
Migrating From a 8xC51 Microcontroller to a TSC8x251G1x Microcontroller

By François Fosse
Technical Marketing Engineer

1. Introduction

The C51 microcontroller is one of the most widely used 8-bit microcontroller. Many applications today require a high performance 8-bit microcontroller with more addressing space, register-based instructions, larger internal data RAM, more stack space and effective high level language programming. The new generation C251 microcontroller is designed to fulfill the high performance applications requirements and provide a natural upgrade path for the C51 microcontroller.

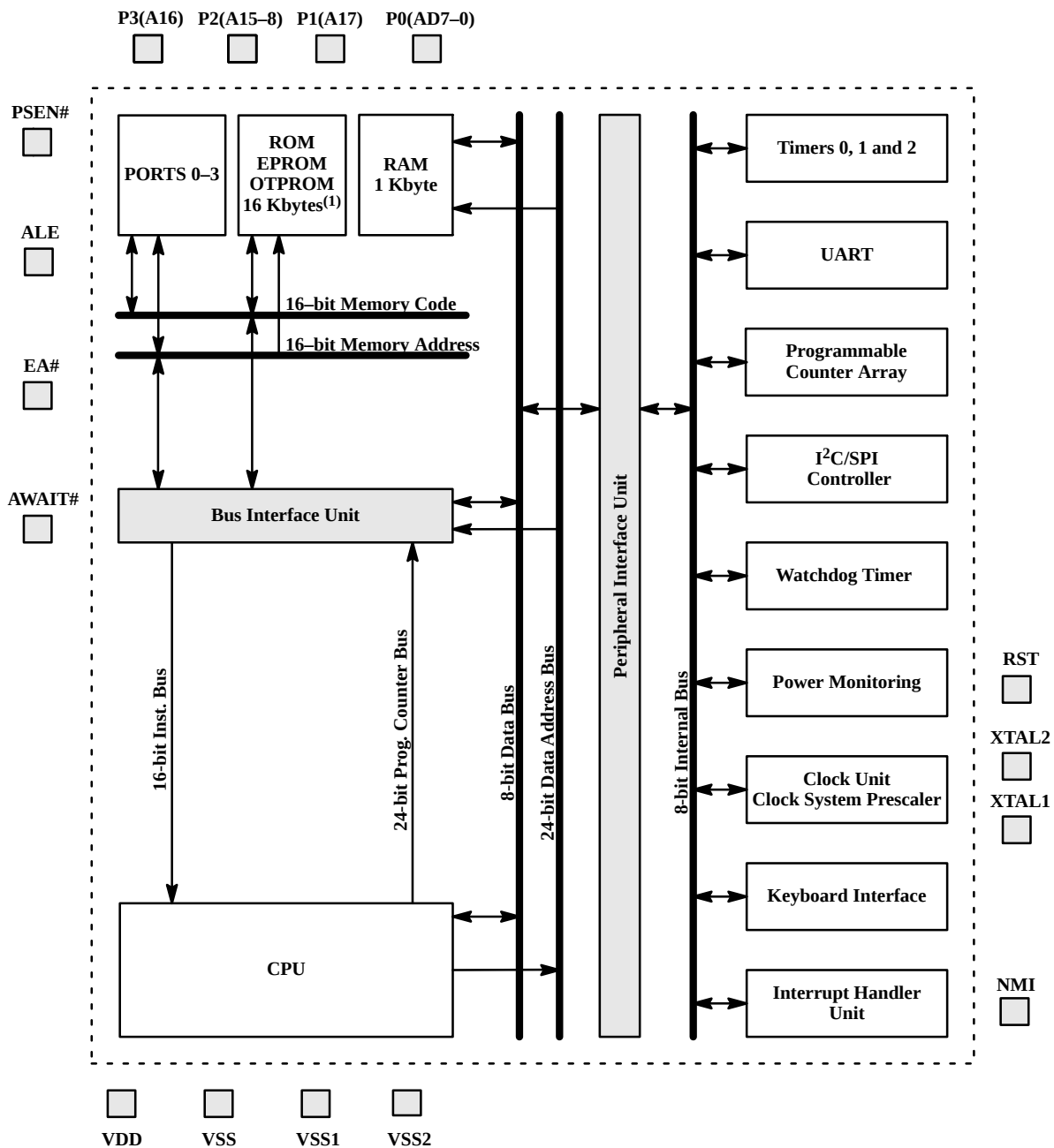
A product derivation from the C251 microcontroller family is the TSC8x251G1x. This product is both binary code compatible and pin compatible with the standard 80C51, 80C52 and 80C51FA/FB/RA/RB products. In the rest of this application note, C51 term is used to designate one of those products. Therefore the TSC8x251G1x can be plugged into an C51 socket and the application will receive an immediate performance boost. The TSC8x251G1x has the same features as the C51 plus some enhanced ones. These features include:

- Three 16-bit Timers/Counters (Timer 0, 1 and 2)
- Programmable Serial Channel with – Framing Error Detection – **Overrun Error Detection** – Automatic Address Recognition – **Dedicated Baud Rate Generator**
- **Programmable Synchronous Serial Link Controller with – I²C Mode – SPI Mode – Dedicated Bit Rate Generator**
- Programmable Counter Array (PCA) with – High Speed Output – Real Time Capture/Compare – Pulse Width Modulation (PWM) – Software Watchdog Timer
- **Secure 14-bit Hardware Watchdog Timer**
- Power Monitoring and Management with – IdleMode – Power Down Mode – **Power-Fail Reset** – **Software Programmable System Clock** – Power-On Reset – Power-Off Flag
- 32 Programmable I/O Lines
- **Keyboard Interrupt interface with – High Level – Low Level**
- **Non Maskable Interrupt Input⁽²⁾**
- **Real Time Wait State Input⁽²⁾**
- **Enriched instruction set with 16 bit and 32 bit capability**
- **TRAP instruction**
- **Additional 8-bit, 16-bit and 32-bit registers with accumulator and index register functionality**
- 16 Kbytes of On-Chip Program Memory
- **1 Kbyte of On-Chip Data Memory**
- **Configurable to 64 Kbytes, 128 Kbytes or 256 Kbytes Extended Addressability**
- **Speed-Up External Code Fetching in the Page Mode**
- **ONCE (On-Circuit Emulation) Mode**

Notes:

1. Features in bold are new or enhanced TSC8x251G1x features.
2. Not available on DIL package and TSC87251G1A.

The TSC8x251G1x provides an easy, low cost, low risk and yet higher performance path for the C51 microcontrollers. The block diagram of the TSC8x251G1x microcontroller is shown in Figure 1.



Note:

1. TSC80251G1D embeds no internal code memory.

Figure 1. TSC8x251G1x Block Diagram

This application note assumes that the user is familiar with C51 microcontrollers.

The purpose of this application note is to address the software and the hardware design considerations when migrating from C51 microcontrollers to the TSC8x251G1x.

The TSC8x251G1x family is composed of three products:

- the TSC87251G1A with on-chip EPROM/OTPROM code memory, based on C251 core revision A (Step A)⁽¹⁾
- the TSC83251G1D with on-chip ROM code memory, based on C251 core revision D (Step D)⁽¹⁾
- the TSC80251G1D with no on-chip code memory, based on C251 core revision D (Step D)⁽¹⁾

The focus is on the plug-in compatibility with the C51. To plug-in the TSC8x251G1x into an C51 socket, the user must consider all the points that are described in the following sections.

The detailed TSC8x251G1x information can be obtained from the Design Guides and Datasheets that are listed in Section 9, page 16.

Note:

1. The differences between TSC87251G1A and TSC8x251G1D are highlighted in this application note.

2. Configuration Bytes

2.1. Description

The TSC8x251G1x provides a variety of features and operating modes by reading two configuration bytes.

Configuration data is retrieved from these bytes just after chip-reset and allows to configure the TSC8x251G1x into the following modes:

- Binary or Source mode
- 0 to 3 Wait States generation⁽¹⁾
- ALE extension
- 2 or 4 bytes interrupt frame
- Extended addressability of 256 Kbytes external memory
- Page Mode
- Mapping the internal code memory for look-up table

In Binary Mode, the TSC8x251G1x is able to run C51 codes without recompilation. The full power of the TSC8x251G1x new instruction set is used in Source Mode. The detail explanation of these two modes is discussed in paragraph 3, page 7.

The TSC8x251G1x can be used to interface with different speed of external devices. The 0 Wait State is used to interface with fast external devices (including memory) and 3 Wait States⁽¹⁾ are used to interface with slow memory and external peripherals. For slower external devices, ALE is extended for an additional state as well. More discussion about the Wait State is explained in the paragraph 4, page 9.

The TSC8x251G1x can be configured to have a 2-byte frame or an extended 4-byte frame pushed on the stack during call to Interrupt Service Routines (ISR). The 2-byte frame option is compatible with C51 and consists of the two bytes of the program Counter (PC). The 4-byte frame consist of the three bytes of the extended PC and the PSW1 byte.

The TSC8x251G1x can be configured to have extended addressability up to 256 Kbytes for external memory devices. The detailed information about this configuration is explained in the paragraph 7, page 11.

In the Page Mode configuration, the TSC8x251G1x reduces the time required for external code fetching cycles to half. More discussion about the page mode design is explained in the Section 7.3, page 13.

One of the configuration bits of the TSC8x251G1x is used to map the upper 8K bytes of internal code memory to data memory addresses 00:E000h–00:FFFFh as look-up table.

Note:

1. TSC87251G1A provides only 0 or 1 wait state generation.

2.2. Location

Depending on the product usage, internal program memory (EA#= 1) or external program memory (EA#= 0), configuration data is retrieved from on-chip or off-chip according to Table 1.

Table 1. Configuration Bytes Location

Microcontroller		Location of configuration bytes	Programming
TSC87251G1A	EA#= 1	On-chip	By user during code programming
	EA#= 0	On-chip	By user during code programming
TSC83251G1D	EA#= 1	On-chip	In factory during code masking
	EA#= 0	Inside external user code memory ⁽¹⁾	By user during external PROM programming
TSC80251G1D ⁽²⁾	EA#= 0	Inside external user code memory ⁽¹⁾	By user during external PROM programming

Notes:

1. Configuration bytes are retrieved from external memory using internal addresses FF:FFF8h and FF:FFF9h, which appear on the microcontroller external address bus (A17, A16, A15:0).
2. The TSC80251G1D has no internal code memory, so can only execute from external memory (EA#= 0).

In romless mode using TSC80251G1D or TSC83251G1D with EA#= 0, Table 2 shows the user's code memory addresses to program depending on the external code memory size.

Table 2. Configuration Byte Location in External Code Memory

Code Memory Size	Configuration Bytes location
16 Kbytes: 0000h–3FFFh	3FF8h–3FF9h ⁽¹⁾
32 Kbytes: 0000h–7FFFh	7FF8h–7FF9h ⁽²⁾
64 Kbytes: 0000h–FFFFh	FFF8h–FFF9h

Note:

1. These addresses assume that memory range B000h–FFFh is a shadow of memory range 0000h–3FFFh (A14 and A15 not decoded).
2. These addresses assume that memory range 8000h–FFFh is a shadow of memory range 0000h–7FFFh. (A15 not decoded).

2.3. Content

Figure 2 and Figure 3 show the bits assignment for the two configuration bytes. The bits that are not defined are reserved for future use. The value found in the reserved bits should be ignored and not to be changed. Table 3 explains the configuration bit setting in configuration byte 0 that control the extended addressing capability of the TSC8x251G1x.

Configuration Byte 0

7	6	5	4	3	2	1	0
—	WSA1#	WSA0#	XALE#	RD1	RD0	PAGE#	SRC

Bit Number	Bit Mnemonic	Description															
7	—	Reserved Set this bit when writing to configuration byte 0.															
6	WSA1#	Wait State A bits⁽¹⁾ Select the number of wait states for RD#, WR# and PSEN# signals for external memory accesses (all regions except 01:). <table><tr><th>WSA1#</th><th>WSA0#</th><th>Number of wait states</th></tr><tr><td>0</td><td>0</td><td>3</td></tr><tr><td>0</td><td>1</td><td>2</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	WSA1#	WSA0#	Number of wait states	0	0	3	0	1	2	1	0	1	1	1	0
WSA1#	WSA0#	Number of wait states															
0	0	3															
0	1	2															
1	0	1															
1	1	0															
5	WSA0#																
4	XALE#	Extend ALE bit Clear to extend the duration of the ALE pulse from T _{Osc} to 3×T _{Osc} . Set to minimize the duration of the ALE pulse to 1×T _{Osc} .															
3	RD1	Memory Signal Select bits Specify a 18-bit, 17-bit or 16-bit external address bus and the usage of RD#, WR# and PSEN# signals (see Table 3).															
2	RD0																
1	PAGE#	Page Mode Select bit Clear to select the faster page mode with A15:8/D7:0 on Port 2 and A7:0 on Port 0. Set to select the non-page mode with A15:8 on Port 2 and A7:0/D7:0 on Port 0.															
0	SRC	Source Mode/Binary Mode Select bit Clear to select the binary mode. Set to select the source mode.															

Note:

- Only WSA0# (named WSA#) is available in TSC87251G1A. WSA1# is reserved and must be set when writing to configuration byte 0.

Figure 2. Configuration Byte 0

Table 3. Address Ranges and Usage of RD#, WR# and PSEN# Signals

RD1	RD0	P1.7	P3.7/RD#	PSEN#	WR#	External Memory
0	0	A17	A16	Read signal for all external memory locations	Write signal for all external memory locations	256 Kbytes
0	1	I/O pin	A16	Read signal for all external memory locations	Write signal for all external memory locations	128 Kbytes
1	0	I/O pin	I/O pin	Read signal for all external memory locations	Write signal for all external memory locations	64 Kbytes
1	1	I/O pin	Read signal for regions 00: and 01:	Read signal for regions FE: and FF:	Write signal for all external memory locations	2 × 64 Kbytes ⁽¹⁾

Note:

- This selection provides compatibility with the standard C51 hardware which has separate external memory spaces for data and code.

Configuration Byte 1

7	6	5	4	3	2	1	0
–	–	–	INTR	WSB	WSB1#	WSB0#	EMAP#

Bit Number	Bit Mnemonic	Description															
7	–	Reserved Set this bit when writing to configuration byte 1.															
6	–	Reserved Set this bit when writing to configuration byte 1.															
5	–	Reserved Set this bit when writing to configuration byte 1.															
4	INTR	Interrupt Mode bit Clear so that the interrupts push two bytes onto the stack (the two lower bytes of the PC register). Set so that the interrupts push four bytes onto the stack (the three bytes of the PC register and the PSW1 register).															
3	WSB	Wait State B bit Clear to generate one wait state for memory region 01:. Set for no wait states for memory region 01:.															
2	WSB1#	Wait State B bits⁽¹⁾ Select the number of wait states for RD#, WR# and PSEN# signals for external memory accesses (only region 01:). <table><tr><th>WSB1#</th><th>WSB0#</th><th>Number of wait states</th></tr><tr><td>0</td><td>0</td><td>3</td></tr><tr><td>0</td><td>1</td><td>2</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	WSB1#	WSB0#	Number of wait states	0	0	3	0	1	2	1	0	1	1	1	0
WSB1#	WSB0#		Number of wait states														
0	0		3														
0	1		2														
1	0	1															
1	1	0															
1	WSB0#																
0	EMAP#	On–Chip Code Memory Map bit Clear to map the upper 8 Kbytes of on–chip code memory (at FF:2000h–FF:3FFFh) to the data space (at 00:E000h–00:FFFFh). Set not to map the upper 8 Kbytes of on–chip code memory (at FF:2000h–FF:3FFFh) to the data space.															

Notes:

1. Not implemented in TSC87251G1A. These bits are reserved and must be set when writing to configuration byte 1.

Figure 3. Configuration Byte 1

2.4. C51 configuration

To replace a C51 by a TSC8x251G1x:

- Select a configuration data (see examples below).
- Program the configuration bytes according to Table 1 and Table 2.

2.4.1. Example 1

To replace a 80C31 by a TSC80251G1D in romless mode, the following configuration can be applied:

- SRC= 0 Binary Mode compatible with 80C31
- PAGE= 1 No page mode
- RD1:0= 11 2 separate 64 Kbytes code and data memories
- XALE= 0 ALE with 0 wait state⁽¹⁾⁽²⁾⁽³⁾
- WSA1:0#= 10 PSEN# with 1 wait state for all regions except 01:⁽²⁾
- EMAP= 1 No internal code memory mapping
- WSB1:0#= 11 RD#/WR# with 0 wait state for region 01:⁽³⁾
- INTR= 0 2 bytes interrupt frame

Notes:

1. This configuration choice depends on the external latch timings, usually no ALE extension is performed.
2. This configuration choice depends on the external program memory access time.
3. This configuration choice depends on the external data memory access time.

This leads to a configuration value DEh for configuration byte 0 and E7h for configuration byte 1.

2.4.2. Example 2

To replace a 87C51FB by a TSC87251G1A, the following configuration can be applied:

- SRC= 0 Binary Mode compatible with 87C51FB
- PAGE= 1 No page mode⁽¹⁾
- RD1:0= 11 2 separate 64 Kbytes code and data memories⁽¹⁾
- XALE= 0 ALE with 0 wait state⁽¹⁾⁽²⁾⁽³⁾⁽⁴⁾
- WSA#= 0 PSEN# with 1 wait state for all regions except 01:⁽¹⁾⁽³⁾
- EMAP= 1 No internal code memory mapping⁽¹⁾
- WSB#= 1 RD#/WR# with 0 wait state for region 01:⁽¹⁾⁽⁴⁾
- INTR= 0 2 bytes interrupt frame

Notes:

- 1.. These bits must be set (default) when TSC87251G1A executes only internally.
2. This configuration choice depends on the external latch timings, usually no ALE extension is performed.
3. This configuration choice depends on the external program memory access time.
4. This configuration choice depends on the external data memory access time.

This leads to a configuration value DEh for configuration byte 0 and E7h for configuration byte 1.

3. Code Compatibility

3.1. Description

The TSC8x251G1x can be configured in either Binary mode or Source mode. When TSC8x251G1x is configured in Binary mode, it is binary code compatible with C51 microcontroller.

There are a total of 255 instruction mnemonics implemented in the 256 hex codes available in the C51 microcontroller. The reserved hex location is A5h. For TSC8x251G1x, A5h is used as an Escape code <ESC>, a window location to generate more hex codes for new instructions.

When configured in Binary mode (see Figure 4), all the new TSC8x251G1x instructions are prefixed with an A5h. Therefore, the length of the hex code size for new instructions is increased by one byte compared to hex code in source mode. All the hex code for C51 microcontroller instructions are identical to TSC8x251G1x. A few hex code examples are shown in paragraph 3.2.

When configured in Source mode (see Figure 5), all the new TSC8x251G1x instructions do not have the A5 prefix. On the other hand all the C51 microcontroller instructions in Register addressing mode now require the prefix. Hence, the length of the hex code for C51 microcontroller instructions in the Register addressing mode is increased by one byte. A few hex code examples are shown in paragraph 3.2.

To configure TSC8x251G1x in Binary mode, the configuration bit, SRC of configuration byte 0 must be set to 0. In this mode, TSC8x251G1x can execute C51 application code directly without any modification, rewrite or new compilation.

To configure TSC8x251G1x in Source mode, the configuration bit, SRC of configuration byte 0 must be set to 1. For Source Mode, either existing C51 codes or/and TSC8x251G1x codes need to be assembled/compiled with a C251 microcontroller assembler/compiler. The TSC8x251G1x will perform better in Source Mode if a majority of the code is based on new C251 instructions.

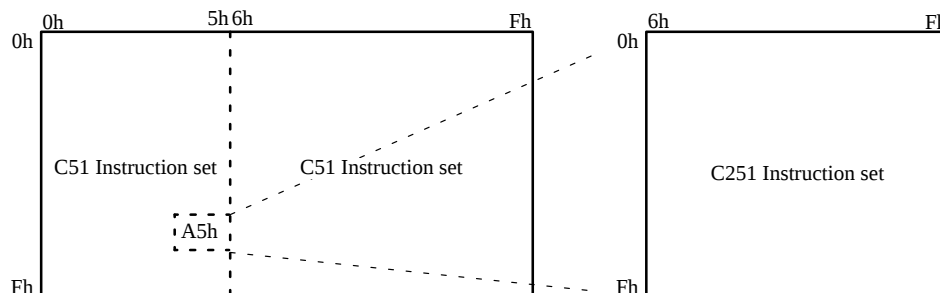


Figure 4. Binary Mode Opcode Map

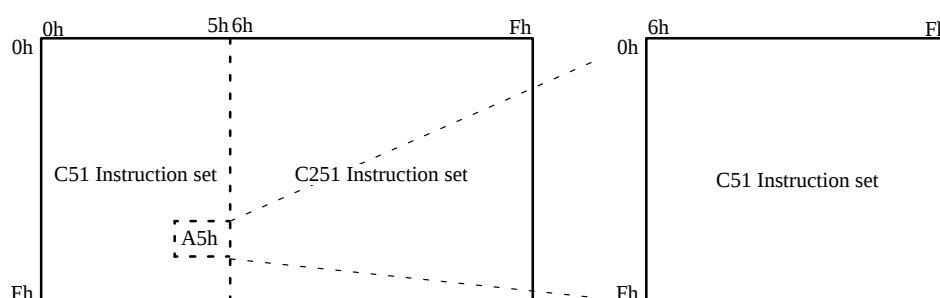


Figure 5. Source Mode Opcode Map

3.2. Example

Table 4 shows a few examples of hex code for the TSC8x251G1x instructions.

Table 4. Hex Code Examples for TSC8x251G1x Instructions

Instruction	In Binary mode	In Source mode
INC A	04	04
INC @R1	07	A5 07
INC R0	08	A5 08
JSLE rel	A5 08	08
ADD A,direct	25	25
ADD A,R0	28	A5 28
JLE rel	A5 28	28

4. Wait State Generation

4.1. Description

When interfacing the TSC8x251G1x with slow devices, wait states can be programmed to extend the external system bus cycle. The TSC8x251G1x can be configured to generate 0 or 1 wait state for ALE and 0 to 3 wait states⁽¹⁾ for PSEN#/RD#/WR# signals.

Once the ALE signal is extended to 1 wait state, the ALE pulse width is 3 clock periods whenever an external access to the memory occurs. The ALE signal is only output when accessing to external memory devices only. When accessing to the internal memory, there is no ALE signal output. This is different from the C51 where the ALE signal is always generated even though it is executing from or accessing the internal memory. The configuration bit, XALE is used to configure the length of the ALE pulse. When XALE is set to 1, the ALE pulse width is 1 clock period. The ALE pulse width becomes 3 clock periods when the XALE bit is set to 0.

For PSEN#/RD#/WR# signals, different wait state can be programmed for separate address regions to interface with various speed of memories and peripheral devices. 1 to 3 wait states⁽¹⁾ can be used to interface with slower memory devices or peripherals. The configuration bits, WSA_n# and WSB_n# configure different wait states for separate address regions. WSA_n# apply to the PSEN#/RD#/WR# signals in 00:0000–00:FFFF, FE:0000–FE:FFFF and FF:0000–FF:FFFF address regions (see Figure 2). WSB_n# apply to the PSEN#/RD#/WR# signals in 01:0000–01:FFFF address region (see Figure 3).

Note:

1. TSC87251G1A provides only 0 or 1 wait state generation.

4.2. Example

The TSC80251G1D (romless part) is interfacing with external peripherals that are slower than the external memory (EPROM). The address region for 00:0000–00:041F is assigned for internal 1-Kbyte RAM and FF:0000–FF:FFFF is assigned to the external memory (EPROM). Since the access time for external memory (EPROM) is faster than the external peripherals, the address region for 00:0000–00:FFFF and FF:0000–FF:FFFF are programmed with 0 wait state. The address region 01:0000–01:FFFF is reserved for the external peripheral devices that are slower than the external memories. This address region is programmed with 2 wait states and the address for external peripheral devices is assigned to this region.

Configure WSA1:0# to 11 and WSB1:0# to 01. Program the external EPROM at address region FF:0000–FF:FFFF. Map all the external peripheral devices to address region 01:0000–01:FFFF.

5. Extended PC, SP and PSW

The TSC8x251G1x has a 24-bit Program Counter (PC), 16-bit Stack Pointer (SP) and two Program Status Word registers, PSW and PSW1. PSW1 holds flags also defined in PSW, but the additional new flags are Negative (N) and Zero (Z) flags. Since the size of the PC and PSW1 are different from the C51, the TSC8x251G1x can be configured to have 2-byte or 4-byte frame pushed on the stack during call to Interrupt Service Routine. Configuration bit INTR of the configuration byte 1 is assigned for this configuration (see Figure 3).

To be compatible with C51, INTR must be configured to 0. In this configuration, 2 bytes for PC are pushed on the stack during the Interrupt Service Routine vectorization. Execution of RETI instruction pops the 2 bytes from the stack.

To protect the contents of PSW1 and allow TSC8x251G1x to interface with 128 or 256 Kbytes external memory, the INTR bit is programmed to 1. In this configuration, 4 bytes: 3 bytes for PC and 1 byte for PSW1 are pushed on the stack during the Interrupt Service Routine vectorization. These 4 bytes are pushed in the following sequence: PSW1, high byte of PC, low byte of PC and middle byte of PC. Execution of RETI instruction pops the 4 bytes from the stack.

6. Timing Loop/Sequence consideration

6.1. Description

The TSC8x251G1x uses the new C251 microcontroller core architecture that is different from the traditional C51 microcontroller's architecture. It is designed based on a pipelined architecture and a register-based machine. Therefore the execution time is different from the standard C51 architecture. With the pipeline full, the average execution time falls down to 1 instruction by state (or 2 oscillator clock periods) when running from internal code memory. This average execution time is 2 states (or 4 oscillator clock periods) when running from an external code memory in non-page mode.

Since a C51 takes 6 states (or 12 oscillator clock periods) per machine cycle to execute most of the instructions, the instruction execution time for TSC8x251G1x has been significantly improved. Therefore, users must consider changing the timing loop or sequence of C51 code used in the TSC8x251G1x application.

6.2. Example

A delay time is used to provide a 13 μs delay in a 12 MHz C51 application. The code is executed from internal memory. The code and the time taken for execution are shown on Table 5.

Table 5. Loop Executed at 12 MHz on a C51

; Internal or external Code Execution		Time taken in number of cycles
	mov R0, #06h	; 1
Loop:	djnz R0, Loop	; 2
; 1 cycle= 12 oscillator periods		
; Total oscillator clock periods= [1 + (6 x 2)] x 12= 156		
; Total time taken in the timing loop= 156 x 83.33 ns= 13 μs		

In the same application, the C51 microcontroller is replaced by a TSC87251G1A configured in binary mode. No modification is made to the code shown on Table 5. But to simplify timing loop computation, the code is executed in internal memory. The code and the time needed for execution are shown on Table 6.

Table 6. Loop Executed at 12 MHz on a TSC87251G1A Without Code Modification

; Internal Binary Code Execution		Time taken in number of states
	mov R0, #06h	; 1
Loop:	djnz R0, Loop	; 5 and 2 for the loop exit
; 1 state= 2 oscillator periods		
; Total oscillator clock periods= [1 + (5 x 5 + 2)] x 2= 56		
; Total time taken in the timing loop= 56 x 83.33 ns= 4.66 μs		

To maintain the same delay time, some changes in the assembly code are needed. The new code and the time needed for execution are shown on Table 7.

Table 7. Loop executed at 12 MHz on TSC80251G1 With Code Modification

; Internal Binary Code Execution		Time taken in number of states
	mov R0, #0Fh	; 1
Loop:	djnz R0, Loop	; 5 and 2 for the loop exit
; 1 state= 2 oscillator periods		
; Total oscillator clock periods= [1 + (15 x 5 + 2)] x 2= 156		
; Total time taken in the timing loop= 156 x 83.33 ns= 13 µs		

The execution time depends on the code location: internal or external. Table 8 illustrates the fetch timings for two bytes of code.

Table 8. Times to Fetch Two Bytes of Code

Code Memory Location	State Times
On-chip Code Memory	1
External Memory in page mode ⁽¹⁾⁽²⁾	2 + 2N ⁽³⁾
External Memory in non-page mode	4 + 2N ⁽³⁾

Notes:

1. To use existing 80C51 applications without changing your hardware, please execute code in non-page mode.
2. In page-hit condition.
3. N is the number of wait state programmed, refer to paragraph 4. page 9.

7. External Memory Access

7.1. Description

The system bus cycle for the TSC8x251G1x is different from the C51. It can be programmed in non-page mode or page mode using PAGE# bit in configuration byte 0 (see Figure 2).

The non-page mode is pin compatible with the C51 hardware. Once page mode is configured, the TSC8x251G1x is no more pin compatible with C51.

Figure 6 shows the Port 0 and 2 connection with external memory in a non-page and page mode designs.

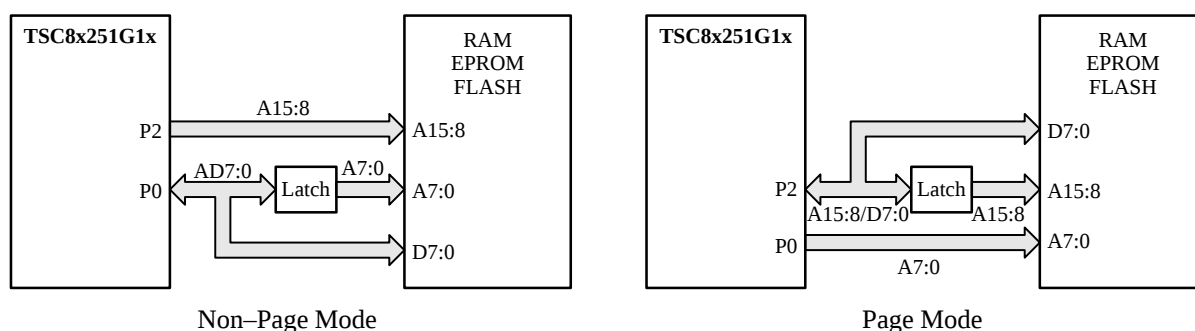


Figure 6. Bus Structure in Non-Page Mode and Page Mode

7.2. Non-Page Mode design

In non page mode, data are multiplexed with the low byte of address (A0–A7) and is generated from Port 0 (see Figure 6). During non-page mode operation, the TSC8x251G1x only requires 2 states to fetch an instruction.

Table 9 shows the number of states for each bus cycles for the TSC8x251G1x (in non-page mode) and a C51.

Figure 7 to Figure 9 show the system bus timings for TSC8x251G1x in non-page mode.

Table 9. Bus Cycle States (No Wait States)

Bus Cycle	C51 States	TSC8x251G1x States
Code Read	3	2
Data Read	6	3
Data Write	6	3

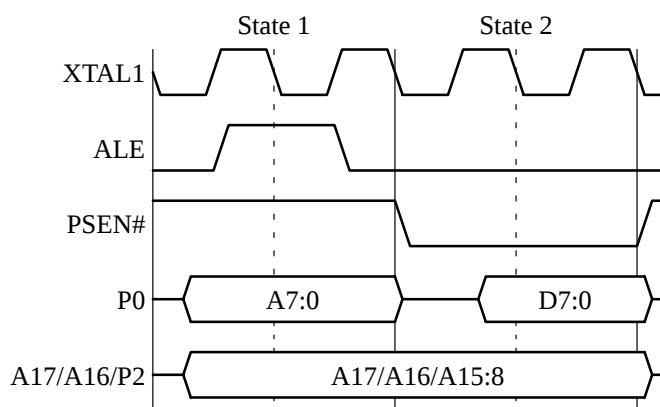


Figure 7. External Code Fetch (Non-Page Mode)

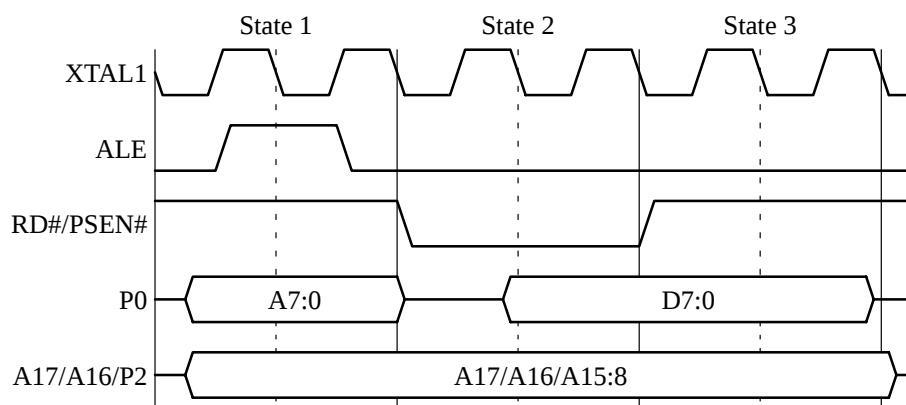


Figure 8. External Data Read (Non-Page Mode)

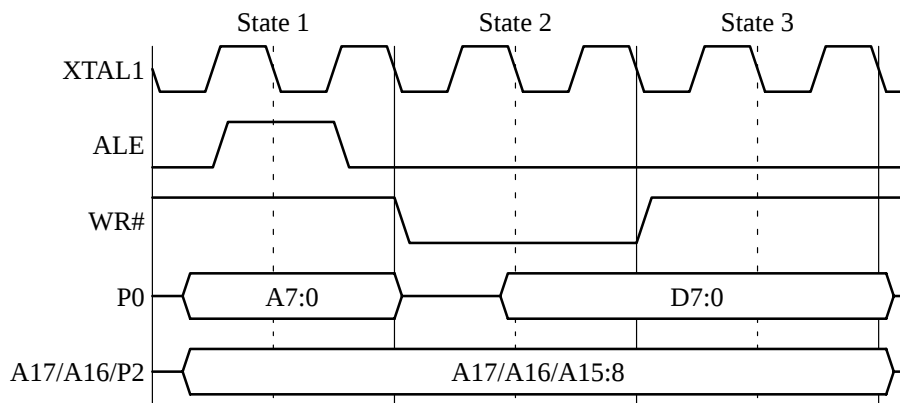


Figure 9. External Data Write (Non-Page Mode)

7.3. Page Mode design

In page mode, data are multiplexed with the high byte of address (A8–A15) and is generated from Port 2 (see Figure 6). During page mode operation, the TSC8x251G1x requires only 1 state to fetch an instruction in the page hit situation. A page consists of 256 bytes. A page hit occurs starting from the second byte until the last byte of a 256-byte page. The ALE signal is only generated in the first byte of 256-byte page. The high byte of the address (A8–A15) is held by an external latch for 256 byte of code fetching.

The page miss occurs in the first byte. It requires 2 states to fetch the first code. The page miss situations are listed as below:

- Conditionnal page miss situations
 - fetch of the first byte after a page rollover
 - fetch of the first byte after the power down and idle mode
 - fetch of the first byte after a read or a write data access
 - fetch of the first byte after a return instruction (ret, reti, eret)
- Systematic page miss situations
 - execution of a branch instruction
 - call to a subroutine or an interrupt routine

Figure 10 to Figure 12 show the system bus timings for TSC8x251G1x in page mode.

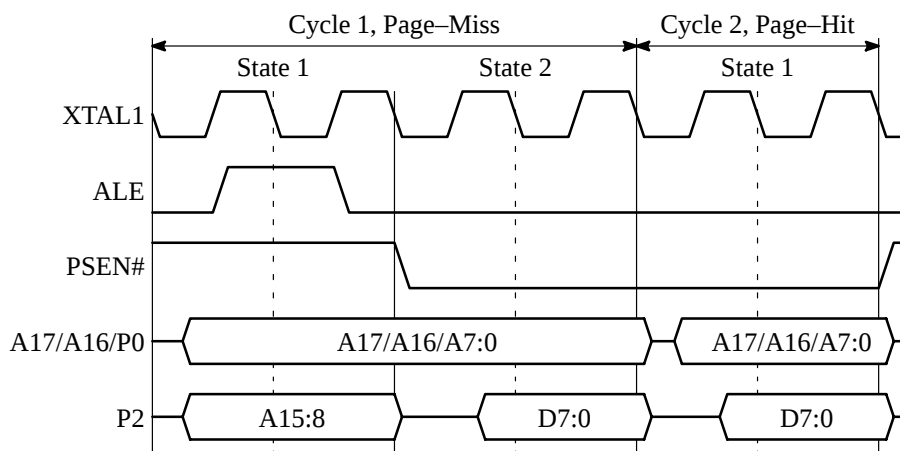


Figure 10. External Code Fetch (Page Mode)

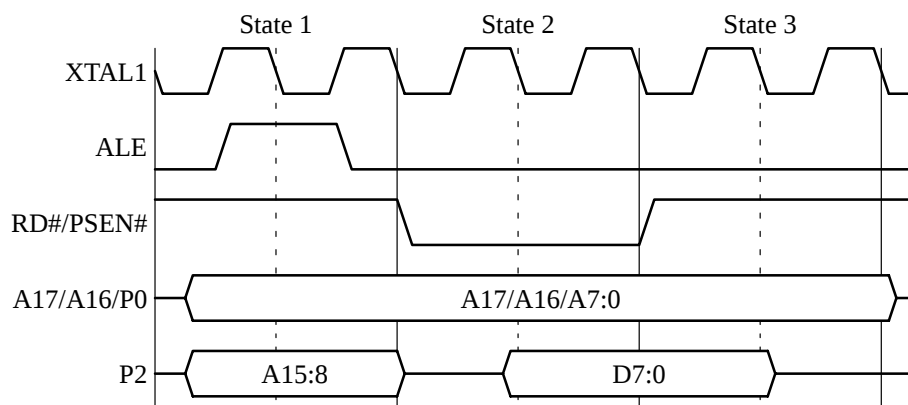


Figure 11. External Data Read (Page Mode)

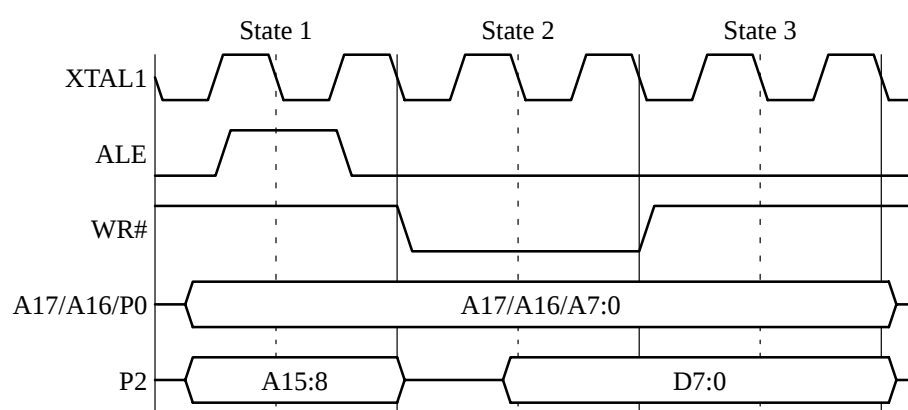
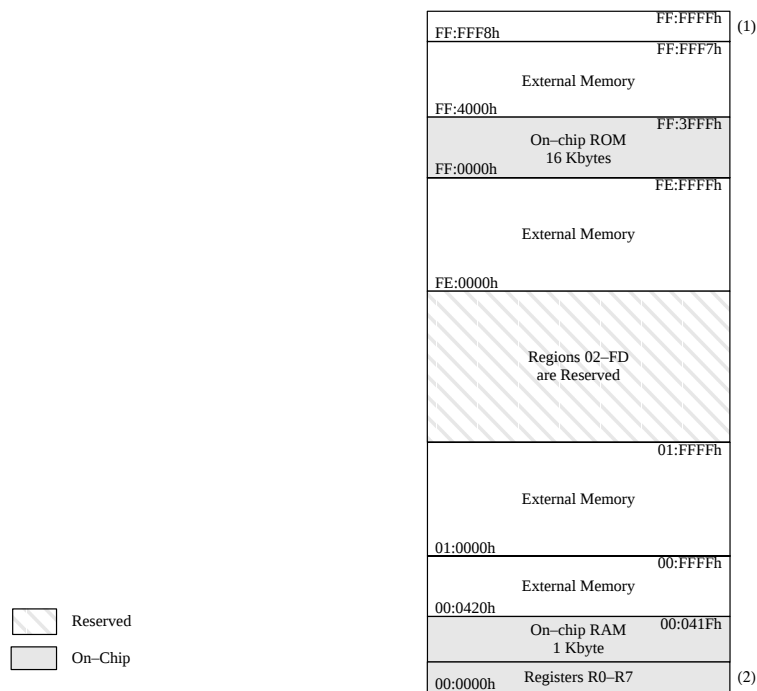


Figure 12. External Data Write (Page Mode)

7.4. Extended Memory Space

The TSC8x251G1x provides 256 Kbytes linear address space. The 256 Kbytes linear address spaces are at regions 00:0000h–00:FFFFh, 01:0000h–01:FFFFh, FE:0000h–FE:FFFFh and FF:0000h–FF:FFFFh (see Figure 13). At any time, the maximum address space available for external memory can be either 64, 128 or 256 Kbytes. To interface TSC8x251G1x with 256K external memory devices, the configuration bits RD1:0 of configuration byte 0 must be programmed to 00 (see Table 3). In this mode the RD# pin functions as address bit A16, P1.7 functions as address bit A17. All the external code or data memory spaces are strobed by PSEN#.

In this mode, out of the 24-bit Program Counter (PC), 18-bit will be used for extended address. To jump beyond 64K address region, EJMP, ECALL and ERET must be used. However, it is mandatory not to use EJMP on the reset vector if the INT0 is used at the same time. This is because the EJMP will take 4 bytes and the Interrupt vector for INT0 is at address 0003h. The address for the last byte of the EJMP instruction will overlap with the INT0 Interrupt vector. The code space can be internal or external.



Notes:

1. Eight-byte configuration array (FF:FFF8h – FF:FFFFh).
2. Four banks of registers R0–R7 (32 BYTES, 00:0000h – 00:001Fh).

Figure 13. Memory Mapping in the TSC8x251G1x Address Space

8. Pin-to-Pin Replacement

TSC8x251G1x provides direct plug-in replacement of a C51. Moreover, the C51 in PLCC and QFP packages has four NC (Not Connected) pins which are used on TSC8x251G1x as VSS1 (#1), AWAIT# (#12)⁽¹⁾, VSS2 (#23) and NMI (#34)⁽¹⁾. To avoid any electrical conflict with those signals, it is mandatory that no user's signals being routed through these NC pins. If it is not the case, do not connect these four pins to the board.

Note:

1. The TSC87C251G1A does not provide the AWAIT# and NMI Features: in PLCC and QFP packages, pin #12 and pin #34 are Reserved.

9. References

- TSC80251G1 Design Guide, Atmel Wireless & Microcontrollers
- TSC80251G1A Datasheet, Atmel Wireless & Microcontrollers
- TSC80251G1D Design Guide, Atmel Wireless & Microcontrollers
- TSC80251G1D Datasheet, Atmel Wireless & Microcontrollers
- TSC80251 Programmer's Guide, Atmel Wireless & Microcontrollers

10. Technical Support

- C251 product hotline E-mail: C251@atmel-wm.com
- C251 tools hotline E-mail: x51_tools@atmel-wm.com

11. Sites to visit

- Atmel Wireless & Microcontrollers microcontrollers web pages: <http://www.atmel-wm.com>