TSK51x MCU

Summary

The TSK51x is a fully functional, 8-bit microcontroller, incorporating the Harvard architecture. This core reference includes architectural and hardware descriptions, instruction sets and on-chip debugging functionality for the TSK51x family.

Core Reference
CR0115 (v2.0) March 13, 2008

The TSK51x is the core of a fast, single-chip, 8-bit microcontroller, which executes all ASM51 instructions and is instruction set compatible with the 80C31. The TSK51x serves software and hardware interrupts, provides an interface for serial communications and incorporates a timer system.

Important Notice: Supply of this soft core under the terms and conditions of the Altium End-User License Agreement does not convey nor imply any patent rights to the supplied technologies. Users are cautioned that a license may be required for any use covered by such patent rights

Features

- Control Unit
  - 8-bit Instruction decoder.
- Arithmetic Logic Unit
  - 8 bit arithmetic operations
  - 8 bit logical operations
  - Boolean manipulations
  - 8 x 8 bit multiplication
  - 8 / 8 bit division.
- 32-bit Input/Output ports
  - Four 8-bit I/O ports
- Two 16-bit Timer/Counters
- Serial Peripheral Interfaces in full duplex mode
  - Synchronous mode, fixed baud rate
  - 8-bit UART mode, variable baud rate
  - 9-bit UART mode, fixed baud rate
  - 9-bit UART mode, variable baud rate
  - Multiprocessor communication.
- Interrupt Controller
  - Two Priority Levels
  - Five interrupt sources.
- Internal memory interface
  - Can address up to 64KB of Internal Program memory space.
  - Can address up to 256 bytes of Read/Write Data memory Space.
- External memory interface
  - Can address up to 64KB of External Program memory Space
  - Can address up to 64KB of External Data memory Space.
- Special Function Registers interface
  - Services up to 107 External Special Function Registers

Available Devices

Both standard and debug-enabled (OCD) versions of the microcontroller are available – the TSK51A and TSK51A_D respectively. These devices can be found in the FPGA Processors integrated library (`FPGA Processors.IntLib`), located in the `\Library\Fpga` folder of the installation.

Architectural Overview

Symbols

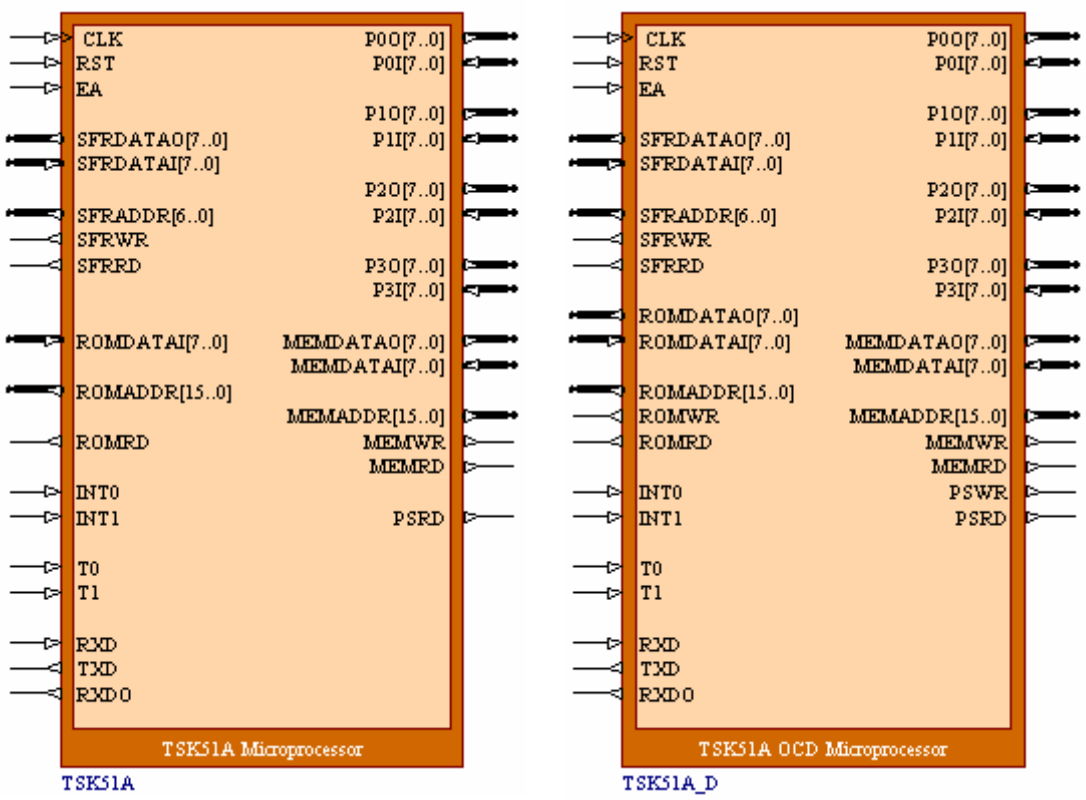


Figure 1. TSK51x family symbols

Pin Description

The pinout of the TSK51x has not been fixed to any specific device I/O - allowing flexibility with user application. The TSK51x contains only unidirectional pins - inputs or outputs.

Table 1. TSK51x Pin description

Name	Type	Polarity/Bus size	Description
Control Signals			
CLK	I	Rise	External system clock (used for internal clock counters and all other synchronous circuitry)
RST	I	High	External system reset. A high on this pin for two clock cycles while the external system clock (CLK) is running resets the device.
EA	I	High	External Access Enable. EA must be externally held High to enable the device to fetch code from external Program memory (0000h - FFFFh). If EA is held Low, the device executes from internal Program memory unless the Program Counter contains an address greater than 0FFFh.
External Special Function Registers Interface Signals			
SFRDATAO	O	8	SFR data bus output

Name	Type	Polarity/Bus size	Description
SFRDATAI	I	8	SFR data bus input
SFRADDR	O	7	SFR address bus
SFRWR	O	High	SFR write enable
SFRRD	O	High	SFR output enable
Internal Program Memory Interface Signals			
ROMDATAI	I	8	Memory data bus input
ROMDATAO[1]	O	8	Memory data bus output
ROMADDR	O	16	Memory address bus
ROMWR[1]	O	High	Memory write enable
ROMRD	O	High	Memory output enable
Interrupt Signals			
INT0	I	Rise/High	External interrupt 0. Interrupt type (rising edge or High level) is determined by setting or clearing bit 0 (IT0) in the TCON register, respectively
INT1	I	Rise/High	External interrupt 1. Interrupt type (rising edge or High level) is determined by setting or clearing bit 2 (IT1) in the TCON register, respectively
Timer Signals			
T0	I	Fall	Timer 0 external clock input
T1	I	Fall	Timer 1 external clock input
Serial Interface Signals			
RXD	I	-	Serial port 0 input (receive)
TXD	O	-	Serial port 0 output (transmit)
RXDO	O	-	Serial port 0 output (transmit in Mode 0)
I/O Port Interface Signals			
P0O P0I	O I	8 8	**Port 0** is an 8-bit bi-directional I/O port with separated inputs and outputs.
P1O P1I	O I	8 8	**Port 1** is an 8-bit bi-directional I/O port with separated inputs and outputs.
P2O P2I	O I	8 8	**Port 2** is an 8-bit bi-directional I/O port with separated inputs and outputs.
P3O P3I	O I	8 8	**Port 3** is an 8-bit bi-directional I/O port with separated inputs and outputs.
External Memory Interface Signals			
MEMDATAO	O	8	External memory output
MEMDATAI	I	8	External memory input

[1] TSK51A_D only

Name	Type	Polarity/Bus size	Description
MEMADDR	O	16	External address bus
MEMWR	O	High	External Data memory write enable
MEMRD	O	High	External Data memory output enable
PSWR[2]	O	High	External Program memory write enable
PSRD	O	High	External Program memory output enable

Memory Organization

Memory in the TSK51x is organized into three distinct areas:

- Program memory (internal ROM or external ROM)
- External Data memory (external RAM)
- Internal Data memory (internal RAM).

Program Memory

The TSK51x can address up to 64KB of Program memory, implemented as either internal ROM, external ROM, or a combination of both.

After a reset has been issued, the CPU starts program execution from location 0000h.

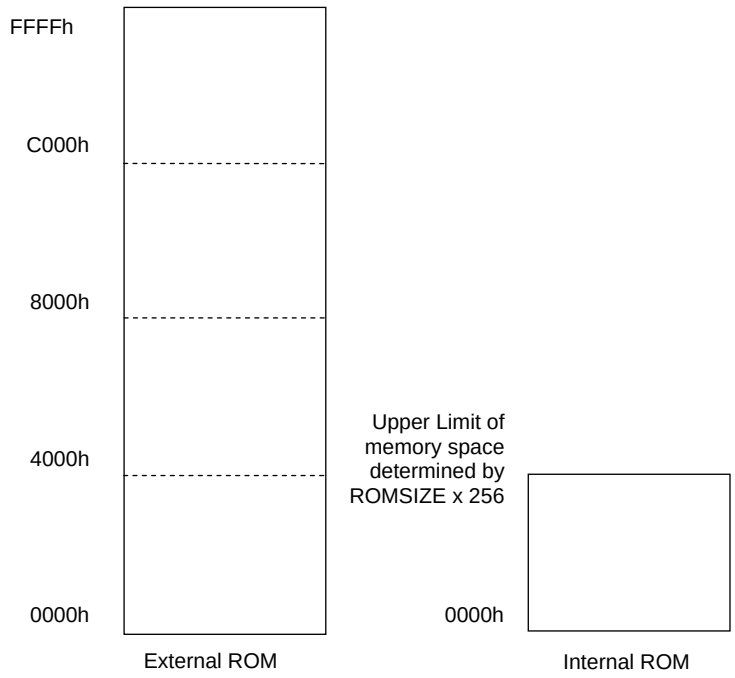


Figure 2. Program memory map

Up to 64KB of internal Program memory space can be addressed. The actual size of the memory space is determined by the value stored in the ROMSIZE register (see 0) and is calculated as:

Internal Program memory = ROMSIZE x 256

The size of internal Program memory is therefore under direct control of software. By default, after a reset, the ROMSIZE register contains the value 10h, which yields a memory space of 4KB. To increase or decrease this size, simply load the ROMSIZE register with the appropriate value.

[2] TSK51A_D only

Program code can be fetched from external or internal Program memory. This selection is made by strapping pin EA (External Address) to VCC or GND respectively. Note that EA can be changed at anytime whilst the processor is running, giving full control over the particular memory space used.

If EA is held High, all the program code is fetched from external memory. If EA is held Low, the lowest n bytes of program code is fetched from internal ROM, where n is the result of ROMSIZE x 256.

When the extent of internal memory space is reached, program code will then automatically be fetched from external memory space. The Program Counter is not reset however, so code will be fetched from the next memory address, but within external memory space.

If the ROMSIZE register contains 00h, the fetch will automatically default to the external Program memory, even if EA is Low.

The lower part of the Program memory includes interrupt and reset vectors. The interrupt vectors are spaced at 8-byte intervals, starting from 0003h for External Interrupt 0.

Table 2. Reset vectors

Location	Service
0003h	External Interrupt 0
000Bh	Timer 0 overflow
0013h	External Interrupt 1
001Bh	Timer 1 overflow
0023h	Serial Port Interrupt

These locations may be used for program code, if the corresponding interrupts are not used (disabled).

When using Internal Program memory, a separate block is placed in the design – external to the component symbol for the core. With the standard version of the core (TSK51A), a block of ROM is used, the size of which depends on the requirements of the design. With the OCD version (TSK51A_D), because this version of the core allows you to write to Program memory space, RAM must be used instead, as shown in Figure 3.

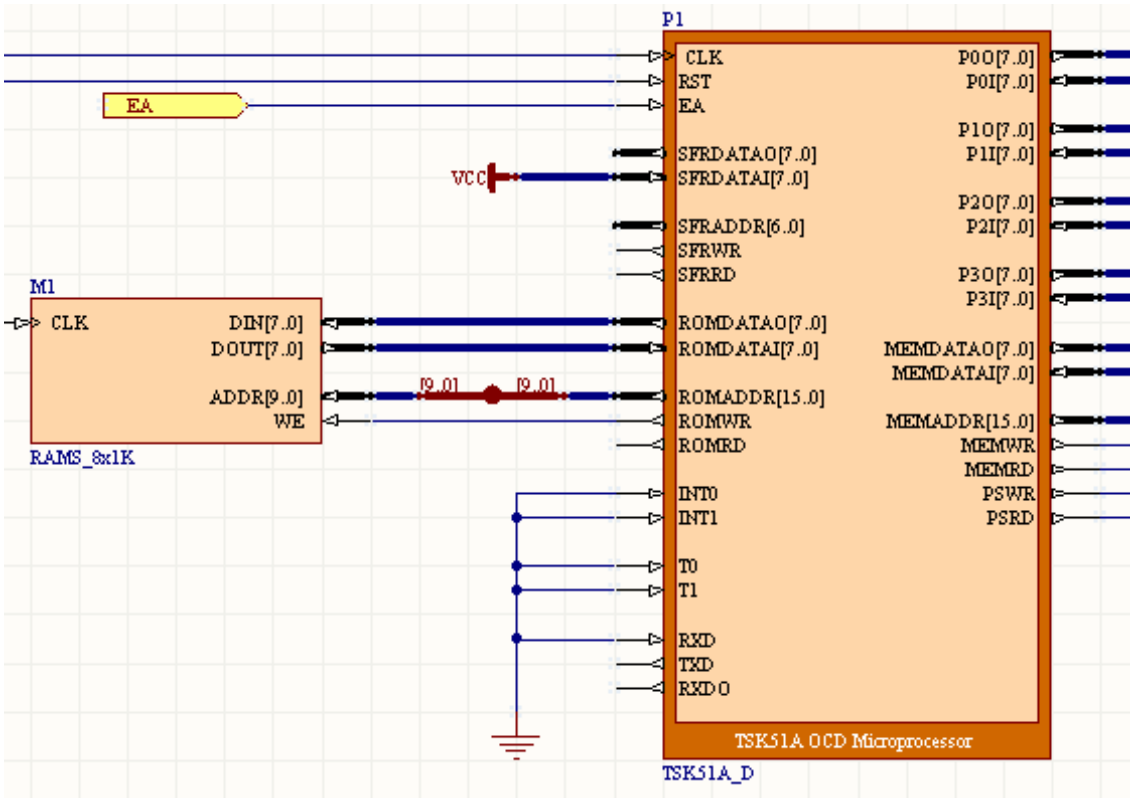


Figure 3. Using RAM for TSK51A_D internal Program memory

RAM and ROM blocks can be found in the FPGA Memories integrated library (`\Library\Fpga\FPGA Memories.IntLib`).

Data Memory

External Data Memory

The TSK51x Microcontroller core incorporates the Harvard architecture, with separate program (code) and data spaces:

- The code from external Program memory is fetched by strobing the PSRD pin.
- Data is read from external Data memory by strobing the MEMRD pin and written to external Data memory by strobing the MEMWR pin.
- The external Data memory space can be accessed directly, through the 16 bit Data Pointer Register (DPTR), or indirectly, using register R0 or R1 and the external Data memory paging register, XP.
- Data is read back on the MEMDATAI bus.

Internal Data Memory

The TSK51x has a 256 byte block of RAM dedicated for use as internal Data memory. This RAM cannot be upgraded in size. The internal Data memory interface is therefore not exposed to the user through the schematic symbol.

The 256 bytes of memory space (00h to FFh) can be accessed by either direct or indirect addressing (where supported). An internal Data memory address is always 1 byte in width.

The upper 128 bytes contain the Special Function Registers (SFRs). This area of internal Data memory is accessible only by direct addressing.

The lower 128 bytes contain work registers and bit-addressable memory. The lower 48 bytes of this area of memory space are further divided as follows:

- The lower 32 bytes (00h – 1Fh) form four banks of eight registers (R0-R7). The RS0 and RS1 bits in the Program Status Word register (PSW) select which bank is currently in use.
- The next 16 bytes (20h – 2Fh) form a block of bit-addressable memory space, covering the bit address range 00h-7Fh.

All of the bytes in this lower half of the internal Data memory space are accessible through direct or indirect addressing.

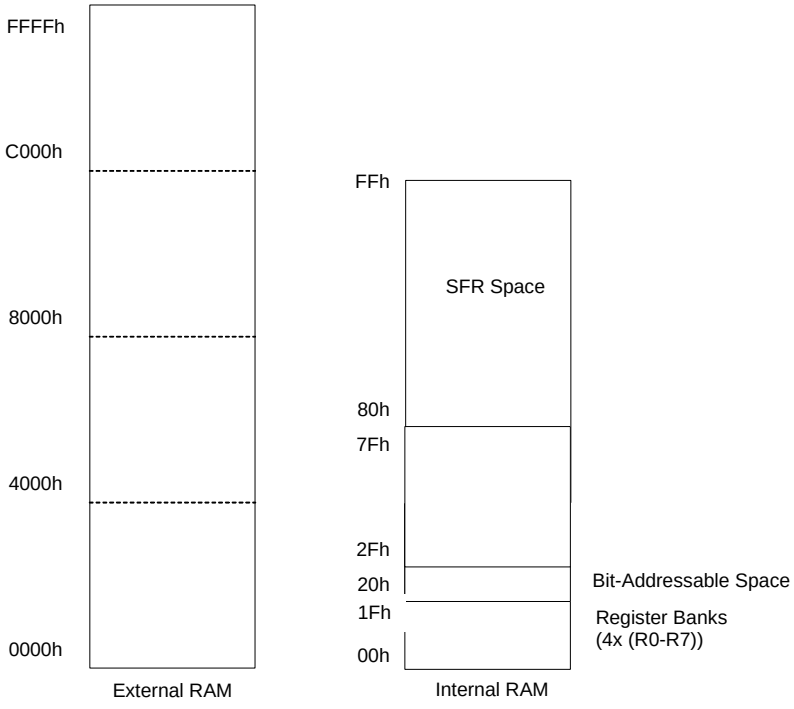


Figure 4. Data memory map

Special Function Registers

A map of the Special Function Registers is shown in Table 3. Only a few addresses are occupied, the others are not implemented. Read access to unimplemented addresses will return undefined data, while writing to them will have no effect.

Table 3. Special Function Registers location

Hex\Bin	X000	X001	X010	X011	X100	X101	X110	X111	Bin/Hex
F8									**FF**
F0	B								**F7**
E8									**EF**
E0	ACC								**E7**
D8									**DF**
D0	PSW								**D7**
C8									**CF**
C0									**C7**
B8	IP								**BF**
B0	P3								**B7**
A8	IE								**AF**
A0	P2								**A7**
98	SCON	SBUF						XP	**9F**
90	P1								**97**
88	TCON	TMOD	TL0	TL1	TH0	TH1		ROMSIZE	**8F**
80	P0	SP	DPL	DPH				PCON	**87**

Accumulator (ACC)

Most instructions use the Accumulator to hold the operand. Note that the mnemonics for Accumulator-specific instructions refer to the Accumulator as A, not ACC.

B register

The B register is used during multiply and divide instructions. It can also be used as a scratch-pad register to hold temporary data.

External Data memory Paging Register (XP)

The content of the XP register is loaded onto the high order byte of the external memory address bus (MEMADDR) during a MOVX @Ri instruction. The XP register is used to implement paging and can provide access to up to 256 pages in external Data memory. Each page can contain up to 256 bytes of data – dependent on the contents of the register Ri. Therefore the maximum addressable Data memory space is 64KB.

Program Status Word Register (PSW)

Table 4. PSW register flags

MSB LSB

CY	AC	F1	RS1	RS0	OV	F0	P

Table 5. PSW register bit functions

Bit	Symbol	Function
PSW.7	CY	Carry flag
PSW.6	AC	Auxiliary Carry flag for BCD operations
PSW.5	F1	General purpose Flag 1 available for user
PSW.4	RS1	Register bank select control bit 1, used to select working register bank
PSW.3	RS0	Register bank select control bit 0, used to select working register bank
PSW.2	OV	Overflow flag
PSW.1	F0	General purpose Flag 0 available for user
PSW.0	P	Parity flag, affected by hardware to indicate odd / even number of “one” bits in the Accumulator, i.e. even parity.

Bits RS1and RS0 are used to select the working register bank as follows:

Table 6. Register Bank selection

RS1:RS0	Bank selected	Location
00	Bank 0	(00h – 07h)
01	Bank 1	(08h – 0Fh)
10	Bank 2	(10h – 17h)
11	Bank 3	(18h – 1Fh)

Stack Pointer Register (SP)

The Stack Pointer is a 1-byte register initialized to 07h after reset. This register is incremented before PUSH and CALL instructions, causing the stack to begin at location 08h.

Data Pointer Register (DPL and DPH)

The Data Pointer (DPTR) is 2 bytes wide. The lower byte is DPL and the higher DPH. It can be loaded as either a single 2 byte register:

MOV DPTR,#data16)

or as two individual, single byte registers:

MOV DPL,#data8

MOV DPH,#data8

It is generally used to access external code or data space, for example:

MOVC A,@A+DPTR or

MOVX A,@DPTR.

Internal Program Memory Sizing Register (ROMSIZE)

The content of this register is used to determine the size of internal Program memory space. The addressable space is defined as:

ROMSIZE x 256

The register, which can have minimum and maximum values of 00h and FFh respectively, can therefore be used to define an internal ROM space, that is multiples of 256 bytes, in the range 0 – 64KB.

By default, the reset value of the ROMSIZE register is 10h, which gives an internal Program memory space of 4KB.

The register is loaded under software control.

Power Control Register (PCON)

Table 7. PCON register flags

MSB							LSB
SMOD	F6	F5	F4	F3	F2	0	0

Table 8. PCON register bit functions

Bit	Symbol	Function
PCON.7	SMOD	Double baud rate bit. If Timer 1 is used to generate the baud rate and SMOD is set (1), the baud rate is doubled when the Serial Port is used in modes 1,2 or 3
PCON.6	F6	General purpose Flag 6 available for user
PCON.5	F5	General purpose Flag 5 available for user
PCON.4	F4	General purpose Flag 4 available for user
PCON.3	F3	General purpose Flag 3 available for user
PCON.2	F2	General purpose Flag 2 available for user
PCON.1	0	This bit is read only and is permanently cleared (0)
PCON.0	0	This bit is read only and is permanently cleared (0)

Hardware Description

The TSK51x core is partitioned into modules as shown in figure 12 and described below.

Core Engine

The core engine of the TSK51x is composed of four components:

- Control Unit
- Arithmetic Logic Unit
- Memory Control Unit
- RAM and SFR Control Unit.

The TSK51x engine allows instructions to be fetched from Program memory and to execute using either RAM or SFR.

Arithmetic Logic Unit:

- 8 bit arithmetic operations
- 8 bit logical operations
- Boolean manipulations
- 8 x 8 bit multiplication
- 8 / 8 bit division

RAM and SFR Control Unit:

- Can address up to 256 bytes of Read/Write Data memory space
- Serves as Interface for off-core Special Function Registers

Memory Control Unit:

- Can address up to 64KB of internal Program memory space
- Can address up to 64KB of external Program memory space
- Can address up to 64KB of external Data memory space.

Block Diagram

Figure 5 shows the core engine and peripheral units for the TSK51x. Note that interface signals PSWR, ROMDATAO and ROMWR in the Memory Control Unit are present only in the debug-enabled version of the core – TSK51A_D.

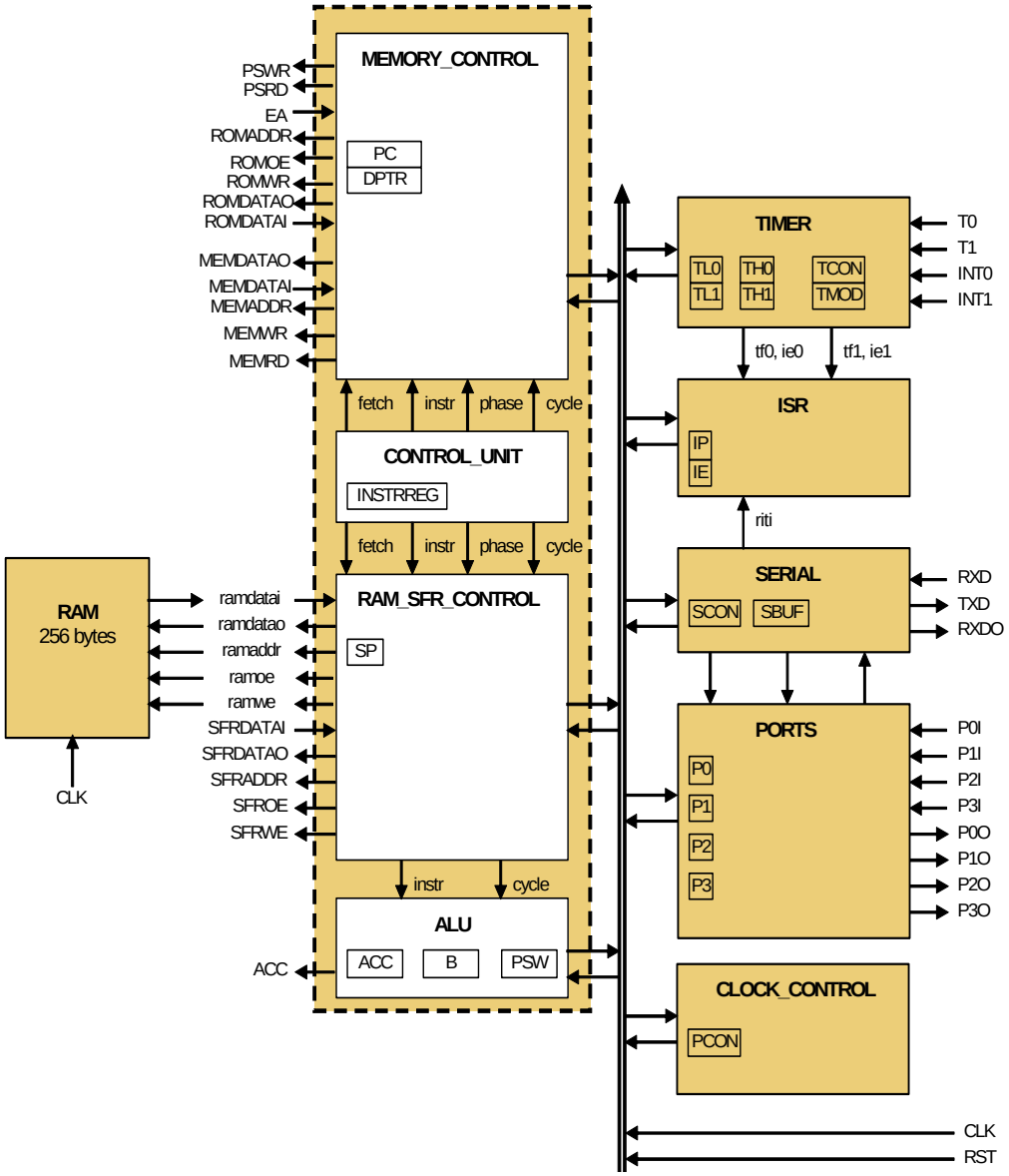


Figure 5. TSK51x Block diagram

Ports

Ports P0, P1, P2 and P3 are Special Function Registers. The contents of the SFR can be observed on the corresponding component symbol interface pins. Writing a ‘1’ to any of the ports causes the corresponding pin to be at the high level and writing a ‘0’ causes the corresponding pin to be held at the low level.

All four ports on the chip are bi-directional. Each of them consists of a Latch (SFR P0 to P3), an output drive and an input buffer, so the CPU can output or read data through any of these ports.

Timers / Counters

Timers 0 and 1

The TSK51x has two 16-bit timer/counter registers: Timer 0 and Timer 1. Both can be configured for counter or timer operations.

In timer mode, the register is incremented every machine (instruction) cycle, which means that it counts up after every 12 clock cycles.

In counter mode, the register is incremented when the falling edge is observed at the corresponding input pin T0 or T1. Since it takes two machine cycles to recognize a 1-to-0 event, the maximum input count rate is 1/24 of the external clock (CLK) frequency. There are no restrictions on the duty cycle, however to ensure proper recognition of 0 or 1 state, an input should be stable for at least one machine cycle (12 clock cycles).

Four operating modes can be selected for Timer 0 and Timer 1. Two Special Function Registers (TMOD and TCON) are used to select the appropriate mode.

Timer / Counter Mode Control Register(TMOD)

Table 9. The TMOD register flags

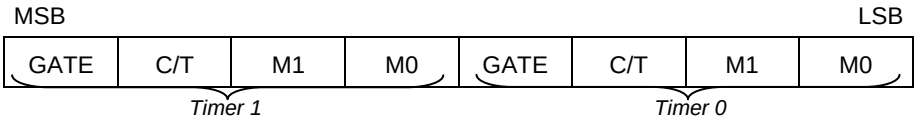


Table 10. The TMOD register bits description

Bit	Symbol	Function
TMOD.3 TMOD.7	GATE	When GATE = 0, Timer/Counter x will run only when TRx bit is set (see TCON register). This allows for Software Control. When GATE = 1, Timer/Counter x will run only when TRx bit is set (in TCON register) AND INTx pin is Low. This allows for Hardware Control.
TMOD.2 TMOD.6	C/T	When C/T = 0, Timer/Counter x will run as a timer, triggered by the internal clock. When C/T = 1, Timer/Counter x will run as a counter, triggered by the falling edge of the external signals entering pin T0 (for Timer/Counter 0) and T1 (for Timer/Counter 1).
TMOD.1 TMOD.5	M1	Selects mode for Timer/Counter 0 or Timer/Counter 1, as shown in Table 13.
TMOD.0 TMOD.4	M0	Selects mode for Timer/Counter 0 or Timer/Counter 1, as shown in Table 13.

Timer / Counter Control Register(TCON)

Table 11. The TCON register flags

MSB							LSB
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Table 12. The TCON register bit functions

Bit	Symbol	Function
TCON.7	TF1	Timer/Counter 1 overflow flag set by hardware when Timer/Counter 1 overflows. This flag is cleared by hardware.
TCON.6	TR1	Timer/Counter 1 Run control bit. If cleared, Timer/Counter 1 stops.
TCON.5	TF0	Timer/Counter 0 overflow flag set by hardware when Timer/Counter 0 overflows. This flag is cleared by hardware
TCON.4	TR0	Timer/Counter 0 Run control bit. If cleared, Timer/Counter 0 stops.
TCON.3	IE1	Interrupt 1 flag. Set by hardware when an interrupt of the type specified by IT1 is observed on external pin INT1. This flag is cleared when the interrupt is processed.
TCON.2	IT1	Interrupt 1 type control bit. Set/cleared by software to specify rising edge/high level triggered External Interrupt.
TCON.1	IE0	Interrupt 0 flag. Set by hardware when an interrupt of the type

		specified by IT0 is observed on external pin INT0. This flag is cleared when the interrupt is processed.
TCON.0	IT0	Interrupt 0 type control bit. Set/cleared by software to specify rising edge/high level triggered External Interrupt.

Timing Modes

Four modes of operation are supported for the two timers, determined by the state of bits M1 and M0 in the TMOD register (TMOD.1 and TMOD.0 respectively for Timer/Counter 0; TMOD.5 and TMOD.4 respectively for Timer/Counter 1). Table 13 summarizes the required states of these bits to achieve the desired operational mode.

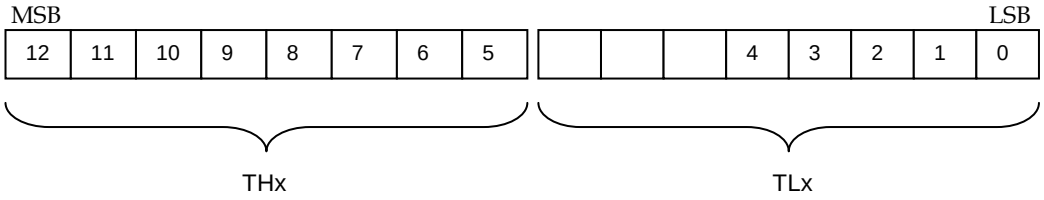
Table 13. Timer/Counter Modes

M1	M0	Mode
0	0	Mode 0
0	1	Mode 1
1	0	Mode 2
1	1	Mode 3

Mode 0

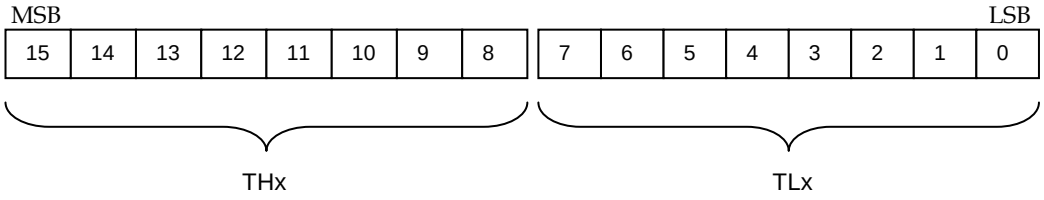
When in Mode 0, the Timer/Counter is set to 13 bits, where the 3 MSB bits of the TLx register are not used. Assuming the Timer/Counter is enabled, it will count from its set value (set by software) up to 1FFFh, at which point TFx is set to 1 to indicate overflow. Hardware then resets this value to 0.

At overflow the Timer/Counter rolls over to 0000h and continues to count up to 1FFFh, at which point TFx is set to 1 once again. This cycle continues until the Timer/Counter is disabled.



Mode 1

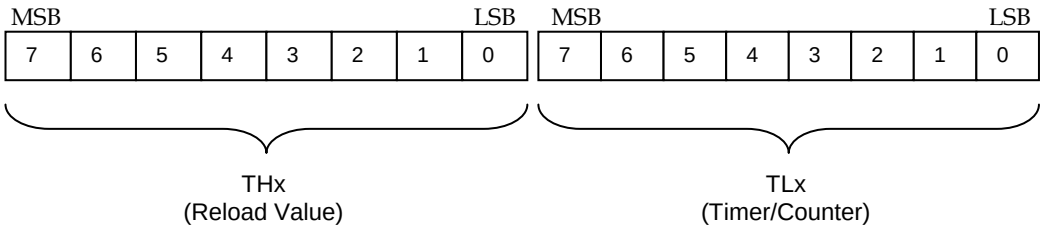
When in Mode 1, the Timer/Counter is set to 16 bits. The operation of the Timer/Counter in this mode is comparable to that in Mode 0. However, for this mode all 8 bits of the TLx register are used and therefore, the maximum value before overflow is FFFFh.



Mode 2

When in Mode 2, the Timer/Counter is set to 8 bits. This mode enables the Timer/Counter to be reloaded with its set value immediately after overflow. The two timing registers THx (the upper 8 bits) and TLx (the Lower 8 bits) are used differently.

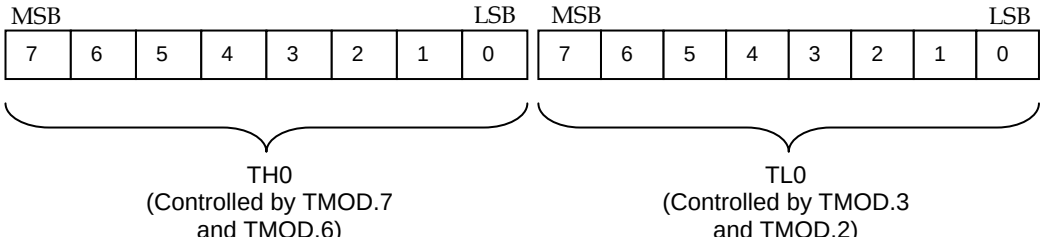
In this mode, THx holds the reload value, which is copied to TLx after overflow is detected, whereas TLx is the 8-bit dedicated Timer/Counter.



Mode 3

When Timer/Counter 1 is configured for operation in Mode 3, it is stopped. When Timer/Counter 0 is configured for operation in this same mode, the TH0 and TL0 registers operate independently of each other as follows:

- TL0 operates as an 8-bit Timer/Counter, controlled by Timer/Counter 0 mode control bits TMOD.3 and TMOD.2.
- TH0 operates as a dedicated 8-bit Timer, controlled by Timer/Counter 1 mode control bits TMOD.7 and TMOD.6, with no external gate control.



TL0, if enabled, will count from its set value to FFh, at which point the overflow flag for Timer/Counter 0 - TF0 (TCON.5) - is set to 1 and then reset to 0 by hardware. TL0 will continue to cycle through from 00h to FFh.

If Timer/Counter 1 is in Mode 3, then TH0, when enabled by Timer/Counter 1's mode control bits, will respond exactly the same as TL0, where TF1 (TCON.7) is set to 1 when overflow occurs. However, if Timer/Counter 1 is in Mode 0, 1 or 2, then the overflow flag, TF1, will be triggered by Timer/Counter 1 and TH0 of Timer/Counter 0.

Serial Interface

Serial Port 0

The serial buffer consists of two separate registers, a transmit buffer and a receive buffer. Writing data to the Special Function Register SBUF loads this data into the serial output buffer and starts the transmission. Reading from the SBUF register takes data from the serial receive buffer.

The serial port can simultaneously transmit and receive data. It can also buffer 1 byte at receive, which prevents the receive data from being lost if the CPU reads the first byte before transmission of the second byte is completed. The serial port can operate in 4 modes.

Mode 0

Pin RXD serves as input and RXDO as output. TXD outputs the shift clock. 8 bits are transmitted with LSB first. The baud rate is fixed at 1/12 of the external system clock frequency.

Mode 1

Pin RXD serves as input and TXD serves as serial output. No external shift clock is used. 10 bits are transmitted: a start bit (always 0), 8 data bits (LSB first), and a stop bit (always 1). On reception, the start bit synchronizes the transmission, 8 data bits are made available by reading SBUF, and the stop bit sets the flag RB8 in the Special Function Register SCON.

Mode 2

This mode is similar to Mode 1, with two differences. The baud rate is fixed at 1/32 or 1/64 of oscillator frequency, and 11 bits are transmitted or received: a start bit (0), 8 data bits (LSB first), a programmable 9^{th} bit, and a stop bit (1). The 9^{th} bit can be used to control the parity of the serial interface: at transmission, bit TB8 in SCON is output as the 9^{th} bit, and at receive, the 9^{th} bit affects RB8 in the Special Function Register SCON.

Mode 3

The only difference between Mode 2 and Mode 3 is that the baud rate is variable in Mode 3.

Reception is initialized in Mode 0 by setting the flags in SCON as follows: RI = 0 and REN = 1. In other modes, a start bit when REN = 1 starts receiving serial data

Multiprocessor Communication

The feature of receiving 9 bits in Modes 2 and 3 can be used for multiprocessor communication. In this case, the slave processors have bit SM2 in SCON set to 1. When the master processor outputs a slave's address, it sets the 9th bit to 1, causing a serial port receive interrupt in all the slaves. The slave processors compare the received byte with their network address. If there is a match, the addressed slave will clear SM2 and receive the rest of the message, while other slaves will leave the SM2 bit unaffected and ignore the message. After addressing the slave, the host will output the rest of the message with the 9th bit set to 0, so no serial port receive interrupt will be generated in unselected slaves.

Serial Port Control Register (SCON)

The function of the serial port depends on the status of the various flags in the Serial Port Control register SCON.

Table 14. The SCON register flags

MSB LSB

SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Table 15. The SCON register Bit functions

Bit	Symbol	Function
SCON.7	SM0	Sets baud rate
SCON.6	SM1	Sets baud rate
SCON.5	SM2	Enables multiprocessor communication feature.
SCON.4	REN	If set, enables serial reception. Cleared by software to disable reception.
SCON.3	TB8	The 9th transmitted data bit in Modes 2 and 3. Set or cleared by the CPU, depending on the function it performs (parity check, multiprocessor communication etc.)
SCON.2	RB8	In Modes 2 and 3, it is the 9th data bit received. In Mode 1, if SM2 is 0, RB8 is the stop bit. In Mode 0 this bit is not used. Must be cleared by software.
SCON.1	TI	Transmit interrupt flag, set by hardware after completion of a serial transfer. Must be cleared by software.
SCON.0	RI	Receive interrupt flag, set by hardware after completion of a serial reception. Must be cleared by software

Table 16. Serial Port Modes

SM0	SM1	Mode	Description	Baud Rate
0	0	0	Shift Register	F_{CLK} /12
0	1	1	8-bit UART	variable
1	0	2	9-bit UART	F_{CLK} /32 or /64
1	1	3	9-bit UART	variable

Table 17. Serial Port baud rates

Mode	Baud rate
Mode 0	F_{CLK} / 12
Mode 1,3	Timer 1 overflow rate
Mode 2	SMOD = 0 F_{CLK} / 64 SMOD = 1 F_{CLK} / 32

Note: SMOD is bit 7 in the Special Function Register PCON.

Generating Variable Baud Rate in Modes 1 and 3

In Modes 1 and 3, the Timer 1 overflow rate is used to generate baud rates. If Timer 1 is configured in auto – reload mode, to establish a baud rate the following equation is used:

$$\text{Baud Rate} = \frac{2^{SMOD} \times F_{CLK}}{32 \times 12 \times (256 - TH1)}$$

Reset

Hardware Reset (RST)

A reset is accomplished by holding the RST pin high for at least two instruction cycles (24 clock cycles) while the external clock (CLK) is running. The CPU responds by generating an internal reset, with the timing shown in Figure 6. The external reset signal is asynchronous to the internal clock. The RST pin is sampled on the rising edge, every 12 clock cycles.

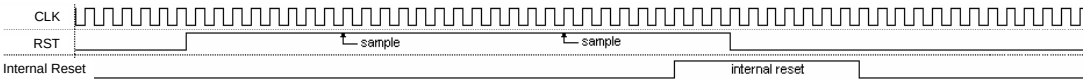


Figure 6. Reset timing

Note: CLK - clock oscillator input
RST - external reset input
Internal Reset - internal signal generated based on an external reset condition
sample - point at which the external reset input is sampled.

Reset Values

The internal reset signal is derived from the external reset (RST). It drives synchronous registers and flip-flops.

Table 18. Reset values

Register	Reset value
PC	0000h
ACC	00000000b
B	00000000b
XP	00000000b
PSW	00000000b
SP	00000000b
DPTR	0000h
P0	11111111b
P1	11111111b
P2	11111111b
P3	11111111b

Register	Reset value
IP	11100000b
IE	01100000b
TMOD	00000000b
TCON	00000000b
TH0	00000000b
TL0	00000000b
TH1	00000000b
TL1	00000000b
SCON	00000000b
SBUF	00000000b
ROMSIZE	00010000b
PCON	01111100b

Interrupts

The TSK51x provides five interrupt sources. There are two external interrupts accessible through pins INT0 and INT1, edge or level sensitive (rising edge or High level). There are, also, internal interrupts associated with Timer 0 and Timer 1 and an internal interrupt from the Serial Port.

External Interrupts

The choice between external interrupt level or transition activity is made by setting IT1 and IT0 bits in the Special Function Register TCON.

When the interrupt event happens, a corresponding Interrupt Control Bit is set (IE0 or IE1). This control bit triggers an interrupt if the appropriate interrupt bit is enabled.

When the interrupt service routine is vectored, the corresponding control bit (IE0 or IE1) is cleared provided that the edge triggered mode was selected. If level mode is active, the external requesting source controls flags IE0 or IE1 by the logic level on pins INT0 or INT1 (0 or 1).

Recognition of an interrupt event is possible if, during low to high transitions, both low and high levels last at least 1 machine (instruction) cycle (12 clock cycles).

Timer 0 and Timer 1 Interrupts

Timer 0 and 1 interrupts are generated by TF0 and TF1 flags, which are set by the rollover of Timer 0 and 1, respectively. When an interrupt is generated, the flag that caused this interrupt is cleared by the hardware, if the CPU accessed the corresponding interrupt service vector. This can be done only if this interrupt is enabled in the IE register.

Serial Port Interrupt

Serial Port interrupt is generated by logical OR of the flags TI and RI in the Special Function Register SCON. TI is set after completion of the transmit data. RI is set when the last bit of the incoming serial data was read. Neither RI nor TI is cleared by hardware, so the interrupt service routine must be responsible to clear these flags.

Interrupt Enable Register (IE)

Table 19. The IE register flags

MSB							LSB
EA	-	-	ES	ET1	EX1	ET0	EX0