Silicon Storage Technology, Inc.

EasyIAP™ Software Example

User's Guide

© 2001 Silicon Storage Technology, Inc. S71202-NR-001 7/01 414

The SST logo and SuperFlash are registered trademarks of Silicon Storage Technology, Inc. FlashFlex, In-Application Programming, IAP, EasyIAP and SoftLock are trademarks of Silicon Storage Technology, Inc. These specifications are subject to change without notice.

TABLE OF CONTENTS

Introduction

EasyIAP is a Boot-Strap Loader (BSL) software example[1] that 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. EasyIAP is also capable of self-detection of serial connection and automatically reset the chip after file download.

The companion SST89C54/58 MCU datasheet/specification and the MCU source code should be reviewed in conjunction with this document for a complete understanding of the software application. Datasheet and source code can be downloaded from the SST web site (www.ssti.com).

EasyIAP MCU Firmware Description

EasyIAP v1.1E shares the same source code as the Internal Memory Mode of SST Boot-Strap Loader v1.1E. The source code of MCU firmware is being supplied to SST customers free of charge as a reference example. Their designers can choose to use the source code in its original form or modify it extensively to suit their own FlashFlex51 application. The source code is intentionally well-documented to hopefully guide these designers in understanding the software.

The flowchart in Figure 1 shows how the host PC interacts with the EasyIAP firmware stored in SST89C5X chips. Figure 2 provides a flowchart with the high-level structure of the EasyIAP firmware.

Entry into the EasyIAP firmware is possible only if the following MCU requirements are met: (1) EasyIAP firmware must be resident in the MCU Block 1 flash memory, (2) the MCU must not be hard locked, and (3) the MCU must default to a 1KB re-map on reset. Once the chip requirements are met, the EasyIAP firmware can be entered by any one of the following ways:

(a) Power cycle the system, which provides a power-on reset.
(b) With power on the system, push the reset button (hardware reset).
(c) Watchdog (WDT) timeout in user application code.
(d) Issue a branch instruction from within user code, for example:

```
MOV   SFCF, #80h    ; make block 1 flash visible
LJMP  0F000h        ; enter EasyIAP
```

Descriptions of MCU resource utilization and memory re-mapping associated with EasyIAP are now presented.

[1] The EasyIAP 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 boot-strap loader.

MCU Resources Used by EasyIAP

The MCU resources used by EasyIAP are:

1. Registers A, B, DPTR and SFCF.
2. Register R0-R6 in register bank 0.
3. Internal RAM 08h-0Fh.
4. Watchdog Timer (WDT).

If EasyIAP routines are called from user code, the user is required to save the contents of the registers and internal RAM, listed as items 1, 2 and 3 above, before branching to EasyIAP routines, and then restoring their original values upon returning from the EasyIAP routines. Additionally, the EasyIAP writes the string "**USER**" into 08h-0Bh of internal RAM to prevent the Watchdog Timer Reset Flag (WDTS) from being cleared before returning to user code. Furthermore, the stack requires use of the internal RAM from 0Ch to 0Fh after the serial link is established and EasyIAP is waiting to execute an IAP command.

If EasyIAP is entered after a power-on-reset, EasyIAP writes the string "**POWR**" into 08h-0Bh of internal RAM to allow the Watchdog Timer Reset Flag (WDTS) to be cleared before returning to user code.

Memory Re-mapping

The size of memory re-mapping selected by the user determines the operational state of the Watchdog Timer (WDT) that is put in place in transitioning from EasyIAP to user code.

Re-mapped to 1 KB – Program access to location 0000h-03FFh is redirected to F000h-F3FFh. The WDT is disabled before executing user code resident in block 0. 1KB re-map is the default setting from the factory.

Re-mapped to 2 KB – Program access to location 0000h-07FFh is redirected to F000h-F7FFh. The WDT is disabled before executing user code resident in block 0.

Re-mapped to 4 KB – Program access to location 0000h-0FFFh is redirected to F000h-FFFFh. The WDT is enabled before executing user code resident in block 0. The user needs to refresh the WDT constantly in user code before the WDT expires.

For the 4KB re-map case, user code can execute for about 2 seconds typically before the WDT expires when programmed for maximum timeout.

During factory test, the SST89C54/58 chips are pre-programmed with EasyIAP into block 1 flash memory, and are re-mapped to the 1 KB range.

There are three appendices at the end of this User's Guide. Appendix A provides the file name convention for the EasyIAP firmware, Appendix B provides a list of available EasyIAP source code and Appendix C lists the pseudo IAP command sequences to be executed by the MCU firmware.

EasyIAP Windows Software Description

The flowchart in Figure 1 provides an operational overview of the SST EasyIAP software. For **Chip erase**, **Memory re-mapping** and **Set security-level**, the user needs to use either the SST BSL Demo Kit or a universal programmer to perform these MCU operations. The Windows software is capable of self-detection of serial link or start/stop/re-start the Easy IAP without touching any hardware. The descriptions of these features and the user interface are provided next.

Self-detection of Serial Link

EasyIAP software can detect whether the serial link is alive or not in two to six 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.

Start/Stop/Re-start EasyIAP w/o Touching Any Hardware

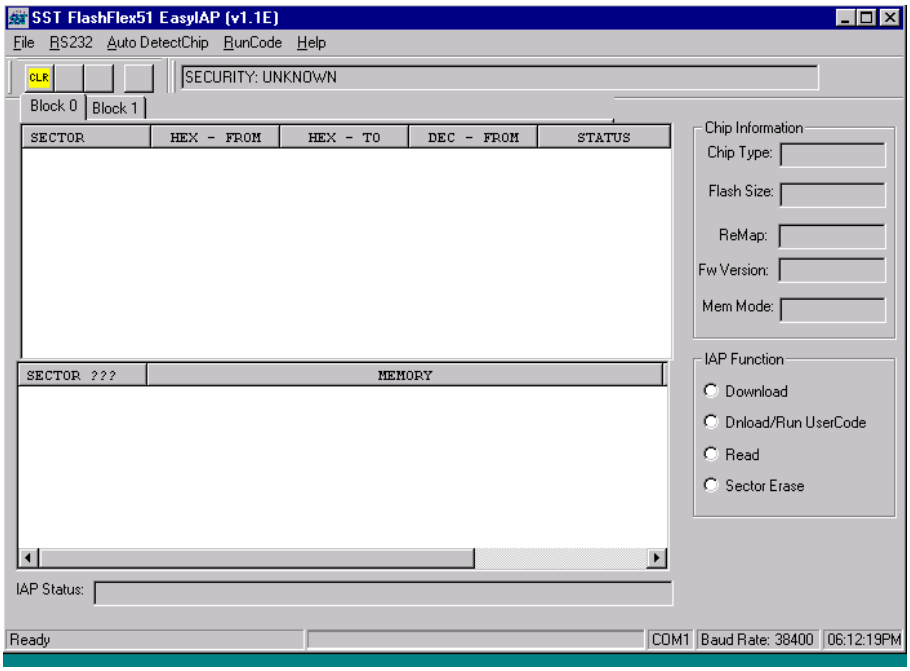
If the chip is 4 KB re-mapped, EasyIAP turns on the WDT before returning to user code. This arrangement promises the MCU to run the user code for 2 seconds typically if EasyIAP routine is blocked by an interruption of serial link. Then the MCU issues a query command to the PC and the PC can automatically re-establish the serial link after the cause of interruption is fixed.

After returning from EasyIAP routines to the user code, the user needs to constantly refresh the WDT in user code to prevent the WDT expiration during normal operation of user code.

User Interface

A. Entry Menu

In the Entry Menu, all system parameters, such as COM port and baud rate, are automatically initialized to their default values. Chip type is set to the chip detected.



B. RS-232 Configuration

The user has to click on RS232 menu button to select the COM port and baud rate before clicking on the Auto DetectChip button. The baud rate selected for COM port must match with the baud rate set in MCU firmware.

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

C. Auto DetectChip

The user is advised to select Auto DetectChip, click on OK and then allow the program to automatically detect the chip type and firmware version, which was pre-programmed in block 1 flash memory, in two seconds. Reset **MCU** to detect the chip type ONLY if the auto-detection failed the two-second window.

D. Compare

To compare an Intel hex or a binary file with the contents in flash memory (both blocks 0 and 1), 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 succeed!" is for a matched comparison or the text of "Unmatched data at memory address xxxxh: xxh(MCU) vs. xxh(File)" is for unmatched comparison.

E. Download

User code can only be downloaded into block 0 flash memory. 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 (0000h), then click on OK.

Prior to downloading, the sectors in block 0 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 the file, the user can click on the Browse button located at the right end of File Name list box.

F. Download/Run User Code

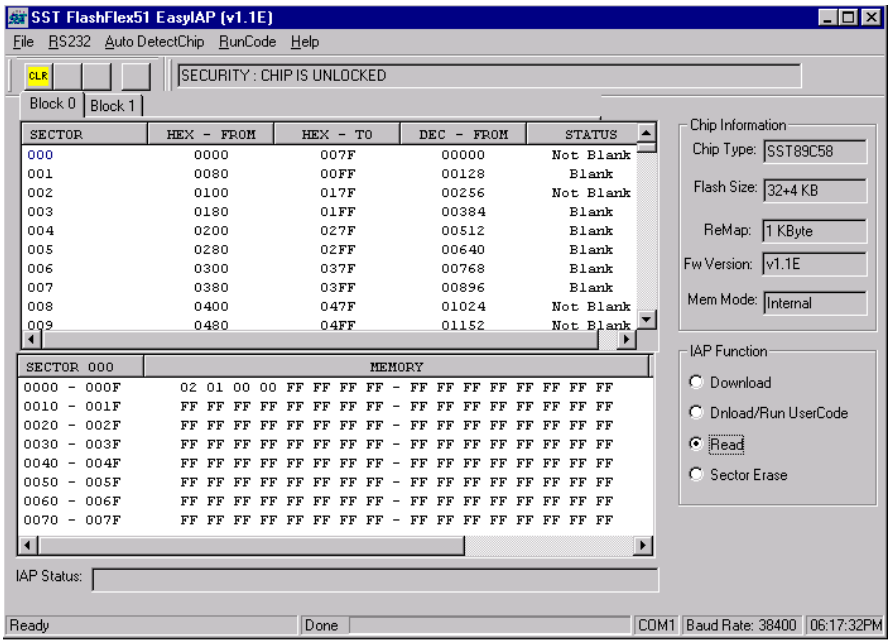
This function combines **Download** and **Run User Code** into one. The Download/Run-user-code command automatically runs user code after reprogramming block 0 flash memory.

G. Print Memory Contents

The Print option from File allows the user to send memory data from flash block 0 or 1 to a printer.

H. Read

This function reads the code from either block 0 or block 1 flash memory, then displays the contents in hex 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 three conditions – *Blank, Not Blank or Not Available*. The content of an unread sector which shows all 00s corresponds to the *Not Available* status.



I. Clear Chip Info

Clicking on the yellow CLR radio button erases the Chip Information on the screen only; it cannot reset the MCU chip.

J. 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 flash memory.

K. Save Data into a File (Upload)

To save the flash contents of block 0 or 1 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.

L. Sector Erase

This option enables the user to select the region of block 0 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.

Installing EasyIAP

The EasyIAP package includes a PC executable program and the MCU source code. The PC executable, SSTEasyIAP.EXE, is a Window-based application and runs directly under Window 95/98/Me 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 SSTEasyIAP.EXE only if they don't exist or are not the latest revision codes.

Working with either 11.0592 MHz or 12 MHz version of the MCU source code, the PC pre-settings for serial communication are: 38.4K/19.2K/9600/4800/2400 baud, 8 data bits, 1 stop bit and no parity.

To facilitate ease of translation, the user source files can be located in the same user folder containing the 8051-MCU software development environment. The PC executable SSTEasyIAP.EXE shall be installed on a user's PC to be ultimately run and linked to the MCU code residing in the SST microcontroller on the development platform. The MCU code, residing in MCU block 1 flash, can be installed in four different ways:

1. by the factory,
2. by the user with SST Boot-Strap Loader(BSL) Demo Kit,
3. by the user with Phytec Evaluation Kit, or
4. by the user with an universal programmer that supports the SST microcontroller being used.

Please visit the SST website for the information on the SST BSL Demo Kit, the Phytec Evaluation Kit and the list of programmer vendors that support the FlashFlex51 family.

Demo Software

Three demo software programs have been supplied with the EasyIAP, 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, the Phytec Evaluation kit 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

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 right 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 left in the same manner. The right-left sequence will be continuous.

BINCTR

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

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.

Serial Cable

The serial cable to be used between the Host-PC and development platform shall be the standard RS-232 DTE-DCE cable, with a female connector on the end that attaches to a PC (Data Terminal Equipment, DTE) and male connector on the end attaching to the development platform (Data Communication Equipment, DCE). The serial cable connections are:

PC DB-9 plug (COM1 or 2)		Development Platform
RxD Pin 2	to	TxD Pin 2
TxD Pin 3	to	RxD Pin 3
GND Pin 5	to	GND Pin 5

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.

For technical support with the EasyIAP via email, please contact the SST FlashFlex51 product Hot-Line: flashflex51@ssti.com. In the future, check the SST website (www.ssti.com) under FlashFlex51 Microcontrollers for information on the available downloadable versions of the EasyIAP code.

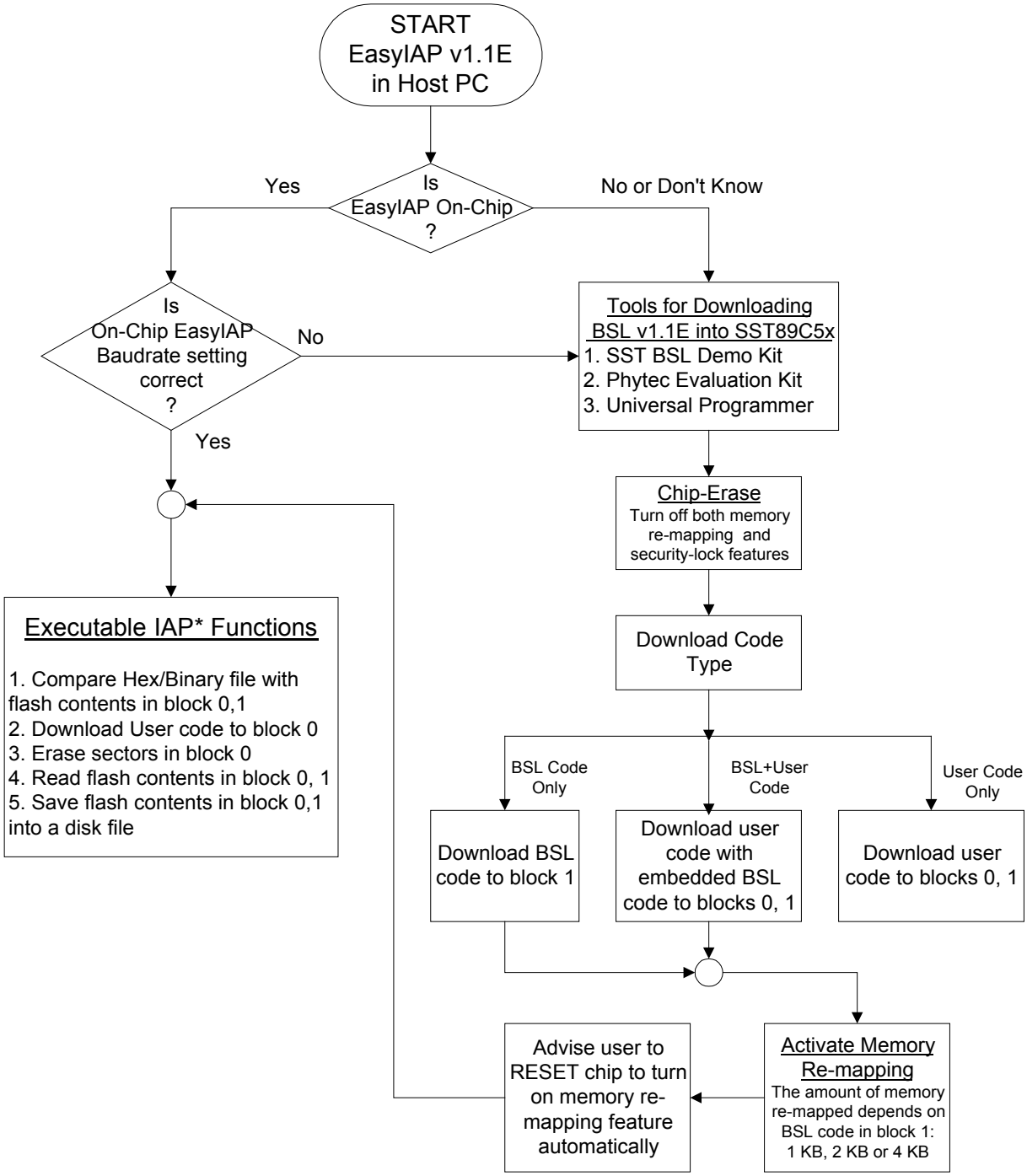


FIG.1 SST EasyIAP* v1.1E and Usage of Memory Re-mapping in SST89C5x MCU

*EasyIAP and In-Application Programming(IAP) are trademarks of Silicon Storage Technology, Inc.

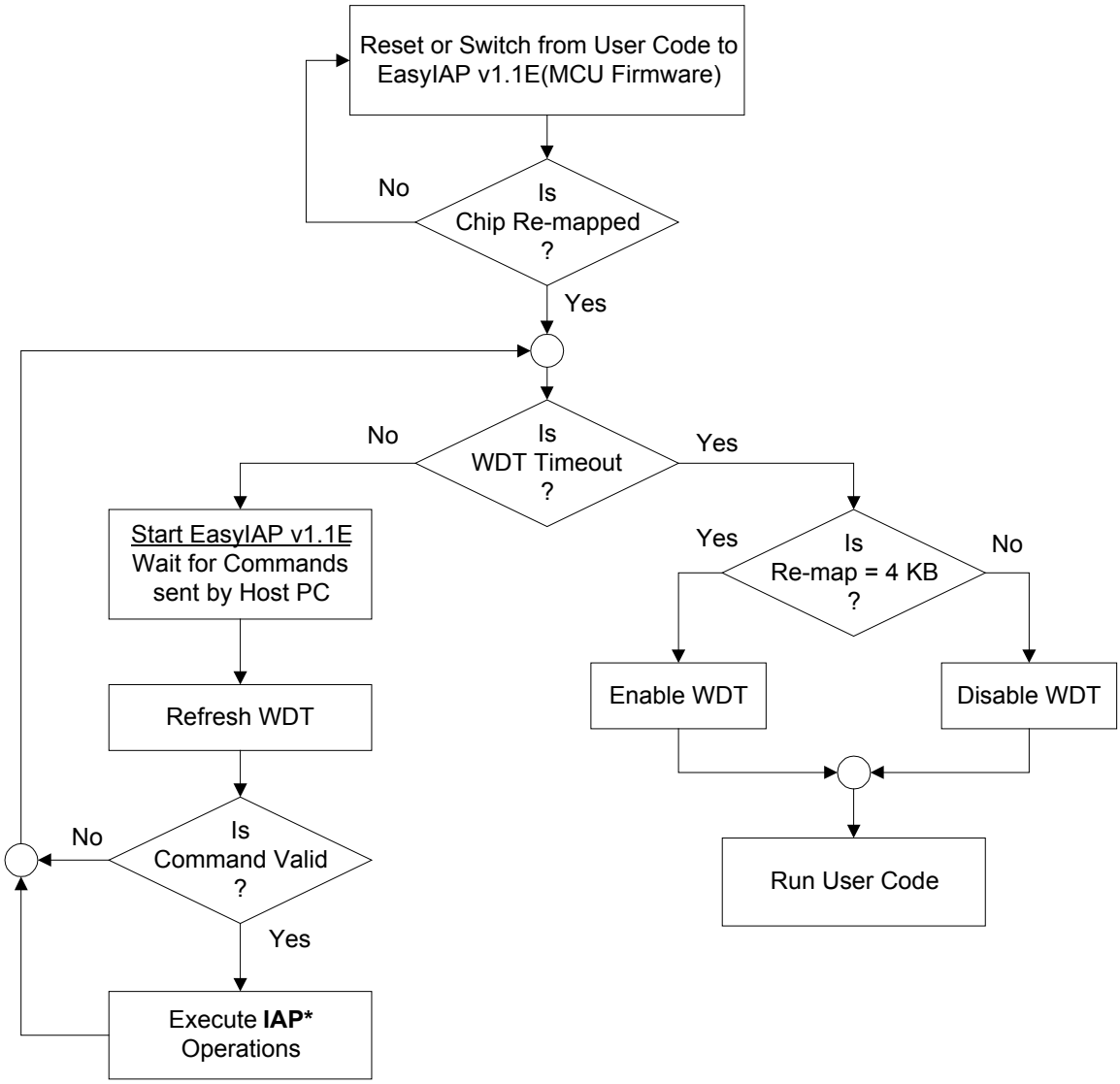


FIG.2 MCU Firmware Architecture of EasyIAP* v1.1E

*EasyIAP and In-Application Programming(IAP) are trademarks of Silicon Storage Technology, Inc.

APPENDIX A. File Name Convention

The EasyIAP 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 = P for MS-DOS, PC-resident code
M for MCU-resident code in Internal Memory Mode
C for Custom MCU-resident code

y = BSL/EasyIAP Version Number for x = P
A = Version 1.0
B = Version 1.1/ 2.0
C = Version 1.2

y = Chip Type and Frequency for x = M
E = C58 11.0592 MHz
F = C58 12 MHz
G = C54 11.0592 MHz
H = C54 12 MHz
(etc.)

y = Customer ID for x = C
A = Infronex C58 7.3728 MHz
B = (next customer)

z = Revision No.
0 = Original Release
1 = First Revision
(etc.)

APPENDIX B. List of Source Code

The EasyIAP package now consists of two parts: for the Window 95/98/Me-resident software, an executable file (SSTEasyIAP.EXE) is supplied, and for the MCU-resident code, the source files (.A51), an Intel hex file (.HEX) and a binary file (.BIN) are furnished. Table B1 lists the files that can be downloaded from the SST web site. Both 11.0592 MHz and 12.0 MHz versions use MCU Timer 2 for baud rate generation for the serial port.

Table B1. List of EasyIAP v1.1E Files

Chip Type	Ext. Crystal Freq.	Baud Rate	PC Files	MCU Files
89C58 MCU	11.0592 MHz	38.4K/19.2K/9.6 K/4.8K/2.4K	SSTEasyIAP.exe	F51MBLE3.A51 F51MBLE3.HEX* F51MBLE3.BIN*
	12.0 MHz	38.4K/19.2K/9.6 K/4.8K/2.4K	SSTEasyIAP.exe	F51MBLF3.A51 F51MBLF3.HEX* F51MBLF3.BIN*
89C54 MCU	11.0592 MHz	38.4K/19.2K/9.6 K/4.8K/2.4K	SSTEasyIAP.exe	F51MBLG3.A51 F51MBLG3.HEX* F51MBLG3.BIN*
	12.0 MHz	38.4K/19.2K/9.6 K/4.8K/2.4K	SSTEasyIAP.exe	F51MBLH3.A51 F51MBLH3.HEX* F51MBLH3.BIN*

* Binary file should be downloaded into block 1 and starts at address F000h. Hex file needs to be downloaded into block 0 and starts at address 0000h.

APPENDIX C. PSEUDO COMMAND SEQUENCES

Table C1 lists the pseudo-command sequences sent from the host PC to the SST89C5x MCU. The code in MCU chip decodes the pseudo-command sequence and executes IAP command accordingly.

Table C1. Pseudo IAP Command Sequence (all values are in Hex format)

HOST PC ←→ SST89C5x MCU	Pseudo CMD Sequence	IAP CMD (MCU)	Description of Pseudo CMD Sequence
Handshaking	05 55	None	Establish the serial link
Burst-Program	06 XX YY ZZ	06	XX/YY/ZZ: Addr-Hi/Addr-Lo/No. of bytes in a ROW(half of a sector)
Sector-Erase	OB XX YY ZZ	0B	XX/YY/ZZ: Addr-Hi/Addr-Lo/Sector-Count
Byte-Verify	0C XX YY ZZ	0C	XX/YY/ZZ: Addr-Hi/Addr-Lo/No. of Sector Bytes
Byte-Program	0E XX YY ZZ	0E	XX/YY/ZZ: Addr-Hi/Addr-Lo/No. of Sector Bytes
Device code & FW version	60	None	User needs to pre-program Device code & FW version in MCU source code
Run-UserCode	62 62	None	Reset the chip and run user code at 0000h
Query Byte	F7	None	MCU is ready to receive CMD from host PC