G-1: MCU FIRMWARE ARCHITECTURE OF THE BOOT-STRAP LOADER V1.1F