SST89F5x to SST89C5x Code Conversion Guide

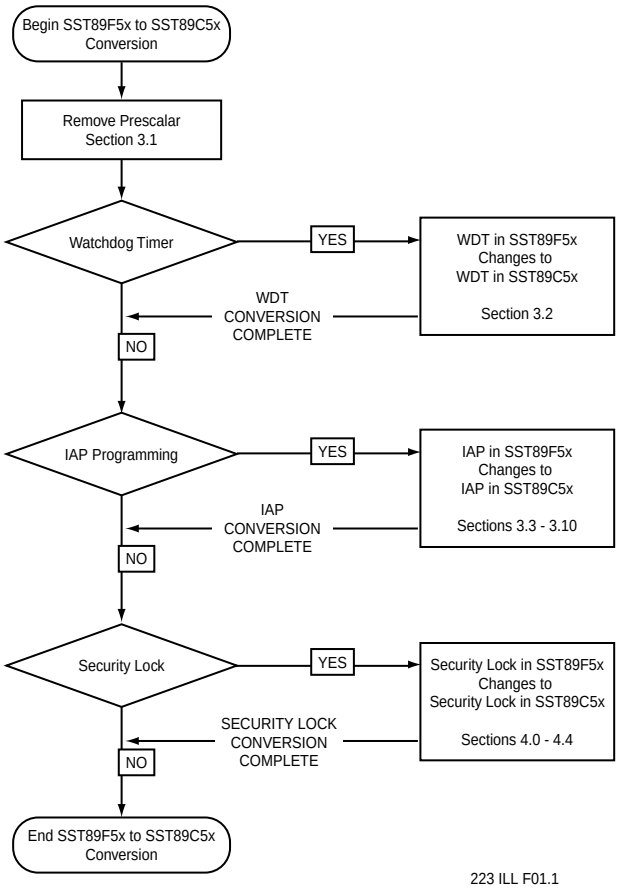
Application Note
March 2000

1.0 Introduction

This application note provides the guidelines for converting existing firmware code of the SST89F5x to run on the SST89C5x. This note is applicable to all applications that use SST89Fx microcontrollers, but mostly pertains to the IAP and Security Lock features. New features that are unique to SST89C5x, and do not pertain to the SST89F5x, are not mentioned in this document. External Host Mode operation is not covered here.

2.0 Code Conversion Flowchart

The following flowchart outlines the code conversion process:



223 ILL F01.1

2.1 Firmware Conversion Process

A summary of the changes required to convert code originally implemented for the SST89F5x to the SST89C5x are as follows:

Prescalar

1. Remove the Frequency (FREQ) Prescalar bits in SFCF. (See 3.1)

Watchdog Timer

2. Recalculate desired timeout. (See 3.2)

IAP Programming

3. Change Flash SFR addresses. (See 3.3)
4. Eliminate IAP-Complete command from all IAP code. (See 3.4)
5. Split SFCF in SST89F5x into SFCF and SFST in SST89C5x. (See 3.5)
6. Set the IAP Enable (IAPEN) bit located in SFCF before IAP code execution. (See 3.6)
7. For Flash operation completion “polling”, use SFST[2] instead of SFCF[3] for SST89C5x. (See 3.7)
8. Change command numbers for the following IAP commands:
   A. Chip-Erase: 07h to 01h (See 3.8)
   B. Block-Erase: 0Fh to 0Dh (See 3.9)

Security Lock

9. Remove setting Security Lock via Block 1 address 0FFFFh in SST89F5x and use the three IAP Security Lock bit commands PROG-SB1 to SB3 for SST89C5x instead. (See 4.1-4.4)
10. Change Security Lock Status monitoring from SFCF in SST89F5x to SFST in SST89C5x. (See 4.1)

3.0 Code Conversion Details

The details of the Prescalar, Watchdog Timer and IAP Programming code conversion, are presented in the following sections.

3.1 Prescalar

SST89F5x

For SST89F5x, clock frequency must be configured using three bits in the SuperFlash Configuration/Status Register (SFCF[2:0]) . The critical timing for all Erase and Program commands is dependent upon the minimum frequency pre-scaling factor specified in the SuperFlash Configuration/Status Register (SFCF[2:0]).

For example, to implement an 11.059MHz oscillator to drive the SST89F5x chip, the bits SFCF[2:0] must be set to 001. The reason is that 001 represents the range of clock frequency from 8 to 16MHz. The operational frequency pre-scaling factor FREQ in (SFCF[2:0]) is set by implementing the following assembly command:

```
MOV SFCF, #01h    ;initializing FREQ bits in the
                  ;SFCF register or load #81h
                  ;into SFCF if Block 1 also
                  ;needs to be visible
```

SST89C5x

For SST89C5x, flash operations for the microcontroller use an internally generated clock. Thus there is no need for a Prescalar, and the firmware associated with setting the Prescalar FREQ bits is to be removed for SST89C5x implementation.

3.2 Watchdog Timer

SST89F5x

The watchdog timer for the SST89F5x device contains a wider range of timer settings than for the SST89C5x because the time tick is based on a crystal-controlled oscillator and prescalar frequency setting established by the software. Since the time is based on a crystal-controlled oscillator, the time tick is very predictable.

SST89C5x

In the SST89C5x device, the time for the watchdog timer is based on an internal 10 MHz oscillator that is affected by process, temperature and voltage variations. Thus, its time tick accuracy is less predictable. The minimum time tick is 7.7 milliseconds and the typical time tick is 10 milliseconds, with the maximum number of time ticks equal to 255. The user needs to recalculate the load value setting for their watchdog timer based on this less accurate time tick.

3.3 SFR Address Changes

SST89F5x and SST89C5x

The SST89F5x microcontroller devices have different Flash Memory Programming SFR addresses compared to the SST89C5x microcontroller. Table 3.3 shows the translations in addresses of the Flash Memory Programming SFRs from the SST89F5x to the SST89C5x microcontrollers. Note that the new SFST register is added with an address of B6h. These changes should be made in the equates (EQU) of the firmware code that uses these Flash Memory Programming SFRs.

TABLE 3.3: FLASH MEMORY PROGRAMMING SFRS

Symbol	Description	SFR Address for SST89F54/58	SFR Address for SST89C54/58
SFAH	SuperFlash Address High	FAh	B4h
SFAL	SuperFlash Address Low	F9h	B3h
SFCF	SuperFlash Configuration	F7h	B1h
SFCM	SuperFlash Command	FBh	B2h
SFDT	SuperFlash Data	F8h	B5h
SFST	SuperFlash Status	Not Supported	B6h

Application Note

3.4 IAP-Complete Command

SST89F5x

The IAP-Complete command is used in SST89F5x to exit the IAP mode and return the device to normal code execution without using the Chip-Erase command. The assembly command for IAP-Complete is

```
MOV SFCM, #00h    ;move IAP-Complete command
                  ;into SFCM register
```

SST89C5x

The IAP-Complete command code is to be removed from the firmware to support SST89C5x.

3.5 SFCF Integration

SST89F5x

For the SST89F5x microcontroller, the configuration and status bits for IAP operations resides in a single SFR register, SFCF. The bit content of the register is the following:

SuperFlash Configuration Register (SFCF)

Location	7	6	5	4	3	2	1	0
F7h	VIS	SECD		-	BUSY	FREQ		

SST89C5x

For the SST89C5x microcontroller, the configuration and status bits are separated into two registers – the configuration register SFCF and status register SFST. The bit content of the two registers is the following:

SuperFlash Configuration Register (SFCF)

Location	7	6	5	4	3	2	1	0
B1h	VIS	IAPEN	-	-	-	-	MAP_EN	

SuperFlash Configuration Register (SFST)

Location	7	6	5	4	3	2	1	0
B6h	SECD[2:0]			-	BUSY	Flash_Busy	-	-

3.6 IAP Enable Bit

SST89F5x

None.

SST89C5x

One of the differences between SST89F5x and SST89C5x is that the latter requires initialization of the IAPEN bit located in SFCF[6] of the Flash Memory Programming SFRs to enable IAP operations. Use the ORL instruction to modify the IAPEN bit in order to avoid conflicts with the VIS, MAP_EN1 and MAP_EN0 bits in SFCF for SST89C5x.

```
ORL SFCF, #40h    ;set IAPEN bit
```

Until the IAPEN bit is set all flash programming IAP commands will be ignored.

3.7 Polling

SST89F5x

For SST89F5x, a command that uses the polling method to signify the completion of a flash operation must check the BUSY bit (SFCF[3]). Sample code for BUSY bit polling is:

```
DONE:  MOV A, SFCF            ;get SFCF contents
       ANL A, #08h            ;mask for BUSY bit
       CJNE A, #00h, DONE     ;if BUSY, repeat loop
                              ;else continue
```

SST89C5x

For SST89C5x, the Flash_Busy bit (SFST[2]) is used for polling to detect the completion of a flash operation. When Flash_Busy de-asserts (logic 0), the device is ready for the next operation. The BUSY bit (SFST[3]) is now provided for Burst-Program operation. In between bytes within a burst sequence, the BUSY bit will become logic 0 to indicate that the next Burst-Program byte should be presented. Completion of the full burst cycle is indicated also by Flash_Busy bit (SFST[2]). Sample code for SST89C5x polling is:

```
DONE:  MOV A, SFST            ;get SFCF contents
       ANL A, #04h            ;mask for Flash_Busy bit
       CJNE A, #00h, DONE     ;repeat loop until flash
                              ;operation complete
```

3.8 Chip-Erase Command

SST89F5x

For SST89F5x the Chip-Erase command erases both memory blocks (16/32 KBytes and 4 KBytes). This command ignores the security lock status so the Security Byte will be erased. The assembly commands for Chip-Erase in SST89F5x are:

```
MOV SFDT,  #55h ;load setup data
MOV SFCM, #07h ;move Chip-Erase IAP command
                ;into SFCM register for polling or
                ;load #87h instead of #07h if
                ;interrupts are used
```

SST89C5x

In SST89C5x, the assembly commands for Chip-Erase changes to

```
MOV SFDT,  #55h   ;load setup data
MOV SFCM, #01h    ;move Chip-Erase IAP command
                  ;into SFCM register for polling or
                  ;load #81h instead of #01h if
                  ;interrupts are used
```

3.9 Block-Erase Command

SST89F5x

The Block-Erase command operates on one of the two memory blocks (16/32 KByte and 4 KByte). The selection of the memory block to be erased is determined by AH (SFAH) of the SuperFlash Address Register. If (SFAH) is a “0Xh”, the primary flash memory is selected (16/32 KByte). If (SFAH) is “FXh”, the secondary flash memory block is selected (4 KByte). The assembly commands for Block-Erase for SST89F5x are:

```
MOV SFAH, #0F0h ;move high byte address into
                ;SFAH register
MOV SFDT,  #55h ;load setup data
MOV SFCM, #0Fh  ;move Block-Erase IAP command
                ;into SFCM register for polling or
                ;load #8Fh instead of #0Fh if
                ;interrupts are used
```

SST89C5x

In SST89C5x, the assembly commands for Block-Erase change to:

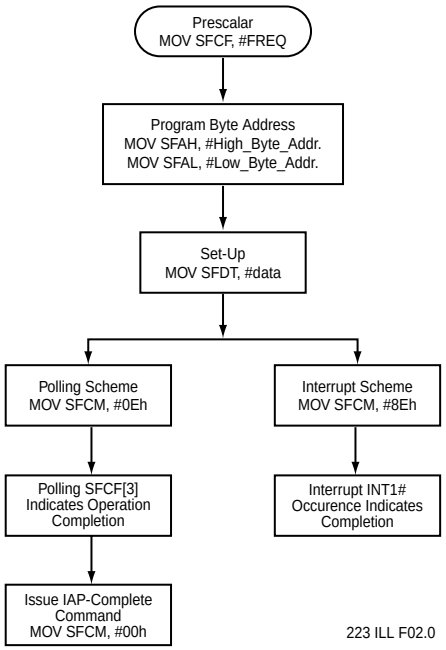
```
MOV SFAH, #0F0h ;select 4K block to erase
MOV SFDT,  #55h ;load set-up data
MOV SFCM, #0Dh  ;move Block-Erase IAP command
                ;into SFCM register for polling or
                ;load #8Dh instead of #0Dh if
                ;interrupts are used
```

3.10 IAP Byte-Program Example

SST89F5x

Flowchart for Byte-Program operation in IAP Mode for SST89F5x is shown in Figure 3.10A.

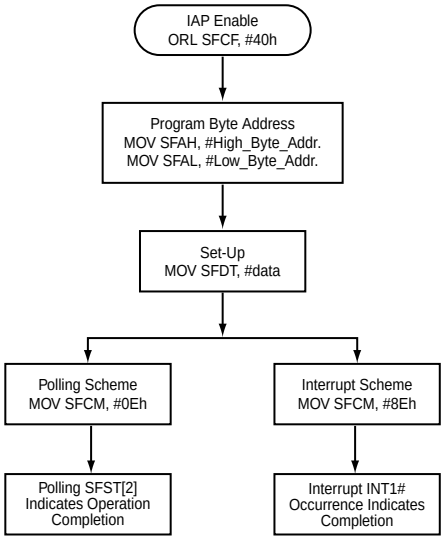
FIGURE 3.10A: BYTE-PROGRAM FOR SST89F5X MICROCONTROLLERS



SST89C5x

Flowcharts for Byte-Program operation in IAP Mode for SST89C5x is shown in Figure 3.10B.

FIGURE 3.10B: BYTE-PROGRAM FOR SST89C5X MICROCONTROLLERS



223 ILL F03.0

4.0 Security Lock

For SST89F5x microcontrollers, there are three security lock options. These are the SoftLock, Block1 Lock, and Hard Lock. SST89C5x microcontrollers provide four levels of security lock options. For SST89F5x microcontrollers, writing to FFFFh of program memory establishes the Security Lock option. For SST89C5x microcontrollers, programming the three security lock bits SB1, SB2 and SB3 establish the security lock. Resetting the SST89C5x device will indicate the security level and lock option in the status register. See below for details.

4.1 Security Lock Options

SST89F5x

The Security Lock options for the SST89F5x are enumerated in Table 4.1A. The Security Byte column indicates the data value to be placed into the FFFFh program memory address for the lock condition indicated.

TABLE 4.1A: SECURITY LOCK OPTIONS FOR SST89F5X

Security Byte	SFCF[6:5]	Description
FFh or 00h	00	No Lock
55h	11	Hard Lock
F5h	01	Block1 Lock
05h	10	Soft Lock

Once the Security Lock option is initiated, the SFCF[6:5] bits will indicate the Security Status upon Hardware Reset.

SST89C5x

Table 4.1B provides the Security Lock options for the SST89C5x. The options are divided into levels, and the security status is now shown in the SFST status register. Once a Security Lock option is established, its status will appear in the status register after the next Hardware Reset.

A description of the Security Levels for SST89C5x Microcontrollers is as follows. Level 1 of the Security Lock options disables all Security Features. Level 2 allows external programmers to read the contents of the MCU through IAP Byte-Verify command on External Host Mode. Level 3 blocks any access of the contents in the MCU but code residing in external memory can be ran. Here the IAP Byte-Verify command is disabled. In Level 4, EA# is disabled so any user will not be able to use the MCU to run code residing in external memory.

TABLE 4.1B: SECURITY LOCK OPTIONS FOR SST89C5X

Level	SFST[7:5]	Block 1	Block 0
1	000	Unlock	Unlock
2	100	Hard Lock	Hard Lock
3	110 101 010 001	Hard Lock Hard Lock Soft Lock Hard Lock	Hard Lock Hard Lock Soft Lock Soft Lock
4	111	Hard Lock	Hard Lock

SFST[7:5] status bit definitions:

000 – No Security Features are enabled.

100 – MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory, EA# is sampled and latched on Reset, and further programming of the flash is disabled.

110 or 101– Level 2 plus Verify disabled. Both blocks locked.

010 – Level 2 plus Verify disabled. Code in Block 1 can program Block 0 and vice versa.

001 – Level 2 plus Verify disabled. Code in Block 1 can program Block 0.

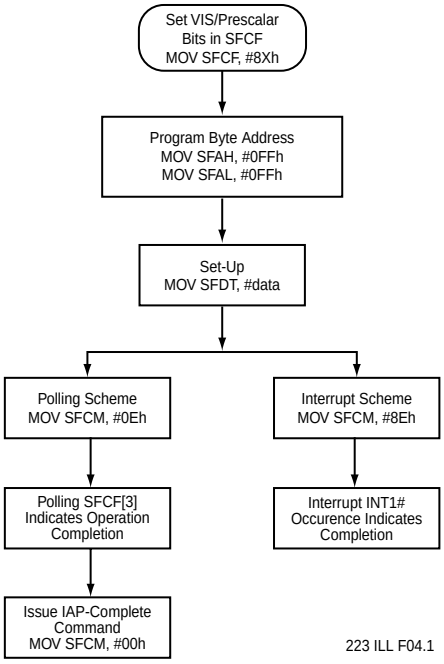
111 – Same as Level 3. External boot is disabled.

4.2 Security Lock Initiation

SST89F5x

The flowchart to initiate the Security Lock options for the SST89F5x is shown in Figure 4.2A.

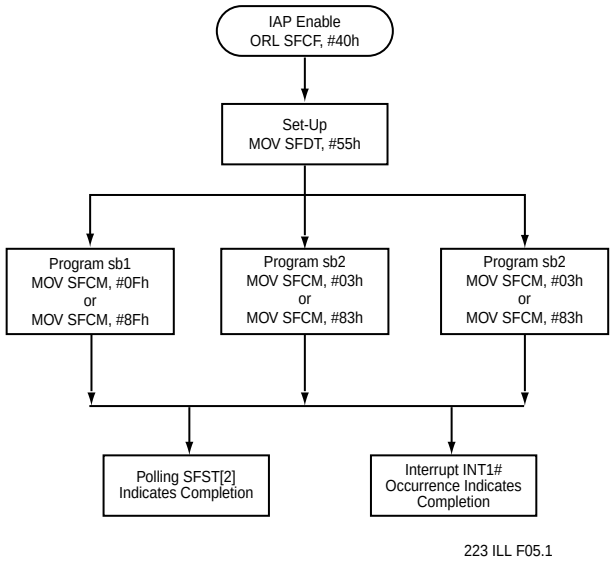
FIGURE 4.2A: SECURITY LOCK FOR SST89F5X MICROCONTROLLERS



SST89C5x

The flowchart to initiate the Security Lock options for the SST89C5x is shown in Figure 4.2B.

FIGURE 4.2B: SECURITY LOCK FOR SST89C5X MICROCONTROLLERS



4.3 Security Lock Conversion

Table 4.3 shows the conversion between an SST89F5x lock state and its corresponding lock state in the SST89C5x.

TABLE 4.3: SECURITY LOCK CONVERSION

SST89F5x	SST89C5x:	Block 1	Block 0
No Lock	Level 1	No Lock	No Lock
Hard Lock	Level 2	Hard Lock	Hard Lock
Block1 Lock	Level 3	Hard Lock	Soft Lock
Soft Lock	Level 3	Soft Lock	Soft Lock

4.4 Security Lock Example

Below is a security lock example implementing Hard Lock onto SST89C5x.

```
ORL   SFCF, #40h       ;Set IAP Enable Bit
MOV   SFDT, #55h       ;Setup IAP
MOV   SFCM, #0Fh       ;Set Security Lock Bit

DONE:

MOV   A, SFST          ;Get Flash Status Byte
ANL   A, #02h          ;Mask for Flash_Busy
                       ;Bit
CJNE  A, #00h, DONE    ;Loop until operation
                       ;is complete
```

5.0 Conclusion

With the Prescalar, IAP, Watchdog Timer and Security Lock firmware conversions shown in this note, any SST89F5x code can be changed to support SST89C5x microcontrollers.