

# ***R8820LV***

## **16-Bit RISC Microcontroller User's Manual**

**RDC** *RISC DSP Controller*

*RDC Semiconductor Co., Ltd*

<http://www.rdc.com.tw>

Tel. 886-3-583-2666

Fax 886-3-583-2688

**Contents**

|                                                                          | page |
|--------------------------------------------------------------------------|------|
| <b>1. Features</b>                                                       | 4    |
| <b>2. Block Diagram</b>                                                  | 4    |
| <b>3. Pin Configuration</b>                                              | 5    |
| <b>4. Pin Description</b>                                                | 8    |
| <b>5. Basic Application System Block &amp; Read/Write timing Diagram</b> | 14   |
| <b>6. Oscillator Characteristics</b>                                     | 17   |
| <b>7. Execution Unit</b>                                                 | 18   |
| 7.1 General Register                                                     | 18   |
| 7.2 Segment Register                                                     | 18   |
| 7.3 Instruction Pointer and Status Flags Register                        | 19   |
| 7.4 Address Generation                                                   | 20   |
| <b>8. Peripheral Control Block Register</b>                              | 21   |
| <b>9. System Clock Block</b>                                             | 23   |
| <b>10. Reset</b>                                                         | 24   |
| <b>11. Bus Interface Unit</b>                                            | 26   |
| 11.1 Memory and I/O Interface                                            | 26   |
| 11.2 Data Bus                                                            | 26   |
| 11.3 Wait States                                                         | 27   |
| 11.4 Bus Hold                                                            | 28   |
| 11.5 Bus Width                                                           | 30   |
| <b>12. Chip Select Unit</b>                                              | 31   |
| 12.1 $\overline{\text{UCS}}$                                             | 31   |
| 12.2 $\overline{\text{LCS}}$                                             | 32   |
| 12.3 $\overline{\text{MCSx}}$                                            | 33   |
| 12.4 $\overline{\text{PCSx}}$                                            | 34   |
| <b>13. Interrupt Controller Unit</b>                                     | 37   |
| 13.1 Master Mode and Slave Mode                                          | 37   |
| 13.2 Interrupt Vector, Type and Priority                                 | 38   |
| 13.3 Interrupt Request                                                   | 39   |
| 13.4 Interrupt Acknowledge                                               | 39   |
| 13.5 Programming Register                                                | 40   |

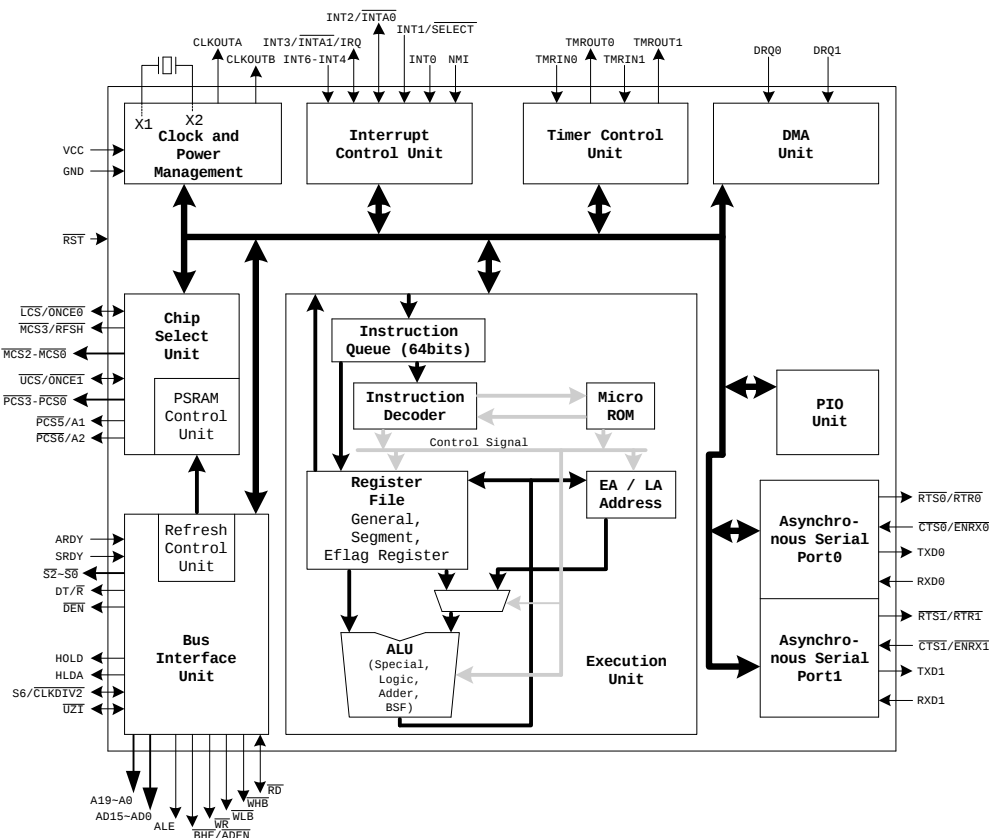
|                                                    |           |
|----------------------------------------------------|-----------|
| <b>14. DMA Unit</b>                                | <b>53</b> |
| 14.1 DMA Operation                                 | 53        |
| 14.2 External Request                              | 59        |
| 14.3 Serial Port / DMA Transfer                    | 61        |
| <b>15. Timer Control Unit</b>                      | <b>62</b> |
| 15.1 Timer/Counter Unit Output Mode                | 66        |
| <b>16. Watchdog Timer</b>                          | <b>68</b> |
| <b>17. Asynchronous Serial Port</b>                | <b>70</b> |
| 17.1 Serial Port Flow Control                      | 70        |
| 17.1.1 DCE/DTE Protocol                            | 70        |
| 17.1.2 CTS/RTR Protocol                            | 71        |
| 17.2 DMA Transfer to/form a serial port function   | 71        |
| 17.3 The Asynchronous Modes description            | 72        |
| <b>18. PIO Unit</b>                                | <b>77</b> |
| 18.1 PIO Multi-Function Pin list Table             | 77        |
| <b>19. PSRAM Control Unit</b>                      | <b>80</b> |
| <b>20. Instruction Set Opcodes and Clock Cycle</b> | <b>81</b> |
| 20.1 R8820LV Execution Timings                     | 85        |
| <b>21. DC Characteristics</b>                      | <b>86</b> |
| <b>22. AC Characteristics</b>                      | <b>88</b> |
| <b>23. Package Information</b>                     | <b>97</b> |

## 16-Bit Microcontroller with 16-bit external data bus

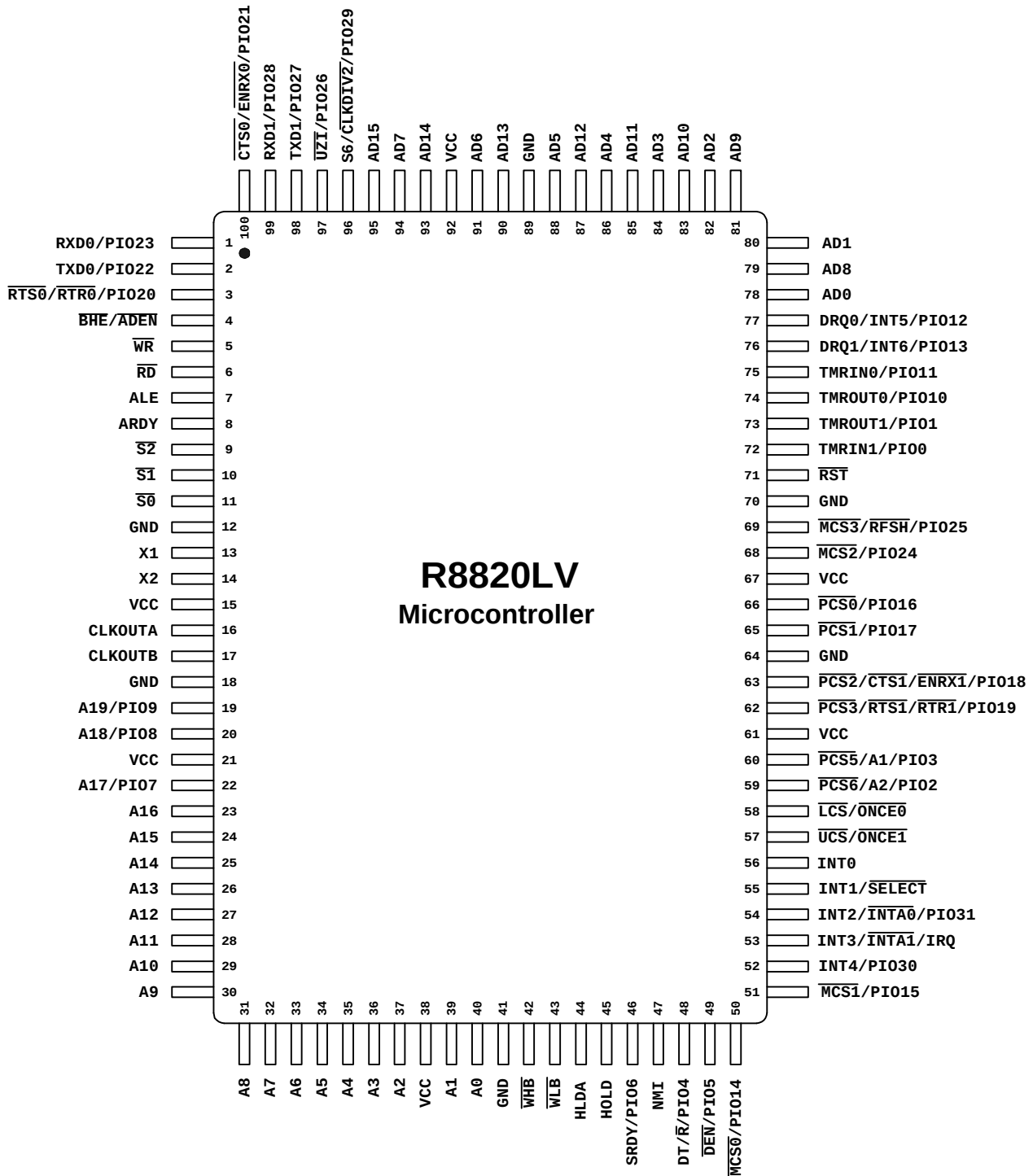
### 1. Features

- Five-stages pipeline
- RISC architecture
- Static Design & Synthesizable design
- Bus interface
  - Multiplexed address and Data bus which is compatible with 80C186 microprocessor
  - Supports nonmultiplexed address bus [A19 : A0]
  - 1M byte memory address space
  - 64K byte I/O space
- Software is compatible with the 80C186 microprocessor
- Support two Asynchronous serial channel with hardware handshaking signals.
- Supports 32 PIO pins
- PSRAM (Pseudo static RAM) interface with auto-refresh control
- Three independent 16-bit timers and one independent watchdog timer
- The Interrupt controller with seven maskable external interrupts and one nonmaskable external interrupt
- Two independent DMA channels
- Programmable chip-select logic for Memory or I/O bus cycle decoder
- Programmable wait-state generator
- Support serial port/ DMA transfers

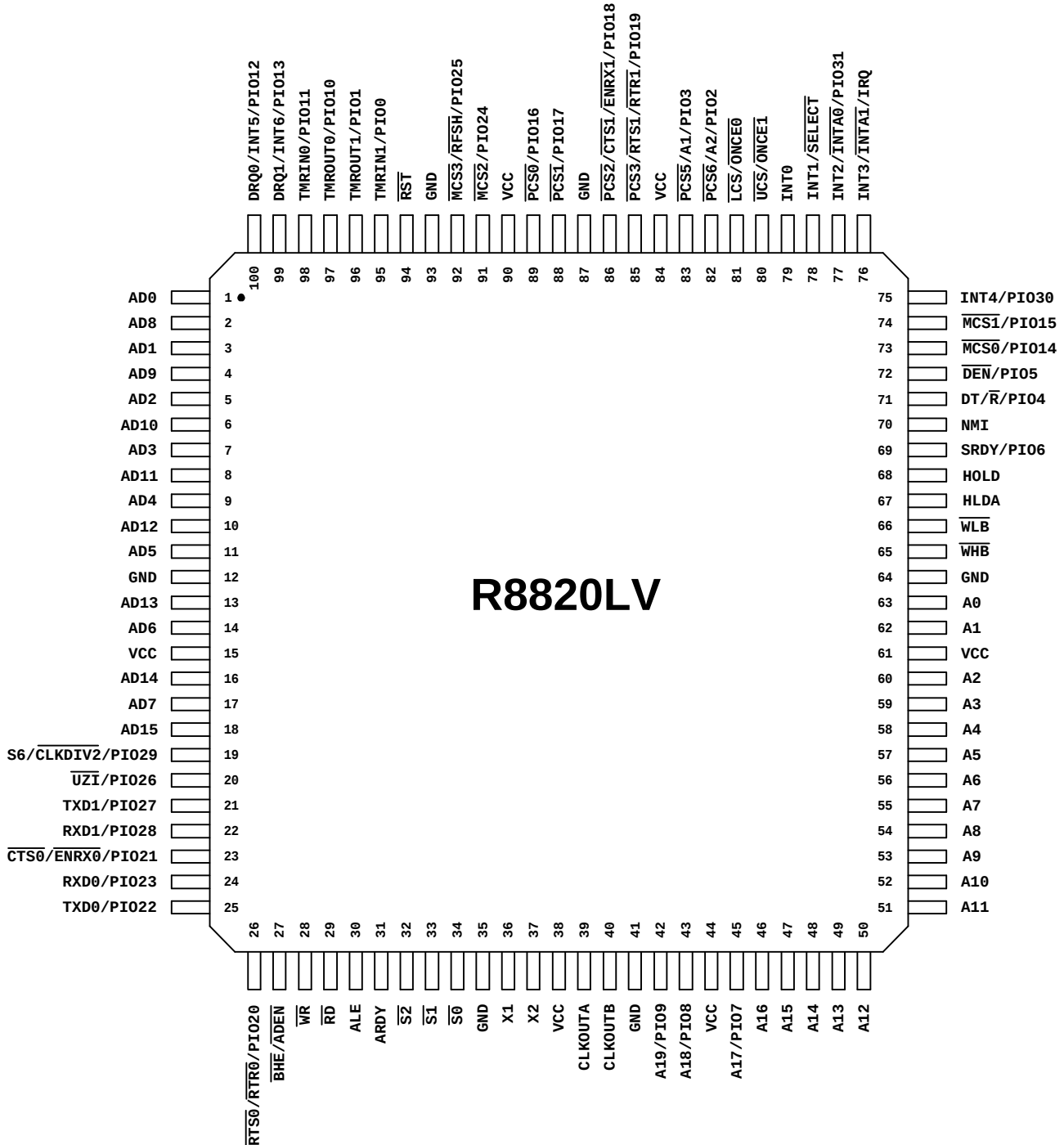
### 2. Block Diagram



### 3. Pin Configuration (PQFP)



(LQFP)



**R8820LV Pin OUT Table**

| Pin name                                      | LQFP Pin No. | PQFP Pin No. | Pin name                                                           | LQFP Pin No. | PQFP Pin No. |
|-----------------------------------------------|--------------|--------------|--------------------------------------------------------------------|--------------|--------------|
| AD0                                           | 1            | 78           | A11                                                                | 51           | 28           |
| AD8                                           | 2            | 79           | A10                                                                | 52           | 29           |
| AD1                                           | 3            | 80           | A9                                                                 | 53           | 30           |
| AD9                                           | 4            | 81           | A8                                                                 | 54           | 31           |
| AD2                                           | 5            | 82           | A7                                                                 | 55           | 32           |
| AD10                                          | 6            | 83           | A6                                                                 | 56           | 33           |
| AD3                                           | 7            | 84           | A5                                                                 | 57           | 34           |
| AD11                                          | 8            | 85           | A4                                                                 | 58           | 35           |
| AD4                                           | 9            | 86           | A3                                                                 | 59           | 36           |
| AD12                                          | 10           | 87           | A2                                                                 | 60           | 37           |
| AD5                                           | 11           | 88           | VCC                                                                | 61           | 38           |
| GND                                           | 12           | 89           | A1                                                                 | 62           | 39           |
| AD13                                          | 13           | 90           | A0                                                                 | 63           | 40           |
| AD6                                           | 14           | 91           | GND                                                                | 64           | 41           |
| VCC                                           | 15           | 92           | $\overline{WHB}$                                                   | 65           | 42           |
| AD14                                          | 16           | 93           | $\overline{WLB}$                                                   | 66           | 43           |
| AD7                                           | 17           | 94           | HLDA                                                               | 67           | 44           |
| AD15                                          | 18           | 95           | HOLD                                                               | 68           | 45           |
| S6/ $\overline{CLKDIV\ 2}$ /PI O29            | 19           | 96           | SRDY/PI O6                                                         | 69           | 46           |
| $\overline{UZI}$ /PI O26                      | 20           | 97           | NMI                                                                | 70           | 47           |
| TXD1/PI O27                                   | 21           | 98           | DT/ $\overline{R}$ /PI O4                                          | 71           | 48           |
| RXD1/PI O28                                   | 22           | 99           | $\overline{DEN}$ /PI O5                                            | 72           | 49           |
| $\overline{CTS0}$ / $\overline{ENRX0}$ /PIO21 | 23           | 100          | $\overline{MCS0}$ /PI O14                                          | 73           | 50           |
| RXD0/PI O23                                   | 24           | 1            | $\overline{MCS1}$ /PI O15                                          | 74           | 51           |
| TXD0/PI O22                                   | 25           | 2            | I NT4/PI O30                                                       | 75           | 52           |
| $\overline{RTS0}$ / $\overline{RTR0}$ /PIO20  | 26           | 3            | I NT3/ $\overline{INTA1}$ /I RQ                                    | 76           | 53           |
| $\overline{BHE}$ / $\overline{ADEN}$          | 27           | 4            | I NT2/ $\overline{INTA0}$ /PI O31                                  | 77           | 54           |
| $\overline{WR}$                               | 28           | 5            | I NT1/ $\overline{SELECT}$                                         | 78           | 55           |
| $\overline{RD}$                               | 29           | 6            | I NT0                                                              | 79           | 56           |
| ALE                                           | 30           | 7            | $\overline{UCS}$ / $\overline{ONCE1}$                              | 80           | 57           |
| ARDY                                          | 31           | 8            | $\overline{LCS}$ / $\overline{ONCE0}$                              | 81           | 58           |
| $\overline{S2}$                               | 32           | 9            | $\overline{PCS6}$ /A2/PI O2                                        | 82           | 59           |
| $\overline{S1}$                               | 33           | 10           | $\overline{PCS5}$ /A1/PI O3                                        | 83           | 60           |
| $\overline{S0}$                               | 34           | 11           | VCC                                                                | 84           | 31           |
| GND                                           | 35           | 12           | $\overline{PCS3}$ / $\overline{RTS1}$ / $\overline{RTR1}$ /PI O19  | 85           | 62           |
| X1                                            | 36           | 13           | $\overline{PCS2}$ / $\overline{CTS1}$ / $\overline{ENRX1}$ /PI O18 | 86           | 63           |
| X2                                            | 37           | 14           | GND                                                                | 87           | 64           |
| VCC                                           | 38           | 15           | $\overline{PCS1}$ /PI O17                                          | 88           | 65           |
| CLKOUTA                                       | 39           | 16           | $\overline{PCS0}$ /PI O16                                          | 89           | 66           |
| CLKOUTB                                       | 40           | 17           | VCC                                                                | 90           | 67           |
| GND                                           | 41           | 18           | $\overline{MCS2}$ /PI O24                                          | 91           | 68           |
| A19/PI O9                                     | 42           | 19           | $\overline{MCS3}$ / $\overline{RFSH}$ /PI O25                      | 92           | 69           |
| A18/PI O8                                     | 43           | 20           | GND                                                                | 93           | 70           |
| VCC                                           | 44           | 21           | $\overline{RST}$                                                   | 94           | 71           |
| A17/PI O7                                     | 45           | 22           | TMRI N1/PI O0                                                      | 95           | 72           |
| A16                                           | 46           | 23           | TMROUT1/PI O1                                                      | 96           | 73           |
| A15                                           | 47           | 24           | TMROUT0/PI O10                                                     | 97           | 74           |
| A14                                           | 48           | 25           | TMRI N0/PI O11                                                     | 98           | 75           |
| A13                                           | 49           | 26           | DRQ1/INT6/PI O13                                                   | 99           | 76           |
| A12                                           | 50           | 27           | DRQ0/INT5/PI O12                                                   | 100          | 77           |

#### 4. Pin Description

| Pin No.(PQFP)                             | Symbol                                                                                    | Type         | Description                                                                                                                                                                                                                                                                                                                                                                           |
|-------------------------------------------|-------------------------------------------------------------------------------------------|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15, 21, 38, 61, 67, 92                    | VCC                                                                                       | Input        | System power: +3.3 volt power supply.                                                                                                                                                                                                                                                                                                                                                 |
| 12, 18, 41, 64, 70, 89                    | GND                                                                                       | Input        | System ground.                                                                                                                                                                                                                                                                                                                                                                        |
| 71                                        | $\overline{\text{RST}}$                                                                   | Input*       | Reset input. When $\overline{\text{RST}}$ is asserted, the CPU immediately terminate all operation, clears the internal registers & logic, and the address transfers to the reset address FFFF0h.                                                                                                                                                                                     |
| 13                                        | X1                                                                                        | Input        | Input to the oscillator amplifier.                                                                                                                                                                                                                                                                                                                                                    |
| 14                                        | X2                                                                                        | Output       | Output from the inverting oscillator amplifier.                                                                                                                                                                                                                                                                                                                                       |
| 16                                        | CLKOUTA                                                                                   | Output       | Clock output A. The CLKOUTA operation is the same as crystal input frequency (X1). CLKOUTA remains active during reset and bus hold conditions.                                                                                                                                                                                                                                       |
| 17                                        | CLKOUTB                                                                                   | Output       | Clock output B. The CLKOUTB operation is the same as crystal input frequency (X1). CLKOUTB remains active during reset and bus hold conditions.                                                                                                                                                                                                                                       |
| <b>Asynchronous Serial Port Interface</b> |                                                                                           |              |                                                                                                                                                                                                                                                                                                                                                                                       |
| 1                                         | RXD0/PIO23                                                                                | Input/Output | Receive data for asynchronous serial port 0. This pin receives asynchronous serial data.                                                                                                                                                                                                                                                                                              |
| 2                                         | TXD0/PIO22                                                                                | Output/Input | Transmit data for asynchronous serial port 0. This pin transmits asynchronous serial data from the UART of the microcontroller.                                                                                                                                                                                                                                                       |
| 3                                         | $\overline{\text{RTS0}} / \overline{\text{RTR0}} / \text{PIO20}$                          | Output/Input | Ready to send/Ready to Receive signal for asynchronous serial port 0. When the $\overline{\text{RTS0}}$ bit in the AUXCON register is set and FC bit in the serial port 0 register is set the $\overline{\text{RTS0}}$ signal is enabled. Otherwise the $\overline{\text{RTS0}}$ bit is cleared and FC bit is set the $\overline{\text{RTR0}}$ signal is enabled.                     |
| 100                                       | $\overline{\text{CTS0}} / \overline{\text{ENRX0}} / \text{PIO21}$                         | Input/Output | Clear to send/Enable Receiver Request signal for asynchronous serial port 0. when $\overline{\text{ENRX0}}$ bit in the AUXCON register is cleared and the FC bit in the serial port 0 control register is set the $\overline{\text{ENRX0}}$ signal is enabled. Other when $\overline{\text{ENRX0}}$ bit is set and the FC bit is set the $\overline{\text{ENRX0}}$ signal is enabled. |
| 98                                        | TXD1/PIO27                                                                                | Output/Input | Transmit data for asynchronous serial port 1. This pin transmits asynchronous serial data from the UART of the microcontroller.                                                                                                                                                                                                                                                       |
| 99                                        | RXD1/PIO28                                                                                | Input/Output | Receive data for asynchronous serial port 1. This pin receives asynchronous serial data.                                                                                                                                                                                                                                                                                              |
| 62                                        | $\overline{\text{PCS3}} / \overline{\text{RTS1}} / \overline{\text{RTR1}} / \text{PIO18}$ | Output/Input | Ready to send/Ready to Receive signal for asynchronous serial port 1. When the $\overline{\text{RTS1}}$ bit in the AUXCON register is set and FC bit in the serial port 1 register is set the $\overline{\text{RTS1}}$ signal is enabled. Otherwise the $\overline{\text{RTS1}}$ bit is cleared and FC bit is set the $\overline{\text{RTR1}}$ signal is enabled.                     |

|               |                                                                                            |              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                        |                             |
|---------------|--------------------------------------------------------------------------------------------|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|-----------------------------|
| 63            | $\overline{\text{PCS2}} / \overline{\text{CTS1}} / \overline{\text{ENRX1}} / \text{PIO19}$ | Input/Output | Clear to send/Enable Receiver Request signal for asynchronous serial port 1. when $\overline{\text{ENRX0}}$ bit in the AUXCON register is cleared and the FC bit in the serial port 1 control register is set the $\overline{\text{ENRX1}}$ signal is enabled. Other when $\overline{\text{ENRX1}}$ bit is set and the FC bit is set the $\overline{\text{ENRX1}}$ signal is enabled.                                                                                                                                                                                  |                        |                             |
| Bus Interface |                                                                                            |              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                        |                             |
| 4             | $\overline{\text{BHE}} / \overline{\text{ADEN}}$                                           | Output/Input | Bus high enable/address enable. During a memory access, the $\overline{\text{BHE}}$ and (AD0 or A0) encodings indicate what type of the bus cycle. $\overline{\text{BHE}}$ is asserted during T1 and keeps the asserted to T3 and Tw. This pin is floating during bus hold and reset.                                                                                                                                                                                                                                                                                  |                        |                             |
|               |                                                                                            |              | $\overline{\text{BHE}}$ and (AD0 or A0) Encodings                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                        |                             |
|               |                                                                                            |              | $\overline{\text{BHE}}$                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | AD0 or A0              | Type of Bus Cycle           |
|               |                                                                                            |              | 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 0                      | Word transfer               |
|               |                                                                                            |              | 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 1                      | High byte transfer (D15-D8) |
|               |                                                                                            |              | 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 0                      | Low byte transfer (D7-D0)   |
| 1             | 1                                                                                          | Refresh      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                        |                             |
|               |                                                                                            |              | The address portion of the AD bus can be enabled or disabled by DA bit in the LMCS and UMCS register during LCS or UCS bus cycle access, if $\overline{\text{BHE}} / \overline{\text{ADEN}}$ is held high during power-on reset. The $\overline{\text{BHE}} / \overline{\text{ADEN}}$ with a internal weak pull-up register, so no external pull-up register is required. The AD bus always drives both address and data during LCS or UCS bus cycle access, if the $\overline{\text{BHE}} / \overline{\text{ADEN}}$ pin with external pull-low resister during reset. |                        |                             |
| 5             | $\overline{\text{WR}}$                                                                     | Output       | Write strobe. This pin indicates that the data on the bus is to be written into a memory or an I/O device. $\overline{\text{WR}}$ is active during T2, T3 and Tw of any write cycle, floats during a bus hold or reset.                                                                                                                                                                                                                                                                                                                                                |                        |                             |
| 6             | $\overline{\text{RD}}$                                                                     | Output       | Read Strobe. Active low signal which indicates that the microcontroller is performing a memory or I/O read cycle. $\overline{\text{RD}}$ floats during bus hold or reset.                                                                                                                                                                                                                                                                                                                                                                                              |                        |                             |
| 7             | ALE                                                                                        | Output       | Address latch enable. Active high. This pin indicates that an address output on the AD bus. Address is guaranteed to be valid on the trailing edge of ALE. This pin is tri-stated during ONCE mode and is never floating during a bus hold or reset.                                                                                                                                                                                                                                                                                                                   |                        |                             |
| 8             | ARDY                                                                                       | Input        | Asynchronous ready. This pin performs the microcontroller that the address memory space or I/O device will complete a data transfer. The ARDY pin accepts a rising edge that is asynchronous to CLKOUTA and is active high. The falling edge of ARDY must be synchronized to CLKOUTA. Tie ARDY high, the microcontroller is always asserted in the ready condition. If the ARDY is not used, tie this pin low to yield control to SRDY.                                                                                                                                |                        |                             |
| 9<br>10<br>11 | $\overline{\text{S2}}$<br>$\overline{\text{S1}}$<br>$\overline{\text{S0}}$                 | Output       | Bus cycle status. These pins are encoded to indicate the bus status. $\overline{\text{S2}}$ can be used as memory or I/O indicator. $\overline{\text{S1}}$ can be used as DT/ $\overline{\text{R}}$ indicator. These pins are floating during hold and reset.                                                                                                                                                                                                                                                                                                          |                        |                             |
|               |                                                                                            |              | Bus Cycle Encoding Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                        |                             |
|               |                                                                                            |              | $\overline{\text{S2}}$                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | $\overline{\text{S1}}$ | $\overline{\text{S0}}$      |

|                                                                  |                                                      |              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |   |   |                       |
|------------------------------------------------------------------|------------------------------------------------------|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|---|-----------------------|
|                                                                  |                                                      |              | 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 0 | 0 | Interrupt acknowledge |
|                                                                  |                                                      |              | 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 0 | 1 | Read data from I/O    |
|                                                                  |                                                      |              | 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 1 | 0 | Write data to I/O     |
|                                                                  |                                                      |              | 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 1 | 1 | Halt                  |
|                                                                  |                                                      |              | 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 0 | 0 | Instruction fetch     |
|                                                                  |                                                      |              | 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 0 | 1 | Read data from memory |
|                                                                  |                                                      |              | 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 1 | 0 | Write data to memory  |
|                                                                  |                                                      |              | 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 1 | 1 | Passive               |
| 19<br>20<br>22<br>23-37<br>39, 40                                | A19/PIO9<br>A18/PIO8<br>A17/PIO7<br>A16-A2<br>A1, A0 | Output/Input | Address bus. Non-multiplex memory or I/O address. The A bus is one-half of a CLKOUTA period earlier than the AD bus. These pins are high-impedance during bus hold or reset.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |   |   |                       |
| 78,80,82,84,8<br>6,88<br>91,94<br>79,81,83,85,8<br>7,90<br>93,95 | AD0-AD7<br>AD8-AD15                                  | Input/Output | The multiplexed address and data bus for memory or I/O accessing. The address is present during the t1 clock phase, and the data bus phase is in t2-t4 cycle. The address phase of the AD bus can be disabled when the BHE / ADEN pin with external pull-Low resister during reset. The AD bus is in high-impedance state during bus hold or reset condition and this bus also be used to load system configuration information (with pull-up or pull-Low resister) into the RESCON register when the reset input from low go high.                                                                                                                                                                                                                                                                                                      |   |   |                       |
| 42                                                               | $\overline{\text{WHB}}$                              | Output       | Write high byte. This pin indicates the high byte data (AD15-AD8) on the bus is to be written to a memory or I/O device. $\overline{\text{WHB}}$ is the logic OR of $\overline{\text{BHE}}$ , $\overline{\text{WR}}$ and AD0 inverting. This pin is floating during reset or bus hold.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |   |   |                       |
| 43                                                               | $\overline{\text{WLB}}$                              | Output       | Write low byte. This pin indicates the low byte data (AD7-AD0) on the bus is to be written to a memory or I/O device. $\overline{\text{WLB}}$ is the logic OR of $\overline{\text{BHE}}$ , $\overline{\text{WR}}$ and AD0. This pin is floating during reset or bus hold.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |   |   |                       |
| 44                                                               | HLDA                                                 | Output       | Bus hold acknowledge. Active high. The microcontroller will issue a HLDA in response to a HOLD request by external bus master at the end of T4 or Ti. When the microcontroller is in hold status (HLDA is high), the AD15-AD0, A19-A0, $\overline{\text{WR}}$ , $\overline{\text{RD}}$ , $\overline{\text{DEN}}$ , $\overline{\text{S0}} - \overline{\text{S1}}$ , $\overline{\text{S6}}$ , $\overline{\text{BHE}}$ , $\overline{\text{DT/R}}$ , $\overline{\text{WHB}}$ and $\overline{\text{WLB}}$ are floating, and the $\overline{\text{UCS}}$ , $\overline{\text{LCS}}$ , $\overline{\text{PCS6}} - \overline{\text{PCS5}}$ , $\overline{\text{MCS3}} - \overline{\text{MCS0}}$ and $\overline{\text{PCS3}} - \overline{\text{PCS0}}$ will be drive high. After HOLD is detected as being low, the microcontroller will lower HLDA. |   |   |                       |
| 45                                                               | HOLD                                                 | Input        | Bus Hold request. Active high. This pin indicates that another bus master is requesting the local bus.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |   |   |                       |
| 46                                                               | SRDY/PIO6                                            | Input/Output | Synchronous ready. This pin performs the microcontroller that the address memory space or I/O device will complete a data transfer. The SRDY pin accepts a falling edge that is asynchronous to CLKOUTA and is active high. SRDY is accomplished by elimination of the one-half clock period required to internally synchronize ARDY. Tie SRDY high the microcontroller is always assert in the ready condition. If the SRDY is not used, tie this pin low to yield control to ARDY.                                                                                                                                                                                                                                                                                                                                                     |   |   |                       |
| 48                                                               | DT/ $\overline{\text{R}}$ /PIO4                      | Output/Input | Data transmit or receive. This pin indicates the direction of data flow through an external data-bus transceiver. DT/ $\overline{\text{R}}$ low, the microcontroller receives data. When DT/R is asserted high, the microcontroller writes data to the data bus.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |   |   |                       |

|                                   |                                                                                                                                                                     |              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 49                                | $\overline{\text{DEN}}$ /PIO5                                                                                                                                       | Output/Input | Data enable. This pin is provided as a data bus transceiver output enable. $\overline{\text{DEN}}$ is asserted during memory and I/O access. $\overline{\text{DEN}}$ is driven high when DT/ $\overline{\text{R}}$ changes state. It is floating during bus hold or reset condition.                                                                                                                                                                                                                                                                                                                                                                                                                |
| 96                                | S6/ $\overline{\text{CLKDIV}}$ /PIO29                                                                                                                               | Output/Input | Bus cycle status bit6/clock divided by 2. For S6 feature, this pin is low to indicate a microcontroller-initiated bus cycle or high to indicate a DMA-initiated bus cycle during T2, T3, Tw and T4. For CLKDIV2 feature. The internal clock of microcontroller is the external clock be divided by 2. (CLKOUTA, CLKOUTB=X1/2), if this pin held low during power-on reset. The pin is sampled on the rising edge of $\overline{\text{RST}}$ .                                                                                                                                                                                                                                                       |
| 97                                | $\overline{\text{UZI}}$ /PIO26                                                                                                                                      | Output/Input | Upper zero indicate. This pin is the logical OR of the inverted A19-A16. It asserts in the T1 and is held throughout the cycle.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>Chip Select Unit Interface</b> |                                                                                                                                                                     |              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 50<br>51<br>68<br>69              | $\overline{\text{MCS0}}$ /PIO14<br>$\overline{\text{MCS1}}$ /PIO15<br>$\overline{\text{MCS2}}$ /PIO24<br>$\overline{\text{MCS3}}$ / $\overline{\text{RFSH}}$ /PIO25 | Output/Input | Midrange memory chip selects. For $\overline{\text{MCS}}$ feature, these pins are active low when enable the MMCS register to access a memory. The address ranges are programmable. $\overline{\text{MCS3}} - \overline{\text{MCS0}}$ are held high during bus hold. When programming LMCS register, pin69 is as a $\overline{\text{RFSH}}$ pin to auto refresh the PSAM.                                                                                                                                                                                                                                                                                                                           |
| 57                                | $\overline{\text{UCS}}$ / $\overline{\text{ONCE1}}$                                                                                                                 | Output/Input | Upper memory chip select/ONCE mode request 1. For $\overline{\text{UCS}}$ feature, this pin acts low when system accesses the defined portion memory block of the upper 512K bytes (80000h-FFFFFh) memory region. $\overline{\text{UCS}}$ default acted address region is from F0000h to FFFFFh after power-on reset. The address range acting $\overline{\text{UCS}}$ is programmed by software. For $\overline{\text{ONCE1}}$ feature. If $\overline{\text{ONCE0}}$ and $\overline{\text{ONCE1}}$ are sampled low on the rising edge of $\overline{\text{RST}}$ . The microcontroller enters ONCE mode. In ONCE mode, all pins are high-impedance. This pin incorporates weakly pull-up resistor. |
| 58                                | $\overline{\text{LCS}}$ / $\overline{\text{ONCE0}}$                                                                                                                 | Output/Input | Lower memory chip select/ONCE mode request 0. For $\overline{\text{LCS}}$ feature, this pin acts low when the microcontroller accesses the defined portion memory block of the lower 512K (00000h-7FFFFh) memory region. The address range acting $\overline{\text{LCS}}$ is programmed by software.<br>For $\overline{\text{ONCE0}}$ feature, see $\overline{\text{UCS}}$ / $\overline{\text{ONCE1}}$ description. This pin incorporates weakly pull-up register.                                                                                                                                                                                                                                  |
| 59<br>60                          | $\overline{\text{PCS6}}$ /A2/PIO2<br>$\overline{\text{PCS5}}$ /A1/PIO3                                                                                              | Output/Input | Peripheral chip selects/latched address bit. For $\overline{\text{PCS}}$ feature, these pins act low when the microcontroller accesses the fifth or sixth region of the peripheral memory (I/O or memory space). The base address of $\overline{\text{PCS}}$ is programmable. These pins assert with the AD address bus and are not float during bus hold.<br>For latched address bit feature. These pins output the latched address A2, A1 when cleared the EX bit in the $\overline{\text{MCS}}$ and $\overline{\text{PCS}}$ auxiliary register. The A2, A1 retains previous latched data during bus hold.                                                                                        |
| 62<br>63                          | $\overline{\text{PCS3}}$ / $\overline{\text{RTS1}}$ / $\overline{\text{RTR1}}$ /PIO19                                                                               | Output/Input | Peripheral chip selects. These pins act low when the microcontroller accesses the defined memory area of the                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

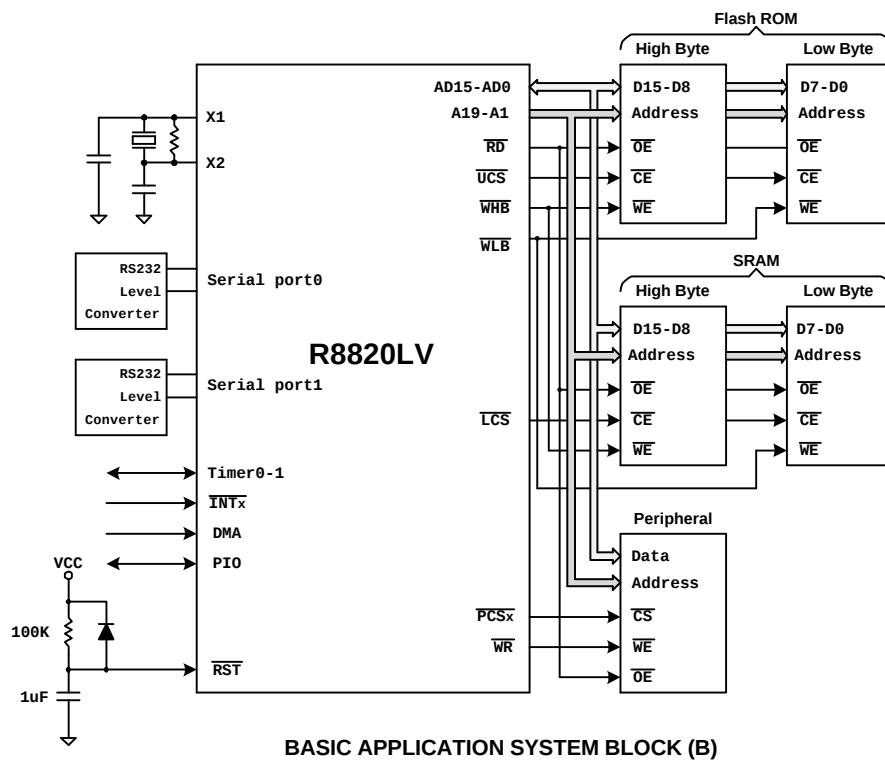
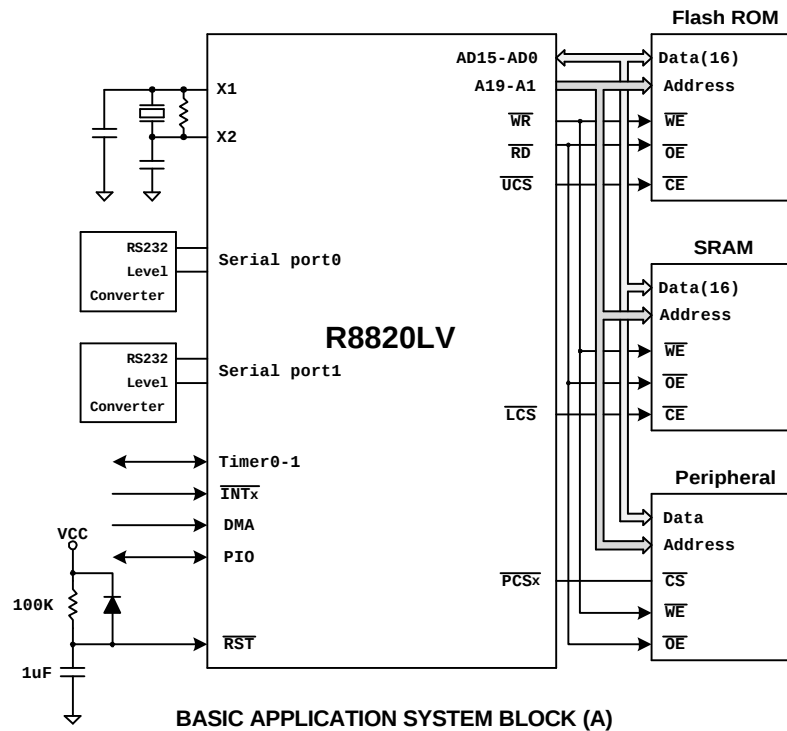
|                                         |                                                                                                                                                                                  |              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 65<br>66                                | $\overline{\text{PCS2}} / \overline{\text{STS1}} / \overline{\text{ENRX1}} / \text{PIO18}$<br>$\overline{\text{PCS1}} / \text{PIO17}$<br>$\overline{\text{PCS0}} / \text{PIO16}$ |              | peripheral memory block (I/O or memory address). For I/O accessed, the base address can be programmed in the region 00000h to 0FFFFh. For memory address access, the base address can be located in the 1M byte memory address region. These pins assert with the multiplexed AD address bus and are not float during bus hold.                                                                                                                                                                                          |
| <b>Interrupt Control Unit Interface</b> |                                                                                                                                                                                  |              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 47                                      | NMI                                                                                                                                                                              | Input        | Nonmaskable Interrupt. The NMI is the highest priority hardware interrupt and is nonmaskable. When this pin is asserted (NMI transition from low to high), the microcontroller always transfers the address bus to the location specified by the nonmaskable interrupt vector in the microcontroller interrupt vector table. The NMI pin must be asserted for at least one CLKOUTA period to guarantee that the interrupt is recognized.                                                                                 |
| 52                                      | INT4/PIO30                                                                                                                                                                       | Input/Output | Maskable interrupt request 4. Act high. This pin indicates that an interrupt request has occurred. The microcontroller will jump to the INT4 address vector to execute the service routine if the INT4 is enable. The interrupt input can be configured to be either edge- or level-triggered. The requesting device must holt the INT4 until the request is acknowledged to guarantee interrupt recognition.                                                                                                            |
| 53                                      | INT3/ $\overline{\text{INTA1}}$ /IRQ                                                                                                                                             | Input/Output | Maskable interrupt request 3/interrupt acknowledge 1/slave interrupt request. For INT3 feature, except the difference interrupt line and interrupt address vector, the function of INT3 is the same as INT4. For $\overline{\text{INTA1}}$ feature, in cascade mode or special fully-nested mode, this pin corresponds the INT1. For IRQ feature, when the microcontroller is as a slave device, this pin issues an interrupt request to the master interrupt controller.                                                |
| 54                                      | INT2/ $\overline{\text{INTA0}}$ /PIO31                                                                                                                                           | Input/Output | Maskable interrupt request 2/interrupt acknowledge 0. For INT2 feature, except the difference interrupt line and interrupt address vector, the function of INT2 is the same as INT4. For $\overline{\text{INTA0}}$ feature, in cascade mode or special fully-nested mode, this pin corresponds the INT0.                                                                                                                                                                                                                 |
| 55                                      | INT1/ $\overline{\text{SELECT}}$                                                                                                                                                 | Input/Output | Maskable interrupt request 1/slave select. For INT1 feature, except the difference interrupt line and interrupt address vector, the function of INT1 is the same as INT4. For $\overline{\text{SELECT}}$ feature, when the microcontroller is as a slave device, this pin is driven from the master interrupt controller decoding. This pin acts to indicate that an interrupt appears on the address and data bus. The INT0 must act before $\overline{\text{SELECT}}$ acts when the interrupt type appears on the bus. |
| 56                                      | INT0 / SO1                                                                                                                                                                       | Input/Output | Maskable interrupt request 0. Except the interrupt line and interrupt address vector, the function of INT0 is the same as INT4.                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>Timer Control Unit Interface</b>     |                                                                                                                                                                                  |              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 72<br>75                                | TMRIN1/PIO0<br>TMRIN0/PIO11                                                                                                                                                      | Input/Output | Timer input. These pins can be as clock or control signal input, which depend upon the programmed timer mode. After internally synchronizing low to high transitions on TMRIN, the timer controller increments. These pins must be pull-up if not being used.                                                                                                                                                                                                                                                            |
| 73<br>74                                | TMROUT1/PIO1<br>TMROUT0/PIO10                                                                                                                                                    | Output/Input | Timer output. Depending on timer mode select these pins provide single pulse or continuous waveform. The duty cycle                                                                                                                                                                                                                                                                                                                                                                                                      |

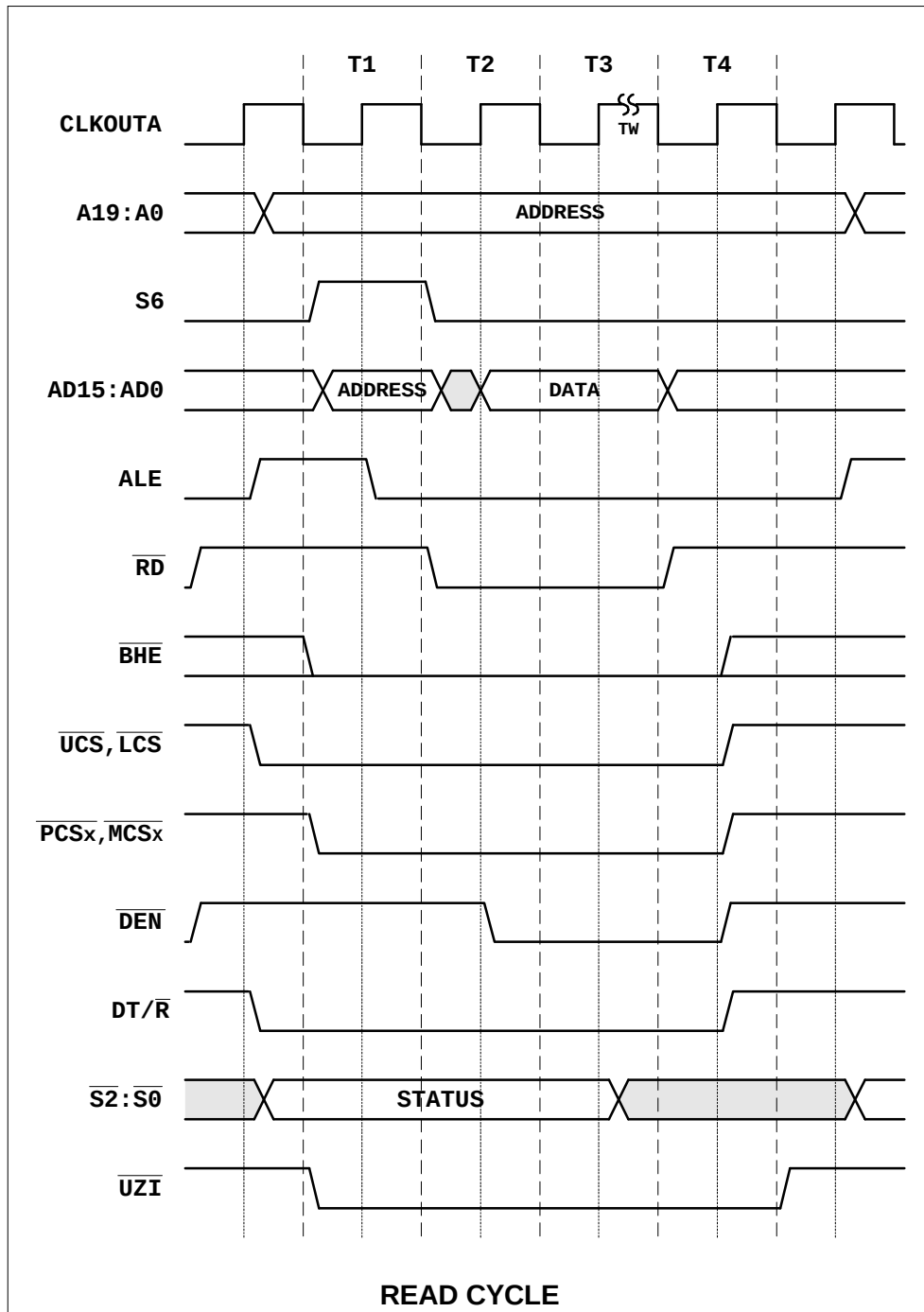
|                           |                                    |              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|---------------------------|------------------------------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                           |                                    |              | of the waveform can be programmable. These pins are floated during a bus hold or reset.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>DMA Unit Interface</b> |                                    |              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| 76<br>77                  | DRQ1/INT6/PIO13<br>DRQ0/INT5/PIO12 | Input/Output | DMA request. These pins are asserted high by an external device when the device is ready for DMA channel 1 or channel 0 to perform a transfer. These pins are level-triggered and internally synchronized. The DRQ signals must remain act until finish serviced and are not latched.<br>For INT6/INT5 function: When the DMA function is not being used, INT6/INT5 can be used as an additional external interrupt request. And they share the corresponding interrupt type and register control bits. The INT6/5 are edge-triggered only and must be hold until the interrupt is acknowledged. |

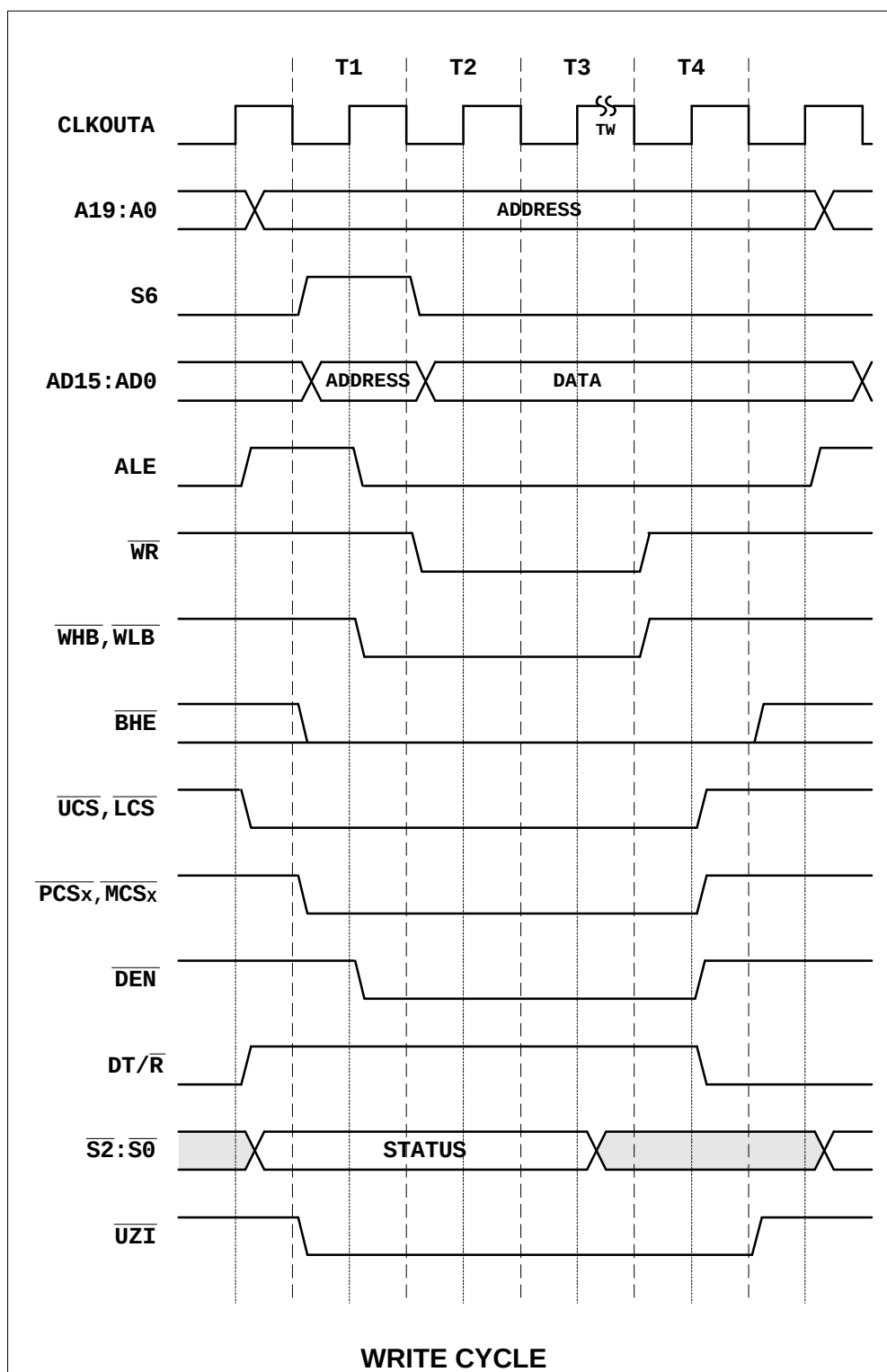
**Notes:**

1. When enable the PIO Data register, there are 32 MUX definition pins can be as a PIO pin. For example, the DRD1/PIO13 (pin76) can be as a PIO13 when enable the PIO Data register.
2. The PIO status during Power-On reset : PIO1, PIO10, PIO22, PIO23 are input with pull-down, PIO4 to PIO9 are normal operation and the others are input with pull-up.

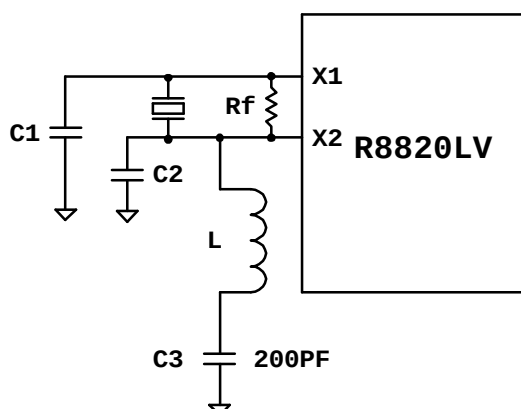
## 5. Basic Application System Block & Read/Write timing Diagram







## 6. Oscillator Characteristics & System Clock



### For fundamental-mode crystal:

C1 ----- 20pF  $\pm$  20%  
 C2 ----- 20pF  $\pm$  20%  
 Rf ----- 1 mega-ohm  
 C3 ----- Don't care  
 L ----- Don't care

### For third-overtone mode crystal:

C1 ----- 20pF  $\pm$  20%  
 C2 ----- 20pF  $\pm$  20%  
 C3 ----- 200pf  
 Rf ----- 1 mega-ohm  
 L ----- 3.0uH  $\pm$  20% (40MHz)  
               4.7uH  $\pm$  20% (33MHz)  
               8.2uH  $\pm$  20% (25MHz)  
               12uH  $\pm$  20% (20MHZ)

## 7. Execution Unit

### 7.1 General Register

The R8820LV has eight 16-bit general registers. And the AX,BX,CX,DX can be subdivided into two 8-bit register (AH,AL,BH,

BL,CH,CL,DH,DL). The functions of these registers are described as follows.

**AX** : Word Divide , Word Multiply, Word I/O operation.

**AH** : Byte Divide , Byte Multiply, Byte I/O , Decimal Arithmetic, Translate operation.

**AL** : Byte Divide , Byte Multiply operation.

**BX** : Translate operation.

**CX** : Loops, String operation

**CL** : Variable Shift and Rotate operation.

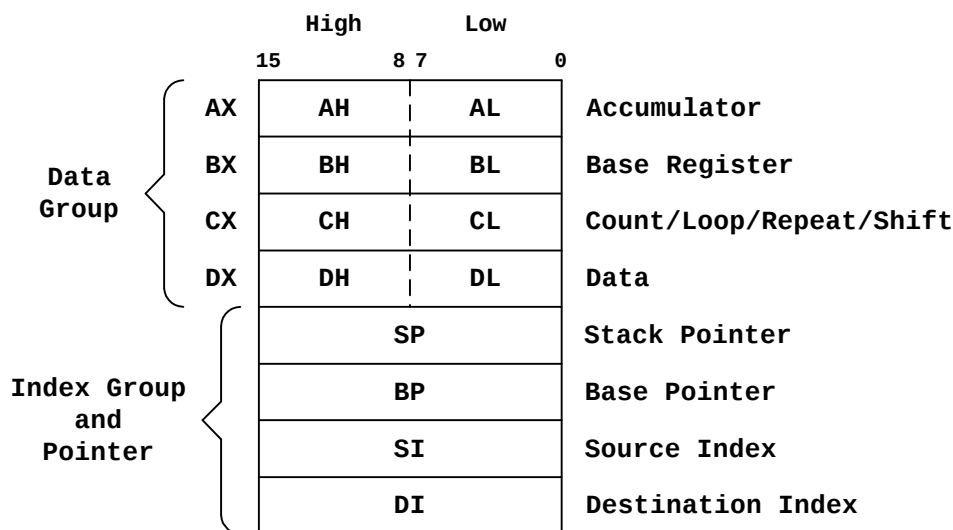
**DX** : Word Divide , Word Multiply, Indirect I/O operation

**SP** : Stack operations (POP, POPA, POPF, PUSH, PUSHA, PUSHF)

**BP** : General-purpose register which can be used to determine offset address of operands in Memory.

**SI** : String operations

**DI** : String operations



### GENERAL REGISTERS

### 7.2 Segment Register

R8820LV has four 16-bit segment registers, CS, DS, SS, ES. The segment registers contain the base addresses (starting location) of these memory segments, and they are immediately addressable for code (CS), data (DS & ES), and stack (SS) memory.

**CS (Code Segment)** : The CS register points to the current code segment, which contains instruction to be fetched. The default location memory space for all instruction is 64K. The initial value of CS register is 0FFFFh.

**DS (Data Segment)** : The DS register points to the current data segment, which generally contains program variables. The DS register initialize to 0000H.

**SS (Stack Segment)** : The SS register points to the current stack segment, which is for all stack operations, such as pushes and pops. The stack segment is used for temporary space. The SS register initialize to 0000H.

**ES (Extra Segment)** : The ES register points to the current extra segment which is typically for data storage, such as large string operations and large data structures. The DS register initialize to 0000H.

|    |   |   |   |                      |
|----|---|---|---|----------------------|
| 15 | 8 | 7 | 0 |                      |
|    |   |   |   | <b>CS</b>            |
|    |   |   |   | <b>Code Segment</b>  |
|    |   |   |   | <b>DS</b>            |
|    |   |   |   | <b>Data Segment</b>  |
|    |   |   |   | <b>SS</b>            |
|    |   |   |   | <b>Stack Segment</b> |
|    |   |   |   | <b>ES</b>            |
|    |   |   |   | <b>Extra Segment</b> |

### SEGMENT REGISTERS

## 7.3 Instruction Pointer and Status Flags Register

**IP (Instruction Pointer)** : The IP is a 16-bit register and it contains the offset of the next instruction to be fetched. Software can not to direct access the IP register and this register is updated by the Bus Interface Unit. It can change, be saved or be restored as a result of program execution. The IP register initialize to 0000H and the CS:IP starting execution address is at 0FFFF0H.

### Processor Status Flags Registers

**FLAGS**  
Reset Value : 0000h

|          |    |    |    |    |    |    |    |    |    |     |    |     |    |     |    |
|----------|----|----|----|----|----|----|----|----|----|-----|----|-----|----|-----|----|
| 15       | 14 | 13 | 12 | 11 | 10 | 9  | 8  | 7  | 6  | 5   | 4  | 3   | 2  | 1   | 0  |
| Reserved |    |    |    | OF | DF | IF | TF | SF | ZF | Res | AF | Res | PF | Res | CF |

These flags reflect the status after the Execution Unit is executed.

**Bit 15-12** : Reserved

**Bit 11: OF**, Overflow Flag. An arithmetic overflow has occurred, this flag will be set.

**Bit 10 : DF**, Direction Flag. If this flag is set, the string instructions are increment address process. If DF is cleared, the string instructions are decrement address process. Refer the STD and CLD instructions for how to set and clear the DF flag.

**Bit 9 : IF**, Interrupt-Enable Flag. Refer the STI and CLI instructions for how to set and clear the IF flag.

Set to 1 : The CPU enables the maskable interrupt request.

Set to 0 : The CPU disables the maskable interrupt request.

**Bit 8: TF**, Trace Flag. Set to enable single-step mode for debugging; Clear to disable the single-step mode. If an application program sets the TF flag using POPF or IRET instruction, a debug exception is generated after the instruction (The CPU automatically generates an interrupt after each instruction) that follows the POPF or IRET instruction.

**Bit 7: SF**, Sign Flag. If this flag is set, the high-order bit of the result of an operation is 1, indicating it is negative.

**Bit 6: ZF**, Zero Flag. The result of operation is zero, this flag is set.

**Bit 5:** Reserved

**Bit 4: AF**, Auxiliary Flag. If this flag is set, there has been a carry from the low nibble to the high or a borrow from the high nibble to the low nibble of the AL general-purpose register. Used in BCD operation.

**Bit 3:** Reserved.

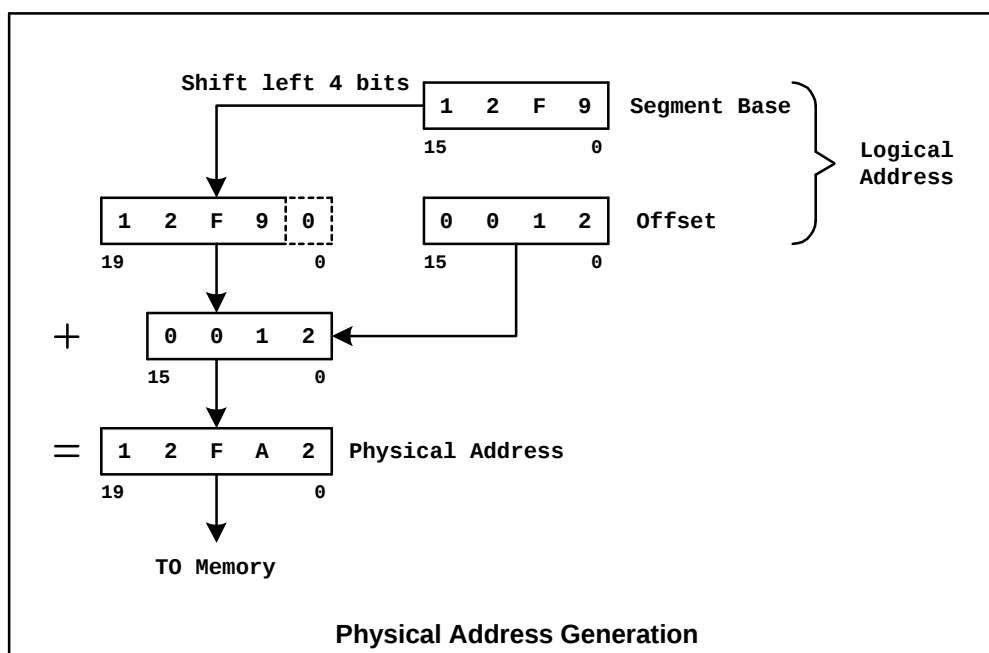
**Bit 2: PF**, Parity Flag. The result of low-order 8 bits operation has even parity, this flag is set.

**Bit 1:** Reserved

**Bit 0: CF**, Carry Flag. If CF is set, there has been a carry out or a borrow into the high-order bit of the instruction result.

## 7.4 Address generation

The Execution Unit generates a 20-bit physical address to Bus Interface Unit by the Address Generation. Memory is organized in sets of segments. Each segment contains a 16 bits value. Memory is addressed using a two-component address that consists of a 16-bit segment and 16-bit offset. The Physical Address Generation figure describes how the logical address transfers to the physical address.



## 8. Peripheral Control Block Register

The peripheral control block can be mapped into either memory or I/O space which is to program the FEh register. And it starts at FF00h in I/O space when reset the microprocessor.

The following table is the definition of all the peripheral Control Block Register , and the detail description will arrange on the relation Block Unit.

| Offset (HEX) | Register Name                                                          | Page | Offset (HEX) | Register Name                            | Page |
|--------------|------------------------------------------------------------------------|------|--------------|------------------------------------------|------|
| FE           | Peripheral Control Block Relocation Register                           | 22   | 70           | PIO Mode 0 Register                      | 79   |
| F6           | Reset Configuration Register                                           | 25   | 66           | Timer 2 Mode / Control Register          | 65   |
| F4           | Processor Release Level Register                                       | 22   | 62           | Timer 2 Maxcount Compare A Register      | 66   |
| F2           | Auxiliary configuration Register                                       | 30   | 60           | Timer 2 Count Register                   | 66   |
| F0           | System configuration register                                          | 23   | 5E           | Timer 1 Mode / Control Register          | 63   |
| E6           | Watchdog timer control register                                        | 68   | 5C           | Timer 1 Maxcount Compare B Register      | 65   |
| E4           | Enable RCU Register                                                    | 80   | 5A           | Timer 1 Maxcount Compare A Register      | 65   |
| E2           | Clock Prescaler Register                                               | 80   | 58           | Timer 1 Count Register                   | 65   |
| E0           | Memory Partition Register                                              | 80   | 56           | Timer 0 Mode / Control Register          | 62   |
| DA           | DMA 1 Control Register                                                 | 56   | 54           | Timer 0 Maxcount Compare B Register      | 63   |
| D8           | DMA 1 Transfer Count Register                                          | 58   | 52           | Timer 0 Maxcount Compare A Register      | 63   |
| D6           | DMA 1 Destination Address High Register                                | 58   | 50           | Timer 0 Count Register                   | 62   |
| D4           | DMA 1 Destination Address Low Register                                 | 59   | 44           | Serial Port 0 interrupt control register | 40   |
| D2           | DMA 1 Source Address High Register                                     | 59   | 42           | Serial port 1 interrupt control register | 41   |
| D0           | DMA 1 Source Address Low Register                                      | 59   | 40           | INT4 Control Register                    | 42   |
| CA           | DMA 0 Control Register                                                 | 55   | 3E           | INT3 Control Register                    | 42   |
| C8           | DMA 0 Transfer Count Register                                          | 55   | 3C           | INT2 Control Register                    | 43   |
| C6           | DMA 0 Destination Address High Register                                | 55   | 3A           | INT1 Control Register                    | 43   |
| C4           | DMA 0 Destination Address Low Register                                 | 56   | 38           | INT0 Control Register                    | 44   |
| C2           | DMA 0 Source Address High Register                                     | 56   | 36           | DMA 1/INT6 Interrupt Control Register    | 45   |
| C0           | DMA 0 Source Address Low Register                                      | 56   | 34           | DMA 0/INT5 Interrupt Control Register    | 45   |
| A8           | $\overline{\text{PCS}}$ and $\overline{\text{MCS}}$ Auxiliary Register | 34   | 32           | Timer Interrupt Control Register         | 46   |
| A6           | Midrange Memory Chip Select Register                                   | 33   | 30           | Interrupt Status Register                | 47   |
| A4           | Peripheral Chip Select Register                                        | 35   | 2E           | Interrupt Request Register               | 47   |
| A2           | Low Memory Chip Select Register                                        | 32   | 2C           | Interrupt In-service Register            | 48   |
| A0           | Upper Memory Chip Select Register                                      | 31   | 2A           | InterruptPriority Mask Register          | 49   |
| 88           | Serial Port 0 Baud Rate Divisor Register                               | 75   | 28           | Interrupt Mask Register                  | 50   |
| 86           | Serial Port 0 Receive Register                                         | 75   | 26           | Interrupt Poll Status Register           | 51   |
| 84           | Serial Port 0 Transmit Register                                        | 74   | 24           | Interrupt Poll Register                  | 51   |
| 82           | Serial Port 0 Status Register                                          | 74   | 22           | Interrupt End-of-Interrupt               | 52   |
| 80           | Serial Port 0 Control Register                                         | 72   | 20           | Interrupt Vector Register                | 52   |
| 7A           | PIO Data 1 Register                                                    | 78   | 18           | Serial port 1 baud rate divisor          | 76   |
| 78           | PIO Direction 1 Register                                               | 78   | 16           | Serial port 1 receive register           | 76   |
| 76           | PIO Mode 1 Register                                                    | 78   | 14           | Serial port 1 transmit register          | 76   |
| 74           | PIO Data 0 Register                                                    | 79   | 12           | Serial port 1 status register            | 75   |
| 72           | PIO Direction 0 Register                                               | 79   | 10           | Serial port 1 control register           | 75   |

**Peripheral Control Block Relocation Register:**
**Offset : FEh**
**Reset Value : 20FFh**

|     |     |     |                    |          |    |   |   |   |   |   |   |   |   |   |   |
|-----|-----|-----|--------------------|----------|----|---|---|---|---|---|---|---|---|---|---|
| 15  | 14  | 13  | 12                 | 11       | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res | S/M | Res | M/I $\overline{O}$ | R19 - R8 |    |   |   |   |   |   |   |   |   |   |   |

The peripheral control block is mapped into either memory or I/O space by programming this register. When the other chip selects (  $\overline{PCSx}$  or  $\overline{MCSx}$  ) are programmed to zero wait states and ignore the external ready, the  $\overline{PCSx}$  or  $\overline{MCSx}$  can overlap the control block.

**Bit 15:** Reserved

**Bit 14:** S/ $\overline{M}$  , Slave/Master – Configures the interrupt controller

**set 0 :** Master mode, **set 1:** Slaved mode

**Bit 13 :** Reserved

**Bit 12:** M/ $\overline{IO}$  , Memory/IO space. At reset, this bit is set to 0 and the PCB map start at FF00h in I/O space.

**set 1-** The peripheral control block (PCB) is located in memory space.

**set 0-** The PCB is located in I/O space.

**Bit 11-0 : R19-R8** , Relocation Address Bits

The upper address bits of the PCB base address. The lower eight bits default to 00h. When the PCB is mapped to I/O space, the R19-R16 must be programmed to 0000b.

**Processor Release Level Register**
**Offset : F4h**
**Reset Value : F9h**

|     |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|-----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15  | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PRL |    |    |    |    |    |   |   | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |

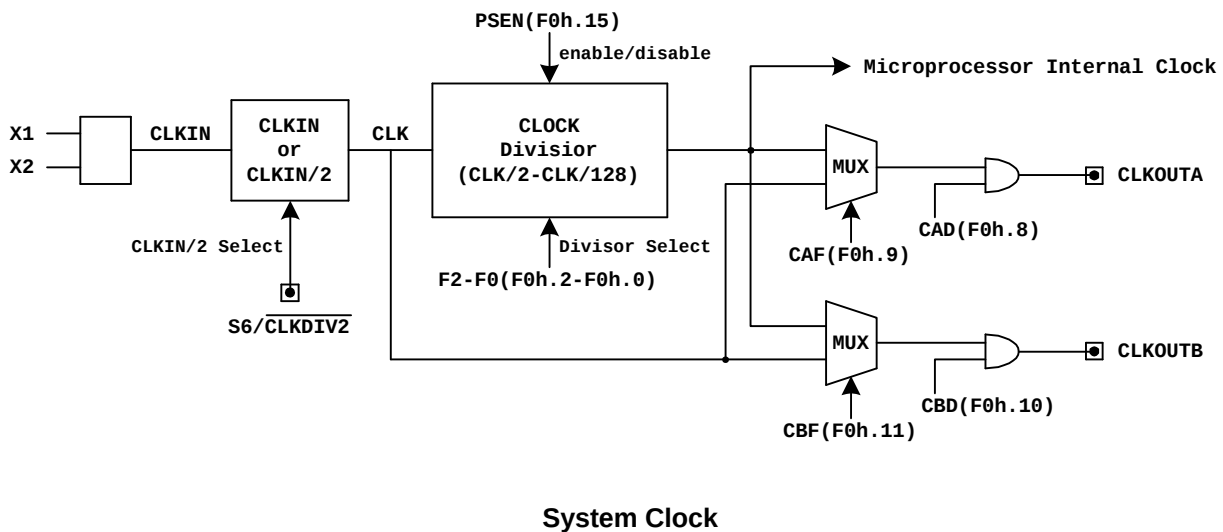
Read only register that specifies the processor release version and RDC identify number

**Bit 15-8 :** Processor version

01h : version A , 02h : version B, 03h : version C, 04h : version D

**Bit 7-0 :** RDC identify number - D9h

## 9. System Clock Block



### Power-Save Control Register

Offset : F0h  
Reset Value : 0000h

| 15   | 14     | 13 | 12 | 11  | 10  | 9   | 8   | 7 | 6 | 5 | 4 | 3 | 2  | 1  | 0  |
|------|--------|----|----|-----|-----|-----|-----|---|---|---|---|---|----|----|----|
| PSEN | MCSBIT | 0  | 0  | CBF | CBD | CAF | CAD | 0 | 0 | 0 | 0 | 0 | F2 | F1 | F0 |

**Bit 15: PSEN**, Enable Power-save Mode. This bit is cleared by hardware when an external interrupt occurs. This bit does not be changed when software interrupts (INT instruction) and exceptions occurs.

Set 1: enable power-save mode and divides the internal operating clock by the value in F2-F0.

**Bit14 : MCSBIT**, MCS0 control bit. Set to 0: The MCS0 operate normally. Set to 1: MCS0 is active over the entire MCSx range

**Bit13-12:** Reserved

**Bit 11: CBF**, CLKOUTB Output Frequency selection.

Set 1: CLKOUTB output frequency is same as crystal input frequency.

Set 0 : CLKOUTB output frequency is from the clock divisor, which frequency is same as that of microprocessor internal clock.

**Bit 10 : CBD**, CLKOUTB Drive Disable

Set 1: Disable the CLKOUTB. This pin will be three-state.

Set 0 : Enable the CLKOUTB.

**Bit 9: CAF**, CLKOUTA Output Frequency selection.

Set 1: CLKOUTA output frequency is same as crystal input frequency.

Set 0 : CLKOUTB output frequency is from the clock divisor, which frequency is same as that of microprocessor

internal clock .

**Bit 8: CAD, CLKOUTA Drive Disable.**

Set 1: Disable the CLKOUTA. This pin will be three-state.

Set 0 : Enable the CLKOUTA.

**Bit 7-3 : Reserved**

**Bit 2-0: F2- F0, Clock Divisor Select.**

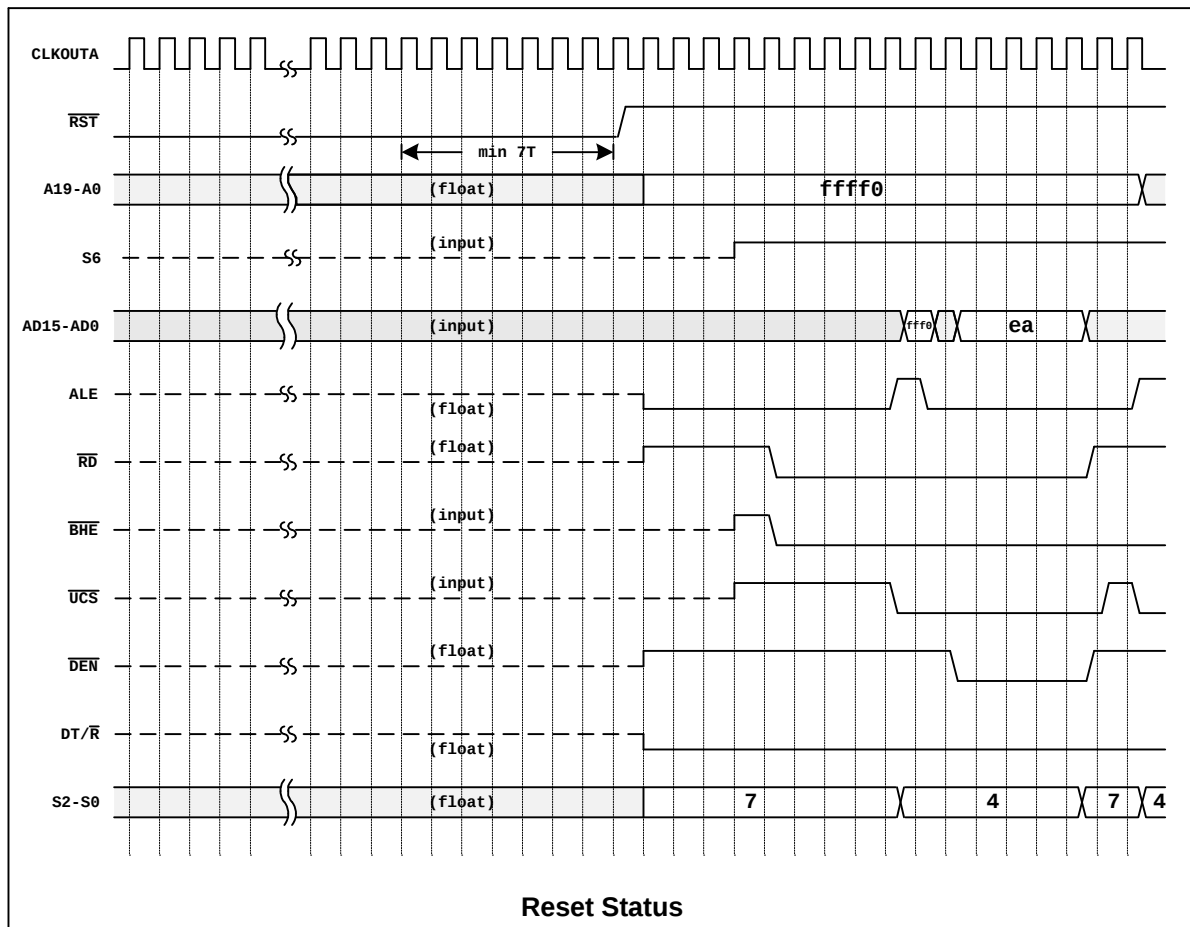
| <b>F2,</b> | <b>F1,</b> | <b>F0</b> | <b>-----</b> | <b>Divider Factor</b> |
|------------|------------|-----------|--------------|-----------------------|
| 0,         | 0,         | 0         | ----         | Divide by 1           |
| 0,         | 0,         | 1         | ----         | Divide by 2           |
| 0,         | 1,         | 0         | ----         | Divide by 4           |
| 0,         | 1,         | 1         | ----         | Divide by 8           |
| 1,         | 0,         | 0         | ----         | Divide by 16          |
| 1,         | 0,         | 1         | ----         | Divide by 32          |
| 1,         | 1,         | 0         | ----         | Divide by 64          |
| 1,         | 1,         | 1         | ----         | Divide by 128         |

---

## 10. Reset

Processor initialization is accomplished with activation of the  $\overline{\text{RST}}$  pin. To reset the processor, this pin should be held low for at least seven oscillator periods. The Reset Status Figure shows the status of the  $\overline{\text{RST}}$  pin and others relation pins.

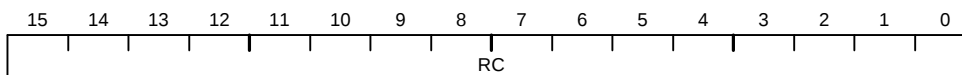
When  $\overline{\text{RST}}$  from low go high , the state of input pin (with weakly pull-up or pull-down) will be latched , and each pin will perform the individual function. The AD15-AD0 will be latched into the register F6h.  $\overline{\text{UCS}} / \overline{\text{ONCE1}}$  ,  $\overline{\text{LCS}} / \overline{\text{ONCE0}}$  will enter ONCE mode (All of the pins will floating except X1 , X2) when with pull-low resistors. The input clock will divide by 2 when S6/  $\overline{\text{CLKDIV2}}$  with pull-low resistor. The AD15-AD0 bus will not drive the address phase during  $\overline{\text{UCS}}$  ,  $\overline{\text{LCS}}$  cycle if  $\overline{\text{BHE}} / \overline{\text{ADEN}}$  with pull-low resistor



### Reset Configuration Register

Offset : F6h

Reset Value : AD15-AD0



**Bit 15- 0 : RC** ,Reset Configuration AD15 – AD0.

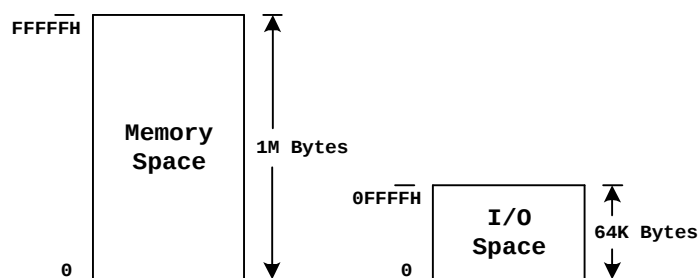
The AD15 to AD0 must with weakly pull-up or pull-down resistors to correspond the contents when AD15-AD0 be latched into this register during the  $\overline{\text{RST}}$  pin from low go high. And the value of the reset configuration register provides the system information when software read this register. This register is read only and the contents remain valid until the next processor reset.

## 11. Bus Interface Unit

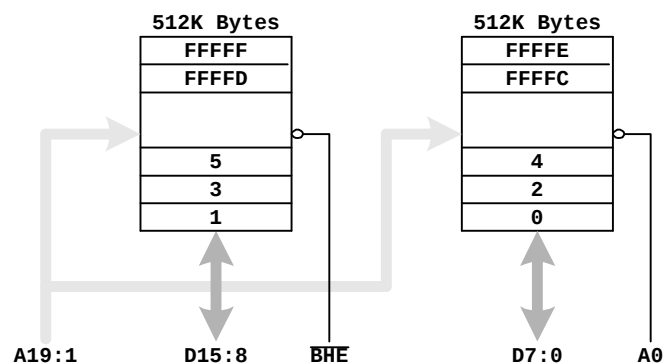
The bus interface unit drives address, data, status and control information to define a bus cycle. The bus A19-A0 are non-multiplex memory or I/O address. The AD15-AD0 are multiplexed address and data bus for memory or I/O accessing. The  $\overline{S2} - \overline{S1}$  are encoded to indicate the bus status, which is described in the Pin Description table in page 5. The Basic Application System Block (page 10) and Read/Write Timing Diagram (page 12) describe the basic bus operation.

### 11.1 Memory and I/O interface

The memory space consists of 1M bytes (512k 16-bit port) and the I/O space consists of 64k bytes (32k 16-bit port). Memory devices exchange information with the CPU during memory read, memory write and instruction fetch bus cycles. I/O read and I/O write bus cycles use a separate I/O address space. Only IN/OUT instruction can access I/O address space, and information must be transferred between the peripheral device and the AX register. The first 256 bytes of I/O space can be accessed directly by the I/O instructions. The entire 64k bytes I/O address space can be accessed indirectly, through the DX register. I/O instructions always force address A19-A16 to low level.



**Memory and I/O Space**



**Physical Data Bus Models**

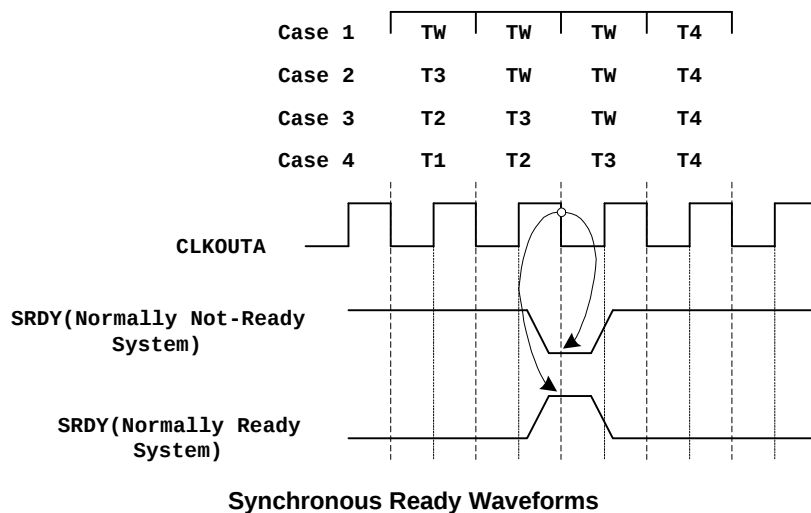
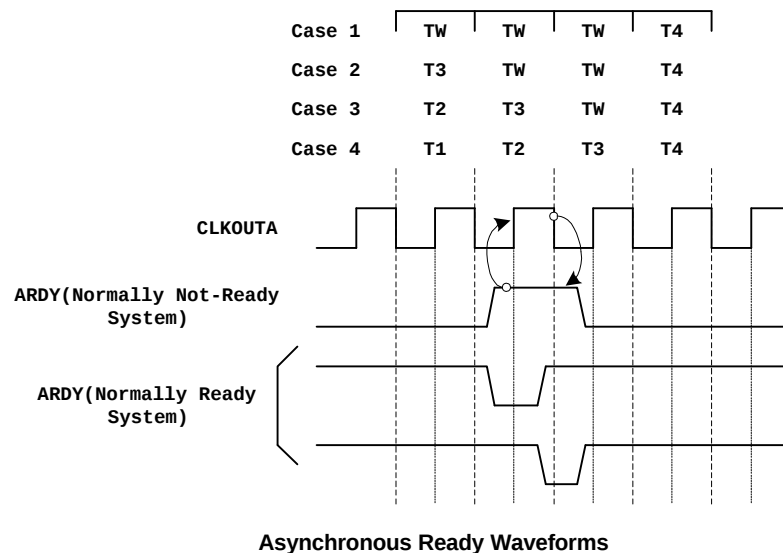
### 11.2 Data Bus

The memory address space data bus is physically implemented by dividing the address space into two banks of up to 512k bytes. Each one bank connects to the lower half of the data bus and contains the even-addressed bytes (A0=0). The other

bank connects to the upper half of the data bus and contains odd-addressed bytes (A0=1). A0 and  $\overline{\text{BHE}}$  determine whether one bank or both banks participate in the data transfer.

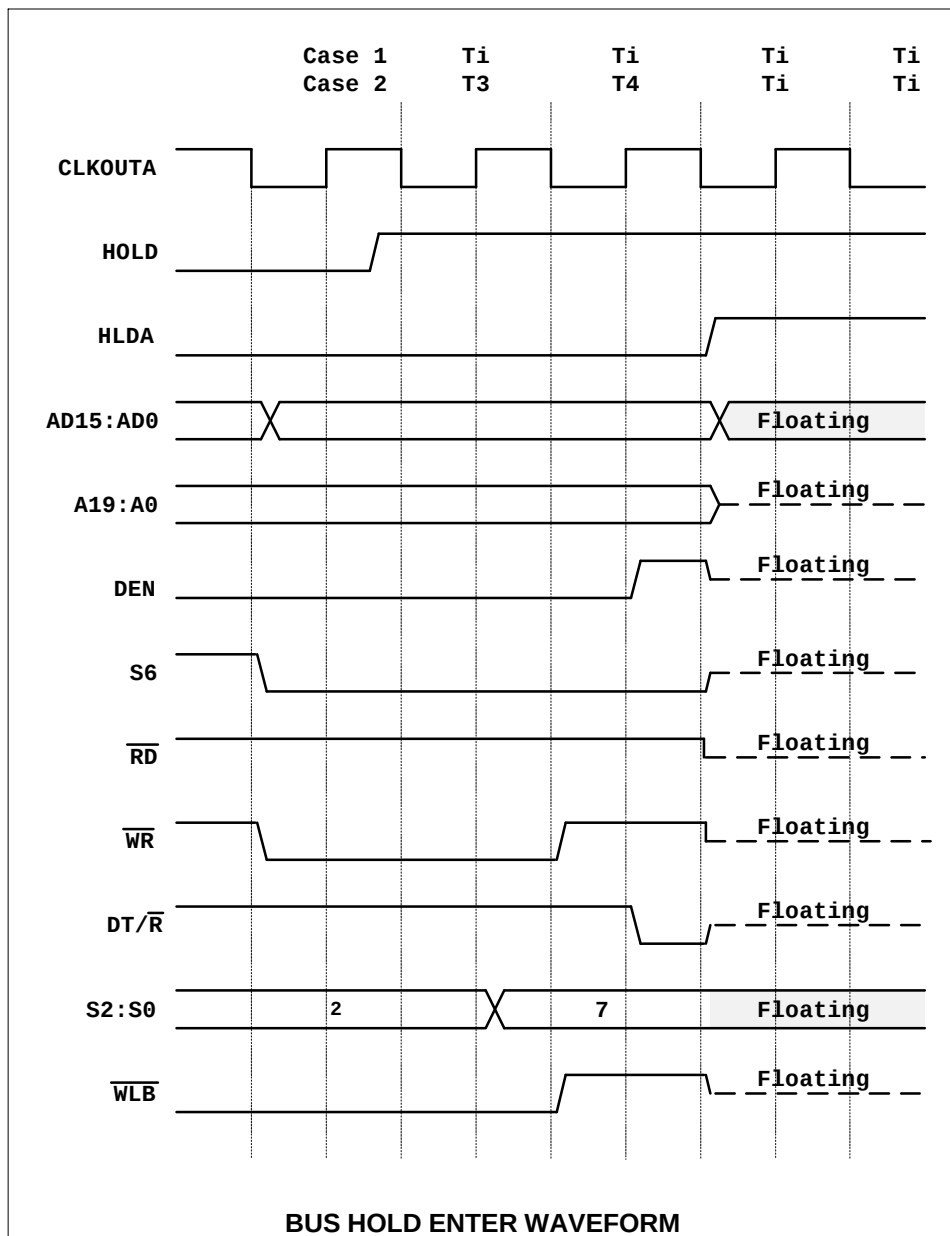
### 11.3 Wait States

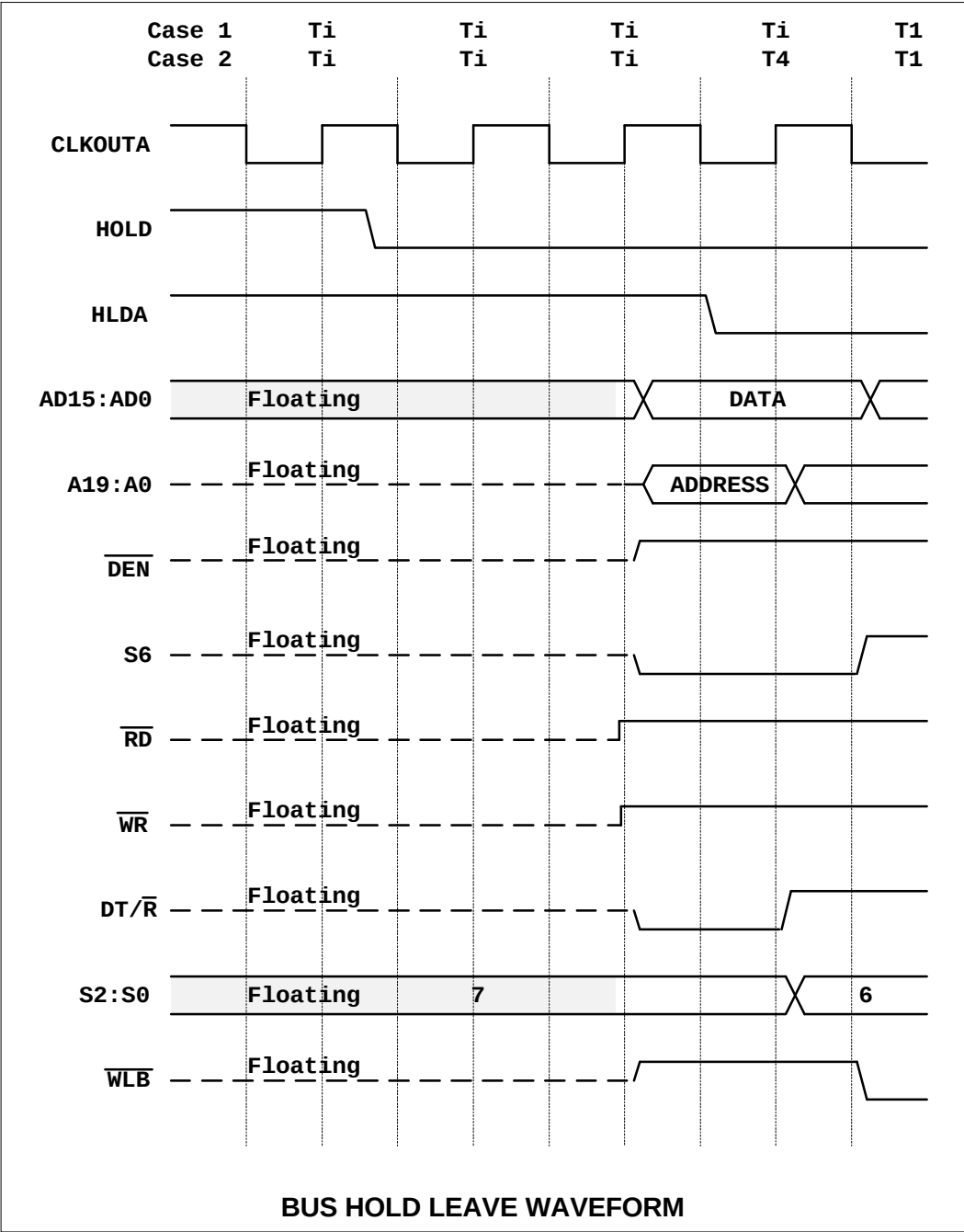
Wait states extend the data phase of the bus cycle. The ARDY or SRDY input with high level will insert wait states. To avoid wait states, ARDY and SRDY must be low within a specified setup time prior to phase 2 of T2. To insert wait states, ARDY or SRDY must drive high within a specified setup time prior to phase 2 of T2 or phase 1 of T3. If the ARDY is not used, tie this pin low to yield control to SRDY. If the SRDY is not used, tie this pin low to yield control to ARDY. The SRDY/PIO6 is multi function pin, and SRDY internally pull-down when this pin is programmed for PIO function.



## 11.4 Bus Hold

When the bus hold requested ( **HOLD** pin active high) by the another bus master, the microprocessor will issue a **HLDA** in response to a **HOLD** request at the end of **T4** or **Ti**. When the microprocessor is in hold status (**HLDA** is high), the **AD15-A0**, **A19-A0**,  $\overline{\text{WR}}$ ,  $\overline{\text{RD}}$ ,  $\overline{\text{DEN}}$ ,  $\overline{\text{S1-S0}}$ ,  $\overline{\text{S6}}$ ,  $\overline{\text{BHE}}$ ,  $\text{DT}/\overline{\text{R}}$ ,  $\overline{\text{WHB}}$  and  $\overline{\text{WLB}}$  are floating, and the  $\overline{\text{UCS}}$ ,  $\overline{\text{LCS}}$ ,  $\overline{\text{PCS6-PCS5}}$ ,  $\overline{\text{MCS3-MCS0}}$  and  $\overline{\text{PCS3-PCS0}}$  will be drive high. After **HOLD** is detected as being low, the microprocessor will lower the **HLDA**.





## 11.5 Bus Width

The R8820LV default is 16 bits bus access. And the bus can be programmed as 8-bits or 16-bits access during memory or I/O access is located in the  $\overline{\text{LCS}}$  or  $\overline{\text{MCSx}}$  or  $\overline{\text{PCSx}}$  address space. The  $\overline{\text{UCS}}$  code- fetched selection is 16 bits bus width, which can not be changed by programmed the register.

| Auxiliary configuration Register |    |    |    |    |    |   |   |   |       |      |       |      |      |      | Offset : F2h<br>Reset Value : 0000h |  |  |  |
|----------------------------------|----|----|----|----|----|---|---|---|-------|------|-------|------|------|------|-------------------------------------|--|--|--|
| 15                               | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6     | 5    | 4     | 3    | 2    | 1    | 0                                   |  |  |  |
| Reserved                         |    |    |    |    |    |   |   |   | ENRX1 | RTS1 | ENRX0 | RTS0 | LSIZ | MSIZ | IOSIZ                               |  |  |  |

**Bit 15-7:** Reserved.

**Bit 6: ENRX1**, Enable the Receiver Request of Serial port 1.

Set 1: The  $\overline{\text{CTS1}}/\overline{\text{ENRX1}}$  pin is configured as  $\overline{\text{ENRX1}}$

Set 0: The  $\overline{\text{CTS1}}/\overline{\text{ENRX1}}$  pin is configured as  $\overline{\text{CTS1}}$

**Bit 5: RTS1**, Enable Request to Send of Serial port 1.

Set 1: The  $\overline{\text{RTR1}}/\overline{\text{RTS1}}$  pin is configured as  $\overline{\text{RTS1}}$

Set 0: The  $\overline{\text{RTR1}}/\overline{\text{RTS1}}$  pin is configured as  $\overline{\text{RTR1}}$

**Bit 4: ENRX0**, Enable the Receiver Request of Serial port 0.

Set 1: The  $\overline{\text{CTS0}}/\overline{\text{ENRX0}}$  pin is configured as  $\overline{\text{ENRX0}}$

Set 0: The  $\overline{\text{CTS0}}/\overline{\text{ENRX0}}$  pin is configured as  $\overline{\text{CTS0}}$

**Bit 3: RTS0**, Enable Request to Send of Serial port 0.

Set 1: The  $\overline{\text{RTR0}}/\overline{\text{RTS0}}$  pin is configured as  $\overline{\text{RTS0}}$

Set 0: The  $\overline{\text{RTR0}}/\overline{\text{RTS0}}$  pin is configured as  $\overline{\text{RTR0}}$

**Bit 2: LSIZ**,  $\overline{\text{LCS}}$  Data Bus Size selection. This bit can not be changed while executing from  $\overline{\text{LCS}}$  space or while the Peripheral Control Block is overlaid with  $\overline{\text{PCS}}$  space.

Set 1: 8 bits data bus access when the memory access located in the  $\overline{\text{LCS}}$  memory space.

Set 0: 16 bits data bus access when the memory access located in the  $\overline{\text{LCS}}$  memory space.

**Bit 1: MSIZ**,  $\overline{\text{MCSx}}$ ,  $\overline{\text{PCSx}}$  Memory Data Bus Size selection. This bit can not be changed while executing from the associated or while the Peripheral Control Block is overlaid on this address space.

Set 1: 8 bits data bus access when the memory access locate in the selection memory space.

Set 0 : 16 bits data bus access when the memory access locate in the selection memory space.

**Bit 0: IOSIZ**, I/O Space Data Bus Size selection. This bit determines the width of the data bus for all I/O space accesses.

Set 1: 8 bits data bus access.

Set 0 : 16 bits data bus access.

## 12. Chip Select Unit

The Chip Select Unit provides 12 programmable chip select pins to access a specific memory or peripheral device.

The chip selects are programmed through five peripheral control registers (A0h, A2h, A4h, A6h, A8h). And all of the chip selects can be insert wait states by programmed the peripheral control register.

### 12.1 UCS

The  $\overline{\text{UCS}}$  default to active on reset for program code access. The memory active range is upper 512k (80000h – FFFFFh), which is programmable. And the default memory active range of  $\overline{\text{UCS}}$  is 64k ( F0000h – FFFFFh).

The  $\overline{\text{UCS}}$  active to drive low four CLKOUTA oscillators if no wait state inserts. There are three wait-states insert to  $\overline{\text{UCS}}$  active cycle on reset.

**Upper Memory Chip Select Register**

Offset : A0h  
Reset Value :F03Bh

| 15 | 14        | 13 | 12 | 11 | 10 | 9 | 8 | 7  | 6 | 5 | 4 | 3 | 2  | 1  | 0  |
|----|-----------|----|----|----|----|---|---|----|---|---|---|---|----|----|----|
| 1  | LB2 - LB0 |    |    | 0  | 0  | 0 | 0 | DA | 0 | 1 | 1 | 1 | R2 | R1 | R0 |

**Bit 15 :** Reserved

**Bit 14-12 :** LB2-LB0, Memory block size selection for  $\overline{\text{UCS}}$  chip select pin.

The  $\overline{\text{UCS}}$  chip select pin active region can be configured by the LB2-LB0.

The default memory block size is from F0000h to FFFFFh.

**LB2, LB1, LB0 ---- Memory Block size , Start address, End Address**

|   |   |   |      |      |  |          |  |          |
|---|---|---|------|------|--|----------|--|----------|
| 1 | 1 | 1 | ---- | 64k  |  | , F0000h |  | , FFFFFh |
| 1 | 1 | 0 | ---- | 128k |  | , E0000h |  | , FFFFFh |
| 1 | 0 | 0 | ---- | 256k |  | , C0000h |  | , FFFFFh |
| 0 | 0 | 0 | ---- | 512k |  | , 80000h |  | , FFFFFh |

**Bit 11-8 :** Reserved

**Bit 7 : DA** , Disable Address. If the  $\overline{\text{BHE}} / \overline{\text{ADEN}}$  pin is held high on the rising edge of  $\overline{\text{RST}}$  , then the DA bit is valid to enable/disable the address phase of the AD bus. If the  $\overline{\text{BHE}} / \overline{\text{ADEN}}$  pin is held high on the rising edge of  $\overline{\text{RST}}$  , the AD bus always drive the address and data.

Set 1 : Disable the address phase of the AD15 – AD0 bus cycle when  $\overline{\text{UCS}}$  is asserted.

Set 0 : Enable the address phase of the AD15 – AD0 bus cycle when  $\overline{\text{UCS}}$  is asserted.

**Bit 6-3:** Reserved

**Bit 2 : R2**, Ready Mode. This bit is used to configure the ready mode for  $\overline{\text{UCS}}$  chip select.

Set 1: external ready is ignored.

Set 0: external ready is required.

**Bit 1-0 : R1-R0**, Wait-State value. When R2 is set to 0, it can inserted wait-state into an access to the  $\overline{\text{UCS}}$  memory area.

(R1,R0) = (0,0) -- 0 wait-state ; (R1,R0) = (0,1) -- 1 wait-state

(R1,R0) = (1,0) -- 2 wait-state ; (R1,R0) = (1,1) -- 3 wait-state

## 12.2 $\overline{\text{LCS}}$

The lower 512k bytes (00000h-7FFFFh) memory region chip selects. The memory active range is programmable, which has no default size on reset. So the A2h register must be programmed first before to access the target memory range. The  $\overline{\text{LCS}}$  pin is not active on reset, but any read or write access to the A2h register activates this pin.

### Low Memory Chip Select Register

Offset : A2h

Reset Value : —

| 15 | 14        | 13 | 12 | 11 | 10 | 9 | 8 | 7  | 6   | 5 | 4 | 3 | 2  | 1  | 0  |
|----|-----------|----|----|----|----|---|---|----|-----|---|---|---|----|----|----|
| 0  | UB2 - UB0 |    |    | 1  | 1  | 1 | 1 | DA | PSE | 1 | 1 | 1 | R2 | R1 | R0 |

**Bit 15:** Reserved

**Bit 14-12 :** UB2-UB0, Memory block size selection for  $\overline{\text{LCS}}$  chip select pin

The  $\overline{\text{LCS}}$  chip select pin active region can be configured by the UB2-UB0.

The  $\overline{\text{LCS}}$  pin is not active on reset, but any read or write access to the A2h (LMCS) register activates this pin.

**UB2, UB1, UB0 ---- Memory Block size , Start address, End Address**

|   |   |   |      |      |  |        |  |        |
|---|---|---|------|------|--|--------|--|--------|
| 0 | 0 | 0 | ---- | 64k  |  | 00000h |  | 0FFFFh |
| 0 | 0 | 1 | ---- | 128k |  | 00000h |  | 1FFFFh |
| 0 | 1 | 1 | ---- | 256k |  | 00000h |  | 3FFFFh |
| 1 | 1 | 1 | ---- | 512k |  | 00000h |  | 7FFFFh |

**Bit 11-8 :** Reserved

**Bit 7 : DA** , Disable Address. If the  $\overline{\text{BHE}} / \overline{\text{ADEN}}$  pin is held high on the rising edge of  $\overline{\text{RST}}$  , then the DA bit is valid to enable/disable the address phase of the AD bus. If the  $\overline{\text{BHE}} / \overline{\text{ADEN}}$  pin is held low on the rising edge of  $\overline{\text{RST}}$  , the AD bus always drive the address and data.

Set 1 : Disable the address phase of the AD15 – AD0 bus cycle when  $\overline{\text{LCS}}$  is asserted.

Set 0 : Enable the address phase of the AD15 – AD0 bus cycle when  $\overline{\text{LCS}}$  is asserted.

**Bit 6 : PSE**, PSRAM Mode Enable. This bit is used to enable PSRAM support for the  $\overline{\text{LCS}}$  chip select memory space. The refresh control unit registers E0h,E2h,E4h must be configured for auto refresh before PSRAM support is enabled.

PSE set to 1: PSRAM support is enable

PSE set to 0: PSRAM support is disable

**Bit 5-3:** Reserved

**Bit 2 : R2**, Ready Mode. This bit is used to configure the ready mode for  $\overline{\text{LCS}}$  chip select.

Set 1: external ready is ignored.

Set 0: external ready is required.

**Bit 1-0 : R1-R0**, Wait-State value. When R2 is set to 0, it can inserted wait-state into an access to the  $\overline{\text{LCS}}$  memory area.

(R1,R0) = (0,0) -- 0 wait-state ; (R1,R0) = (0,1) -- 1 wait-state

(R1,R0) = (1,0) -- 2 wait-state ; (R1,R0) = (1,1) -- 3 wait-state

## 12.3 $\overline{\text{MCSx}}$

The memory block of  $\overline{\text{MCS4}}$  -  $\overline{\text{MCS0}}$  can be located anywhere within the 1M bytes memory space, exclusive of the areas associated with the  $\overline{\text{UCS}}$  and  $\overline{\text{LCS}}$  chip selects. The maximum  $\overline{\text{MCSx}}$  active memory range is 512k bytes.

The  $\overline{\text{MCSx}}$  chip selects are programmed through two registers A6h and A8h, and these select pins are not active on reset.

Both A6h and A8h registers must be accessed with a read or write to activate  $\overline{\text{MCS4}}$  -  $\overline{\text{MCS0}}$ . There aren't default value on A6h and A8h registers, so the A6h and A8h must be programmed first before  $\overline{\text{MCS4}}$  -  $\overline{\text{MCS0}}$  active.

### Midrange Memory Chip Select Register

Offset : A6h

Reset Value : —

| 15          | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2  | 1  | 0  |
|-------------|----|----|----|----|----|---|---|---|---|---|---|---|----|----|----|
| BA19 - BA13 |    |    |    |    |    |   |   | 1 | 1 | 1 | 1 | 1 | R2 | R1 | R0 |

**Bit 15-7 : BA19-BA13**, Base Address. The BA19-BA13 correspond to bits 19-13 of the 1M bytes (20-bits) programmable base address of the  $\overline{\text{MCS}}$  chip select block. The bits 12 to 0 of the base address are always 0.

The base address can be set to any integer multiple of the size of the memory block size selected in these bits. For example, if the midrange block is 32Kbytes, only the bits BA19 to BA15 can be programmed. So the block address could be locate at 20000h or 38000h but not in 22000h.

The base address of the  $\overline{\text{MCS}}$  chip select can be set to 00000h only if the  $\overline{\text{LCS}}$  chip select is not active. And the  $\overline{\text{MCS}}$  chip select address range is not allowed to overlap the  $\overline{\text{LCS}}$  chip select address range.

The  $\overline{\text{MCS}}$  chip select address range also is not allowed to overlap the  $\overline{\text{UCS}}$  chip select address range.

**Bit 8-3** : Reserved

**Bit 2: R2**, Ready Mode. This bit is configured to enable/disable the wait states inserted for the  $\overline{\text{MCS}}$  chip selects. The R1,R0 bits of this register determine the number of wait state to insert.

set to 1: external ready is ignored

set to 0: external ready is required

**Bit 1-0 : R1-R0**, Wait-State value. The R1,R0 determines the number of wait states inserted into a  $\overline{\text{MCS}}$  access.

(R1,R0) : (1,1) – 3 wait states , (1,0) – 2 wait states, (0,1) – 1 wait states , (0,0) – 0 wait states

### PCS and MCS Auxiliary Register

Offset : A8h

Reset Value : —

|    |    |    |    |    |    |   |   |    |    |   |   |   |    |    |    |
|----|----|----|----|----|----|---|---|----|----|---|---|---|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7  | 6  | 5 | 4 | 3 | 2  | 1  | 0  |
| 1  |    |    |    |    |    |   |   | EX | MS | 1 | 1 | 1 | R2 | R1 | R0 |

**Bit 15:** Reserved

**Bit 14-8: M6-M0,** MCS Block Size. These bits determines the total block size for the  $\overline{\text{MCS3}}$  -  $\overline{\text{MCS0}}$  chip selects. Each individual chip select is active for one quarter of the total block size. For example, if the block size is 32K bytes and the base address is located at 20000h. The individual active memory address range of  $\overline{\text{MCS3}}$  to  $\overline{\text{MCS0}}$  is  $\overline{\text{MCS0}}$  – 20000h to 21FFF,  $\overline{\text{MCS1}}$  - 22000 to 23FFFh,  $\overline{\text{MCS2}}$  - 24000h to 25FFFh,  $\overline{\text{MCS3}}$  - 26000h to 27FFFh.  $\overline{\text{MCSx}}$  total block size is defined by M6-M0,

**M6-M0** , **Total block size,**  $\overline{\text{MCSx}}$  **address active range**

|          |   |      |   |      |
|----------|---|------|---|------|
| 0000001b | , | 8k   | , | 2k   |
| 0000010b | , | 16k  | , | 4k   |
| 0000100b | , | 32k  | , | 8k   |
| 0001000b | , | 64k  | , | 16k  |
| 0010000b | , | 128k | , | 32k  |
| 0100000b | , | 256k | , | 64k  |
| 1000000b | , | 512k | , | 128k |

**Bit 7 : EX,** Pin Selector. This bit configures the multiplex output which the  $\overline{\text{PCS6}}$  -  $\overline{\text{PCS5}}$  pins as chip selects or A2-A1.

Set 1 :  $\overline{\text{PCS6}}$  ,  $\overline{\text{PCS5}}$  are configured as peripheral chip select pins.

Set 0:  $\overline{\text{PCS6}}$  is configured as address bit A2,  $\overline{\text{PCS5}}$  is configured as A1.

**Bit 6: MS,** Memory or I/O space Selector.

Set 1: The  $\overline{\text{PCSx}}$  pins are active for memory bus cycle.

Set 0: The  $\overline{\text{PCSx}}$  pins are active for I/O bus cycle.

**Bit 5-3 :** Reserved

**Bit 2 : R2,** Ready Mode. This bit is configured to enable/disable the wait states inserted for the  $\overline{\text{PCS5}}$ ,  $\overline{\text{PCS6}}$  chip selects. The

R1,R0 bits of this register determine the number of wait state to insert.

set to 1: external ready is ignored

set to 0: external ready is required

**Bit 1-0 : R1-R0,** Wait-State value. The R1,R0 determines the number of wait states inserted into a  $\overline{\text{PCS5}}$  -  $\overline{\text{PCS6}}$  access.

(R1,R0) : (1,1) – 3 wait states , (1,0) – 2 wait states, (0,1) – 1 wait states , (0,0) – 0 wait states

## 12.4 $\overline{\text{PCSx}}$

The peripheral or memory chip selects which are programmed through A4h and A8h register to define these pins.

The base address memory block can be located anywhere within the 1M bytes memory space, exclusive of the areas associated with the  $\overline{\text{MCS4}}$ ,  $\overline{\text{LCS}}$  and  $\overline{\text{MCS}}$  chip elects. If the chip selects are mapped to I/O space, the access range is 64k bytes.

$\overline{\text{PCS6}} - \overline{\text{PCS5}}$  can be configured from 0 wait-state to 3 wait-states.  $\overline{\text{PCS3}} - \overline{\text{PCS0}}$  can be configured from 0 wait-state to 15 wait-states.

## Peripheral Chip Select Register

Offset : A4h

Reset Value : —

| 15          | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3  | 2  | 1  | 0  |
|-------------|----|----|----|----|----|---|---|---|---|---|---|----|----|----|----|
| BA19 - BA11 |    |    |    |    |    |   |   |   | 1 | 1 | 1 | R3 | R2 | R1 | R0 |

**Bit 15-7 : BA19-BA11**, Base Address. BA19-BA11 correspond to bit 19-11 of the 1M bytes (20-bits) programmable base address of the  $\overline{\text{PCS}}$  chip select block.

When the  $\overline{\text{PCS}}$  chip selects are mapped to I/O space, BA19-BA16 must be wrote to 0000b because the I/O address bus in only 64K bytes (16-bits) wide.

### $\overline{\text{PCSx}}$ address range:

|                          |   |                   |   |                   |
|--------------------------|---|-------------------|---|-------------------|
| $\overline{\text{PCS0}}$ | : | Base Address      | - | Base Address+255  |
| $\overline{\text{PCS1}}$ | : | Base Address+256  | - | Base Address+511  |
| $\overline{\text{PCS2}}$ | : | Base Address+512  | - | Base Address+767  |
| $\overline{\text{PCS3}}$ | : | Base Address+768  | - | Base Address+1023 |
| $\overline{\text{PCS4}}$ | : | Base Address+1280 | - | Base Address+1535 |
| $\overline{\text{PCS5}}$ | : | Base Address+1536 | - | Base Address+1791 |

**Bit 6-4:** Reserved

**Bit 3: R3; Bit 1-0: R1,R0** ,Wait-State Value. The R3,R1,R0 determines the number of wait-states inserted into a  $\overline{\text{PCS3}}$  -  $\overline{\text{PCS0}}$  access.

### R3, R1, R0 -- Wait States

|    |    |   |    |    |
|----|----|---|----|----|
| 0, | 0, | 0 | -- | 0  |
| 0, | 0, | 1 | -- | 1  |
| 0, | 1, | 0 | -- | 2  |
| 0, | 1, | 1 | -- | 3  |
| 1, | 0, | 0 | -- | 5  |
| 1, | 0, | 1 | -- | 7  |
| 1, | 1, | 0 | -- | 9  |
| 1, | 1, | 1 | -- | 15 |

**Bit 2 : R2**, Ready Mode. This bit is configured to enable/disable the wait states inserted for the  $\overline{\text{PCS3}}$  -  $\overline{\text{PCS0}}$  chip selects.

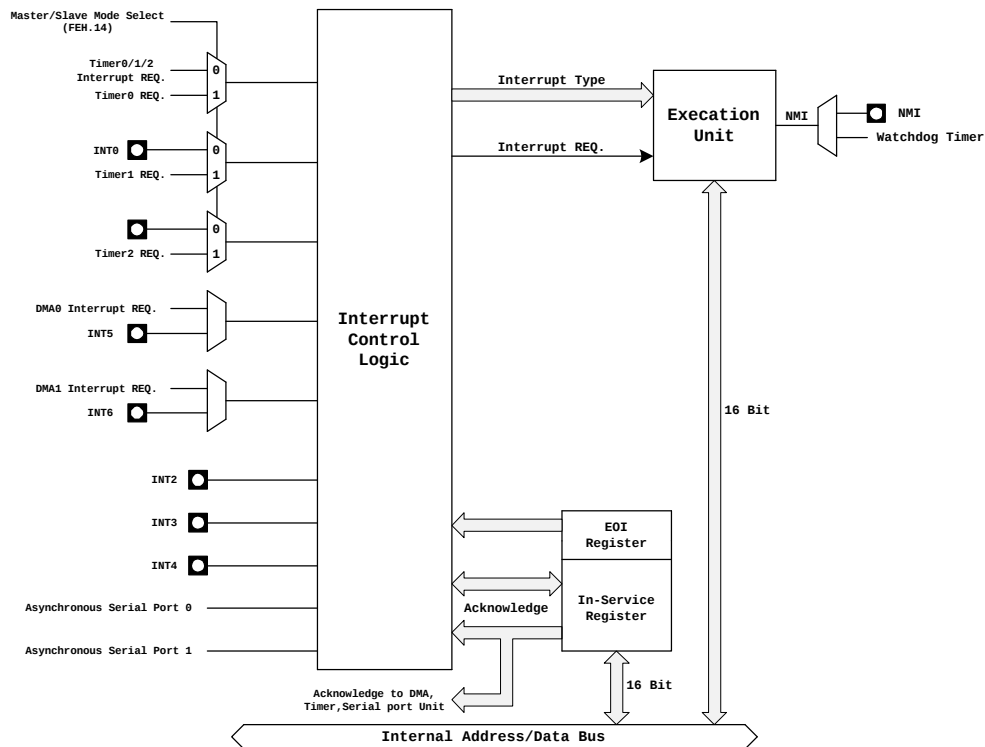
The R3,R1,R0 bits determine the number of wait state to insert.

set to 1: external ready is ignored

set to 0: external ready is required

## 13. Interrupt Controller Unit

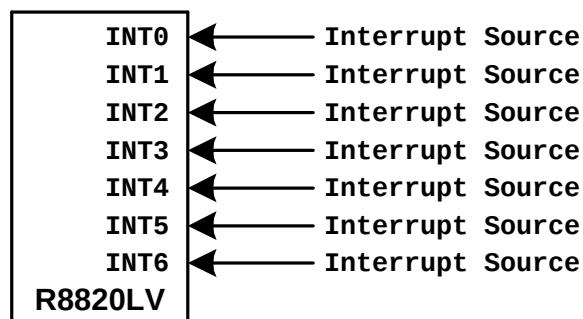
There are 16 interrupt requests source connect to the controller: 7 maskable interrupt pins ( INT0 – INT6); 2 non-maskable interrupts (NMI pin , WDT) ; 7 internal unit request source ( Timer 0, 1,2 ;DMA 0,1 ; Asynchronous serial port 0, 1).



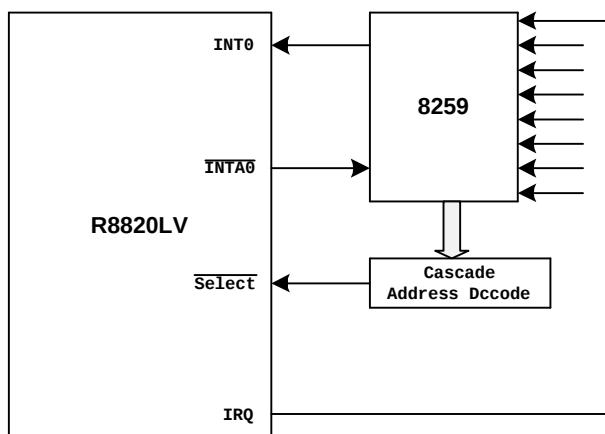
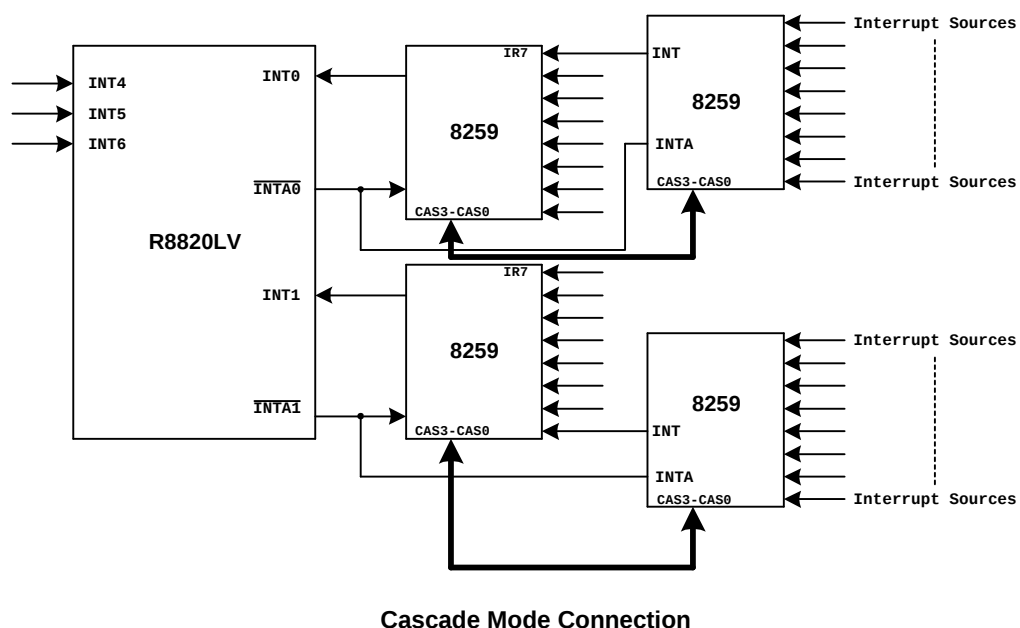
**Interrupt Control Unit Block Diagram**

### 13.1 Master Mode and Slave Mode

The interrupt controller can be programmed as a master or slave mode. (program FEh , bit 14). The master mode has two connections : Fully Nested Mode connection or Cascade Mode connection.



**Fully Nested Mode Connections**



## 13.2 Interrupt Vector, Type and Priority

The following table shows the interrupt vector addresses, type and the priority. The maskable interrupt priority can be changed by programmed the priority register. The Vector addresses for each interrupt are fixed.

| Interrupt source       | Interrupt Type | Vector Address | EOI Type | Priority | Note |
|------------------------|----------------|----------------|----------|----------|------|
| Divide Error Exception | 00h            | 00h            |          | 1        |      |
| Trace interrupt        | 01h            | 04h            |          | 1-1      | *    |
| NMI                    | 02h            | 08h            |          | 1-2      | *    |
| Breakpoint Interrupt   | 03h            | 0Ch            |          | 1        |      |

|                                   |         |     |    |     |      |
|-----------------------------------|---------|-----|----|-----|------|
| INT0 Detected Over Flow Exception | 04h     | 10h |    | 1   |      |
| Array Bounds Exception            | 05h     | 14h |    | 1   |      |
| Undefined Opcode Exception        | 06h     | 18h |    | 1   |      |
| ESC Opcode Exception              | 07h     | 1Ch |    | 1   |      |
| Timer 0                           | 08h     | 20h | 08 | 2-1 | */** |
| Reserved                          | 09h     |     |    |     |      |
| DMA 0/INT5                        | 0Ah     | 28h | 0A | 3   | **   |
| DMA 1/INT6                        | 0Bh     | 2Ch | 0B | 4   | **   |
| INT0                              | 0Ch     | 30h | 0C | 5   |      |
| INT1                              | 0Dh     | 34h | 0D | 6   |      |
| INT2                              | 0Eh     | 38h | 0E | 7   |      |
| INT3                              | 0Fh     | 3Ch | 0F | 8   |      |
| INT4                              | 10h     | 40h | 10 | 9   |      |
| Asynchronous Serial port 1        | 11h     | 44h | 11 | 9   |      |
| Timer 1                           | 12h     | 48h | 08 | 2-2 | */** |
| Timer 2                           | 13h     | 4Ch | 08 | 2-3 | */** |
| Asynchronous Serial port 0        | 14h     | 50h | 14 | 9   |      |
| Reserved                          | 15h-1Fh |     |    |     |      |

Note \* : When the interrupt occurs in the same time, the priority is (1-1 > 1-2) ; (2-1 > 2-2 > 2-3)

Note \*\*: The interrupt types of these sources are programmable in slave mode.

### 13.3 Interrupt Request

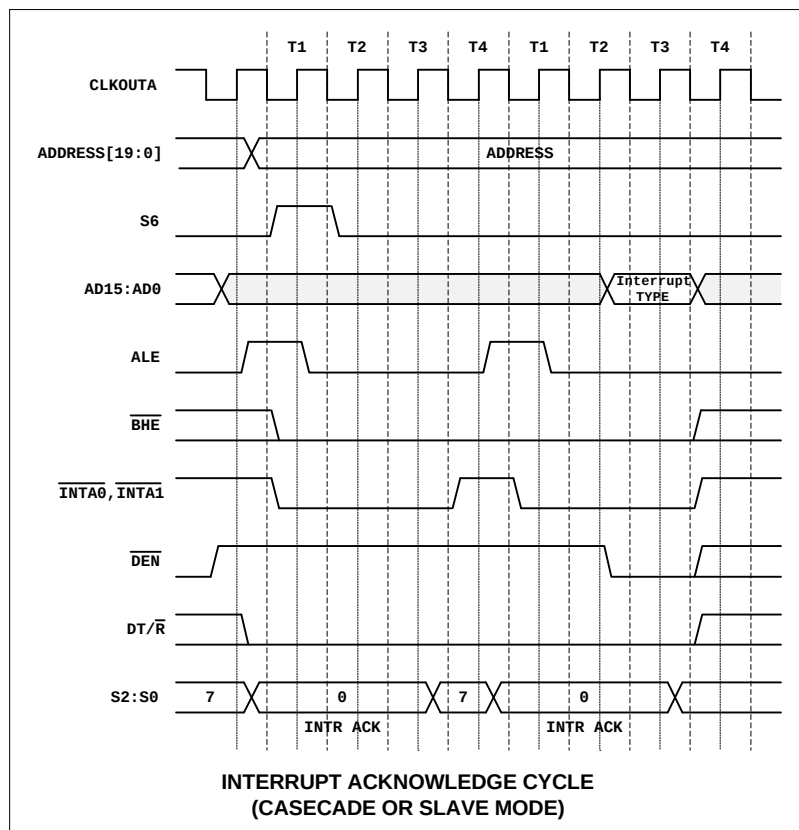
When an interrupt is request, the internal interrupt controller verifies the interrupt is enable (The IF flag is enable, no MSK bit set ) and that there are no higher priority interrupt requests being serviced or pending. If the interrupt is granted , the interrupt controller uses the interrupt type to access a vector from the interrupt vector table.

If the external INT is active (level-trigger) to request the interrupt controller service, and the INT pins must hold till the microcontroller enter the interrupt service routine. There is no interrupt-acknowledge output when running in fully nested mode, so it should use PIO pin to simulate the interrupt-acknowledge pin if necessary.

### 13.4 Interrupt Acknowledge

The processor requires the interrupt type as an index into the interrupt table. The internal interrupt can provide the interrupt type or an external controller can provide the interrupt type.

The internal interrupt controller provides the interrupt type to processor without external bus cycles generation. When an external interrupt controller is supplying the interrupt type, the processor generates two acknowledge bus cycles, and the interrupt type is written to the AD7-AD0 lines by the external interrupt controller.



### 13.5 Programming the Registers

Software is programmed through the registers ( **Master mode:** 44h, 42h, 40h, 3Eh, 3Ch, 3Ah, 38h, 36h, 34h, 32h, 30h, 2Eh, 2Ch, 2Ah, 28h, 26h, 24h, 22h; **Slave Mode:** 3Ah, 38h, 36h, 34h, 32h, 30h, 2Eh, 2Ch, 2Ah, 28h, 26h, 24h, 22h, 20h ) to define the interrupt controller operation.

#### Serial Port 0 Interrupt Control Register

Offset : 44h  
Reset Value : 000Fh

|    |    |    |    |    |    |   |   |   |   |   |   |     |   |     |     |
|----|----|----|----|----|----|---|---|---|---|---|---|-----|---|-----|-----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3   | 2 | 1   | 0   |
|    |    |    |    |    |    |   |   |   |   |   | 1 | MSK |   | PR1 | PR0 |

#### (Master Mode)

**Bit 15-4 :** Reserved

**Bit 3: MSK,** Mask.

Set 1: Mask the interrupt source of the asynchronous serial port 0.

Set 0: Enable the serial port 0 interrupt.

**Bit 2-0 : PR2-PR0,** Priority. These bits determine the priority of the serial port relative to the other interrupt signals.

**The priority selection:**

**PR2, PR1, PR0 -- Priority**

|   |   |   |   |   |    |   |         |
|---|---|---|---|---|----|---|---------|
| 0 | , | 0 | , | 0 | -- | 0 | ( High) |
| 0 | , | 0 | , | 1 | -- | 1 |         |
| 0 | , | 1 | , | 0 | -- | 2 |         |
| 0 | , | 1 | , | 1 | -- | 3 |         |
| 1 | , | 0 | , | 0 | -- | 4 |         |
| 1 | , | 0 | , | 1 | -- | 5 |         |
| 1 | , | 1 | , | 0 | -- | 6 |         |
| 1 | , | 1 | , | 1 | -- | 7 | ( Low ) |

### Serial Port 1 Interrupt Control Register

Offset : 42h

Reset Value : 000Fh

|    |    |    |    |   |   |   |   |     |   |         |
|----|----|----|----|---|---|---|---|-----|---|---------|
| 15 | 14 | 12 | 11 | 9 | 8 | 6 | 5 | 3   | 2 | 0       |
|    |    |    |    |   |   |   | 1 | MSK |   | PR1 PR0 |

**(Master Mode)**

**Bit 15-4 :** Reserved

**Bit 3: MSK,** Mask.

Set 1: Mask the interrupt source of the asynchronous serial port 1.

Set 0: Enable the serial port 1 interrupt.

**Bit 2-0 : PR2-PR0,** Priority. These bits determine the priority of the serial port relative to the other interrupt signals.

**The priority selection:**

**PR2, PR1, PR0 -- Priority**

|   |   |   |   |   |    |   |         |
|---|---|---|---|---|----|---|---------|
| 0 | , | 0 | , | 0 | -- | 0 | ( High) |
| 0 | , | 0 | , | 1 | -- | 1 |         |
| 0 | , | 1 | , | 0 | -- | 2 |         |
| 0 | , | 1 | , | 1 | -- | 3 |         |
| 1 | , | 0 | , | 0 | -- | 4 |         |
| 1 | , | 0 | , | 1 | -- | 5 |         |
| 1 | , | 1 | , | 0 | -- | 6 |         |
| 1 | , | 1 | , | 1 | -- | 7 | ( Low ) |

**INT4 Control Register**
**Offset : 40h**
**Reset Value : 000Fh**

|             |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
|-------------|----|--|----|----|--|---|---|--|---|---|--|---|---|--|---|
| 15          | 14 |  | 12 | 11 |  | 9 | 8 |  | 6 | 5 |  | 3 | 2 |  | 0 |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |
| <div></div> |    |  |    |    |  |   |   |  |   |   |  |   |   |  |   |

**(Master Mode)**
**Bit 15- 8, bit 6-5 :** Reserved

**Bit 7: ETM**, Edge trigger enable. When this bit set to 1 and Bit 4 set to 0, interrupt is triggered by low go high edge.

The low go high edge will be latched (one level) till this interrupt is been serviced.

**Bit 4: LTM**, Level-Triggered Mode.

Set 1: Interrupt is triggered by high active level

Set 0 : Interrupt is triggered by low go high edge.

**Bit 3 : MSK**, Mask.

Set 1: Mask the interrupt source of the INT4

Set 0: Enable the INT4 interrupt.

**Bit 2-0: PR**, Interrupt Priority

These bits setting for priority selection is same as bit 2-0 of 44h

**INT3 Control Register**
**Offset : 3Eh**
**Reset Value : 000Fh**

|    |    |    |    |    |    |   |   |     |   |   |     |     |     |     |     |
|----|----|----|----|----|----|---|---|-----|---|---|-----|-----|-----|-----|-----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7   | 6 | 5 | 4   | 3   | 2   | 1   | 0   |
|    |    |    |    |    |    |   |   | ETM |   |   | LTM | MSK | PR2 | PR1 | PR0 |

**(Master Mode)**
**Bit 15- 8, bit 6- 5 :** Reserved

**Bit 7: ETM**, Edge trigger enable. When this bit set to 1 and Bit 4 set to 0, interrupt is triggered by low go high edge.

The low go high edge will be latched (one level) till this interrupt is been serviced.

**Bit 4: LTM**, Level-Triggered Mode.

Set 1: Interrupt is triggered by high active level

Set 0 : Interrupt is triggered by low go high edge.

**Bit 3 : MSK**, Mask.

Set 1: Mask the interrupt source of the INT3

Set 0: Enable the INT3 interrupt.

**Bit 2-0: PR**, Interrupt Priority

These bits setting for priority selection is same as bit 2-0 of 44h

**INT2 Control Register**
**Offset : 3Ch**
**Reset Value : 000Fh**

| 15       | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7   | 6 | 5 | 4   | 3   | 2   | 1   | 0   |
|----------|----|----|----|----|----|---|---|-----|---|---|-----|-----|-----|-----|-----|
| Reserved |    |    |    |    |    |   |   | ETM |   |   | LTM | MSK | PR2 | PR1 | PR0 |

**(Master Mode)**
**Bit 15- 8, bit 6-5 :** Reserved

**Bit 7: ETM**, Edge trigger enable. When this bit set to 1 and Bit 4 set to 0, interrupt is triggered by low go high edge.

The low go high edge will be latched (one level) till this interrupt is been serviced.

**Bit 4: LTM**, Level-Triggered Mode.

Set 1: Interrupt is triggered by high active level

Set 0 : Interrupt is triggered by low go high edge.

**Bit 3 : MSK**, Mask.

Set 1: Mask the interrupt source of the INT2

Set 0: Enable the INT2 interrupt.

**Bit 2-0: PR**, Interrupt Priority

These bits setting for priority selection is same as bit 2-0 of the register 44h

**INT1 Control Register**
**Offset : 3Ah**
**Reset Value : 000Fh**

| 15       | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7   | 6    | 5 | 4   | 3   | 2   | 1   | 0   |
|----------|----|----|----|----|----|---|---|-----|------|---|-----|-----|-----|-----|-----|
| Reserved |    |    |    |    |    |   |   | ETM | SFNM | C | LTM | MSK | PR2 | PR1 | PR0 |

**(Master Mode)**
**Bit 15-8 :** Reserved

**Bit 7: ETM**, Edge trigger enable. When this bit set to 1 and Bit 4 set to 0, interrupt is triggered by low go high edge.

The low go high edge will be latched (one level) till this interrupt is been serviced.

**Bit 6: SFNM**, Special Fully Nested Mode.

Set 1: Enable the special fully nested mode of INT1

**Bit 5: C**, Cascade Mode. Set this bit to 1 to enable the cascade mode for INT1 or INT0.

**Bit 4: LTM**, Level-Triggered Mode.

Set 1: Interrupt is triggered by high active level

Set 0 : Interrupt is triggered by low go high edge.

**Bit 3 : MSK**, Mask.

Set 1: Mask the interrupt source of the INT1

Set 0: Enable the INT1 interrupt.

**Bit 2-0: PR, Interrupt Priority**

These bits setting for priority selection is same as bit 2-0 of the register 44h

**(Slave Mode) , This register is for timer 2 interrupt control,** reset value is 0000h

**Bit 15- 4 : Reserved**
**Bit 3 : MSK, Mask.**

Set 1: Mask the interrupt source of the Timer 2

Set 0: Enable the Timer 2 interrupt.

**Bit 2-0: PR, Interrupt Priority**

These bits setting for priority selection is same as bit 2-0 of the register 44h

**INT0 Control Register**
**Offset : 38h**
**Reset Value : 000Fh**

| 15       | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7   | 6    | 5 | 4   | 3   | 2   | 1   | 0   |
|----------|----|----|----|----|----|---|---|-----|------|---|-----|-----|-----|-----|-----|
| Reserved |    |    |    |    |    |   |   | ETM | SFNM | C | LTM | MSK | PR2 | PR1 | PR0 |

**(Master Mode)**
**Bit 15-8 : Reserved**

**Bit 7: ETM, Edge trigger enable.** When this bit set to 1 and Bit 4 set to 0, interrupt is triggered by low go high edge.

The low go high edge will be latched (one level) till this interrupt is been serviced.

**Bit 6: SFNM, Special Fully Nested Mode.**

Set 1: Enable the special fully nested mode of INT0.

**Bit 5: C, Cascade Mode.** Set this bit to 1 to enable the cascade mode for INT1 or INT0.

**Bit 4: LTM, Level-Triggered Mode.**

Set 1: Interrupt is triggered by high active level

Set 0 : Interrupt is triggered by low go high edge.

**Bit 3 : MSK, Mask.**

Set 1: Mask the interrupt source of the INT0

Set 0: Enable the INT0 interrupt.

**Bit 2-0: PR, Interrupt Priority**

These bits setting for priority selection is same as bit 2-0 of the register 44h

**(Slave Mode),For Timer 1 interrupt control register,** reset value is 0000h

**Bit 15-4 : Reserved**
**Bit 3: MSK , Mask.**

Set 1: Mask the interrupt source of the timer 1

Set 0: Enable the timer 1 interrupt.

**Bit 2-0: PR, Interrupt Priority**

These bits setting for priority selection is same as bit 2-0 of the register 44h

**DMA 1/INT6 Interrupt Control Register**

Offset : 36h

Reset Value : 000Fh

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3   | 2   | 1   | 0   |
|----|----|----|----|----|----|---|---|---|---|---|---|-----|-----|-----|-----|
| 0  | 0  | 0  | 0  | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0 | MSK | PR2 | PR1 | PR0 |

**(Master Mode)**

**Bit 15-4 :** Reserved

**Bit 3: MSK ,** Mask.

Set 1: Mask the interrupt source of the DMA 1 controller

Set 0: Enable the DMA 1 controller interrupt.

**Bit 2-0: PR, Interrupt Priority**

These bits setting for priority selection is same as bit 2-0 of the register 44h

**(Slave Mode),** reset value is 0000h

**Bit 15-4 :** Reserved

**Bit 3: MSK ,** Mask.

Set 1: Mask the interrupt source of the DMA 1 controller

Set 0: Enable the DMA 1 controller interrupt.

**Bit 2-0: PR, Interrupt Priority**

These bits setting for priority selection is same as bit 2-0 of the register 44h

**DMA 0/INT5 Interrupt Control Register**

Offset : 34h

Reset Value : 000Fh

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3   | 2   | 1   | 0   |
|----|----|----|----|----|----|---|---|---|---|---|---|-----|-----|-----|-----|
| 0  | 0  | 0  | 0  | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0 | MSK | PR2 | PR1 | PR0 |

**(Master Mode)**

**Bit 15-4 :** Reserved

**Bit 3: MSK ,** Mask.

Set 1: Mask the interrupt source of the DMA 0 controller

Set 0: Enable the DMA 0 controller interrupt.

**Bit 2-0: PR**, Interrupt Priority

These bits setting for priority selection is same as bit 2-0 of the register 44h

**(Slave Mode)**, reset value is 0000h

**Bit 15-4** : Reserved

**Bit 3: MSK** , Mask.

Set 1: Mask the interrupt source of the DMA 0 controller

Set 0: Enable the DMA 1 controller interrupt.

**Bit 2-0: PR**, Interrupt Priority

These bits setting for priority selection is same as bit 2-0 of the register 44h

**Timer Interrupt Control Register**

Offset : 32h

Reset Value : 000Fh

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3   | 2   | 1   | 0   |
|----|----|----|----|----|----|---|---|---|---|---|---|-----|-----|-----|-----|
| 0  | 0  | 0  | 0  | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0 | MSK | PR2 | PR1 | PR0 |

**(Master Mode)**

**Bit 15-4** : Reserved

**Bit 3: MSK** , Mask.

Set 1: Mask the interrupt source of the timer controller

Set 0: Enable the timer controller interrupt.

**Bit 2-0: PR**, Interrupt Priority

These bits setting for priority selection is same as bit 2-0 of the register 44h

**(Slave Mode)**, reset value is 0000h

**Bit 15-4** : Reserved

**Bit 3: MSK** , Mask.

Set 1: Mask the interrupt source of the timer 0 controller

Set 0: Enable the timer 0 controller interrupt.

**Bit 2-0: PR**, Interrupt Priority

These bits setting for priority selection is same as bit 2-0 of the register 44h

**Interrupt Status Register**

Offset : 30h

Reset Value : —

|      |    |    |    |    |    |   |   |   |   |   |   |   |      |      |      |
|------|----|----|----|----|----|---|---|---|---|---|---|---|------|------|------|
| 15   | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2    | 1    | 0    |
| DHLT |    |    |    |    |    |   |   |   |   |   |   |   | TMR2 | TMR1 | TMR0 |

**(Master Mode)**, Reset value undefine

**Bit 15 : DHLT**, DMA Halt.

Set 1: halts any DMA activity. When non-maskable interrupts occur.

Set 0: When an IRET instruction is executed.

**Bit 14-3 : Reserved**.

**Bit 2-0 : TMR2-TMR0**,

Set 1: indicates the corresponding timer has an interrupt request pending.

**(Slave Mode)**, Reset value is 0000h

**Bit 15 : DHLT**, DMA Halt.

Set 1: halts any DMA activity. When non-maskable interrupts occur.

Set 0: When an IRET instruction is executed.

**Bit 14-3 : Reserved**.

**Bit 2-0 : TMR2-TMR0**,

Set 1: indicates the corresponding timer has an interrupt request pending.

**Interrupt Request Register**

Offset : 2Eh

Reset Value : —

|    |    |    |    |    |     |     |    |    |    |    |    |       |       |     |     |
|----|----|----|----|----|-----|-----|----|----|----|----|----|-------|-------|-----|-----|
| 15 | 14 | 13 | 12 | 11 | 10  | 9   | 8  | 7  | 6  | 5  | 4  | 3     | 2     | 1   | 0   |
|    |    |    |    |    | SP0 | SP1 | I4 | I3 | I2 | I1 | I0 | D1/I6 | D0/I5 | Res | TMR |

**(Master Mode)**

The Interrupt Request register is a read-only register. For internal interrupts (SP0, SP1, D1/I6, D0/I5, and TMR), the corresponding bit is set to 1 when the device requests an interrupt. The bit is reset during the internally generated interrupt acknowledge. For INT4-INT0 external interrupts, the corresponding bit (I4-I0) reflects the current value of the external signal.

**Bit 15-11 : Reserved**.

**Bit 10 : SP0**, Serial Port 0 Interrupt Request. Indicates the interrupt state of the serial port 0.

**Bit 9 : SP1**, Serial Port 1 Interrupt Request. Indicates the interrupt state of the serial port 1.

**Bit 8-4 : I4-I0**, Interrupt Requests.

Set 1: The corresponding INT pin has an interrupt pending.

**Bit 3-2 : D1/I6-D0/I5**, DMA Channel or INT Interrupt Request.

Set 1: The corresponding DMA channel or INT has an interrupt pending.

**Bit 1:** Reserved.

**Bit 0 : TMR**, Timer Interrupt Request.

Set 1: The timer control unit has an interrupt pending.

### Interrupt Request Register

Offset : 2Eh

Reset Value : 0000h

| 15       | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5    | 4    | 3     | 2     | 1   | 0    |
|----------|----|----|----|----|----|---|---|---|---|------|------|-------|-------|-----|------|
| Reserved |    |    |    |    |    |   |   |   |   | TMR2 | TMR1 | D1/I6 | D0/I5 | Res | TMR0 |

### (Slave Mode)

The Interrupt Request register is a read-only register. For internal interrupts (D1/I6, D0/I5, TMR2, TMR1, and TMR0), the corresponding bit is set to 1 when the device requests an interrupt. The bit is reset during the internally generated interrupt acknowledge.

**Bit 15-6 :** Reserved.

**Bit 5-4 : TMR2/TMR1**, Timer2/Timer1 Interrupt Request.

Set 1: Indicates the state of any interrupt requests from the associated timer.

**Bit 3-2 : D1/I6-D0/I5**, DMA Channel or INT Interrupt Request.

Set 1: Indicates the corresponding DMA channel or INT has an interrupt pending.

**Bit 1 :** Reserved.

**Bit 0 : TMR0**, Timer 0 Interrupt Request.

Set 1: Indicates the state of an interrupt request from Timer 0.

### In - Service Register

Offset : 2Ch

Reset Value : 0000h

| 15       | 14 | 13 | 12 | 11 | 10  | 9   | 8  | 7  | 6  | 5  | 4  | 3     | 2     | 1   | 0   |
|----------|----|----|----|----|-----|-----|----|----|----|----|----|-------|-------|-----|-----|
| Reserved |    |    |    |    | SP0 | SP1 | I4 | I3 | I2 | I1 | I0 | D1/I6 | D0/I5 | Res | TMR |

### (Master Mode)

The bits in the INSERV register are set by the interrupt controller when the interrupt is taken. Each bit in the register is cleared by writing the corresponding interrupt type to the EOI register.

**Bit 15-11 :** Reserved.

**Bit 10 : SP0**, Serial Port 0 Interrupt In-Service.

Set 1: the serial port 0 interrupt is currently being serviced.

**Bit 9 : SP1**, Serial Port 1 Interrupt In-Service.

Set 1: the serial port 1 interrupt is currently being serviced.

**Bit 8-4 : I4-I0**, Interrupt In-Service.

Set 1: the corresponding INT interrupt is currently being serviced.

**Bit 3-2 : D1/I6-D0/I5**, DMA Channel or INT Interrupt In-Service.

Set 1: the corresponding DMA channel or INT interrupt is currently being serviced.

**Bit 1** : Reserved.

**Bit 0 : TMR**, Timer Interrupt In-Service.

Set 1: the timer interrupt is currently being serviced.

### In - Service Register

Offset : 2Ch

Reset Value : 0000h

| 15       | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5    | 4    | 3  | 2  | 1   | 0    |
|----------|----|----|----|----|----|---|---|---|---|------|------|----|----|-----|------|
| Reserved |    |    |    |    |    |   |   |   |   | TMR2 | TMR1 | D1 | D0 | Res | TMR0 |

### (Slave Mode)

The bits in the In-Service register are set by the interrupt controller when the interrupt is taken. The in-service bits are cleared by writing to the EOI register.

**Bit 15-6** : Reserved.

**Bit 5-4 : TMR2-TMR1**, Timer2/Timer1 Interrupt In-Service.

Set 1: the corresponding timer interrupt is currently being serviced.

**Bit 3-2 : D1/I6-D0/I5**, DMA Channel or INT Interrupt In-Service.

Set 1: the corresponding DMA Channel or INT Interrupt is currently being serviced.

**Bit 1** : Reserved.

**Bit 0 : TMR0**, Timer 0 Interrupt In-Service.

Set 1: the Timer 0 interrupt is currently being serviced.

### Priority Mask Register

Offset : 2Ah

Reset Value : 0007h

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2    | 1    | 0    |
|----|----|----|----|----|----|---|---|---|---|---|---|---|------|------|------|
| 0  | 0  | 0  | 0  | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | PRM2 | PRM1 | PRM0 |

### (Master Mode)

Determining the minimum priority level at which maskable interrupts can generate an interrupt.

**Bit 15-3** : Reserved.

**Bit 2-0 : PRM2-PRM0**, Priority Field Mask. Determining the minimum priority that is required in order for a maskable interrupt source to generate an interrupt.

| Priority | PR2-PR0 |
|----------|---------|
| (High) 0 | 000     |
| 1        | 001     |
| 2        | 010     |
| 3        | 011     |
| 4        | 100     |
| 5        | 101     |
| 6        | 110     |
| (Low) 7  | 111     |

### (Slave Mode)

Determining the minimum priority level at which maskable interrupts can generate an interrupt.

**Bit 15-3** : Reserved.

**Bit 2-0 : PRM2-PRM0**, Priority Field Mask. Determining the minimum priority that is required in order for a maskable interrupt source to generate an interrupt.

| Priority | PR2-PR0 |
|----------|---------|
| (High) 0 | 000     |
| 1        | 001     |
| 2        | 010     |
| 3        | 011     |
| 4        | 100     |
| 5        | 101     |
| 6        | 110     |
| (Low) 7  | 111     |

### Interrupt Mask Register

Offset : 28h

Reset Value : 07FDh

| 15       | 14 | 13 | 12 | 11 | 10  | 9   | 8  | 7  | 6  | 5  | 4  | 3     | 2     | 1   | 0   |
|----------|----|----|----|----|-----|-----|----|----|----|----|----|-------|-------|-----|-----|
| Reserved |    |    |    |    | SP0 | SP1 | I4 | I3 | I2 | I1 | I0 | D1/I6 | D0/I5 | Res | TMR |

### (Master Mode)

**Bit 15-11** : Reserved.

**Bit 10 : SP0**, Serial Port 0 Interrupt Mask. The state of the mask bit of the asynchronous serial port 0 interrupt.

**Bit 9 : SP1**, Serial Port 1 Interrupt Mask. The state of the mask bit of the asynchronous serial port 1 interrupt.

**Bit 8-4 : I4-I0**, Interrupt Masks. Indicates the state of the mask bit of the corresponding interrupt.

**Bit 3-2 : D1/I6-D0/I5**, DMA Channel or INT Interrupt Masks.

Indicates the state of the mask bit of the corresponding DMA Channel or INT interrupt.

**Bit 1**: Reserved.

**Bit 0 : TMR**, Timer Interrupt Mask. The state of the mask bit of the timer control unit .

**Interrupt Request Register**
**Offset : 28h**
**Reset Value : 003Dh**

|    |    |    |    |    |    |   |   |   |   |      |      |       |       |     |      |
|----|----|----|----|----|----|---|---|---|---|------|------|-------|-------|-----|------|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5    | 4    | 3     | 2     | 1   | 0    |
|    |    |    |    |    |    |   |   |   |   | TMR2 | TMR1 | D1/I6 | D0/I5 | Res | TMR0 |

**(Slave Mode)**
**Bit 15-6 :** Reserved.

**Bit 5-4 :** **TMR2-TMR1**, Timer 2/Timer1 Interrupt Mask. The state of the mask bit of the Timer Interrupt Control register.

Set 1: Timer2 or Time1 has its interrupt requests masked

**Bit 3-2 :** **D1/I6-D0/I5**, DMA Channel or INT Interrupt Mask.

Indicating the state of the mask bits of the corresponding DMA or INT6/INT5 control register.

**Bit 1 :** Reserved.

**Bit 0 :** **TMR0**, Timer 0 Interrupt Mask. The state of the mask bit of the Timer Interrupt Control Register

**Poll Status Register**
**Offset : 26h**
**Reset Value : —**

|      |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15   | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IREQ |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

**(Master Mode)**

The Poll Status (POLLST) register mirrors the current state of the Poll register. the POLLST register can be read without affecting the current interrupt request.

**Bit 15 :** **IREQ**, Interrupt Request.

Set 1: if an interrupt is pending. The S4-S0 field contains valid data.

**Bit 14-5 :** Reserved.

**Bit 4-0 :** **S4-S0**, Poll Status. Indicates the interrupt type of the highest priority pending interrupt.

**Poll Register**
**Offset : 24h**
**Reset Value : —**

|      |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15   | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IREQ |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

**(Master Mode)**

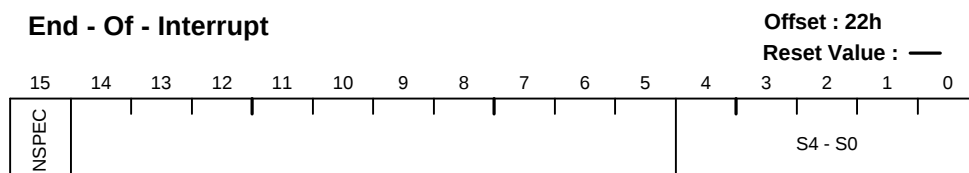
When the Poll register is read, the current interrupt is acknowledged and the next interrupt takes its place in the Poll register.

**Bit 15 :** **IREQ**, Interrupt Request.

Set 1: if an interrupt is pending. The S4-S0 field contains valid data.

**Bit 14-5 : Reserved.**

**Bit 4-0 : S4-S0, Poll Status.** Indicates the interrupt type of the highest priority pending interrupt.



**(Master Mode)**

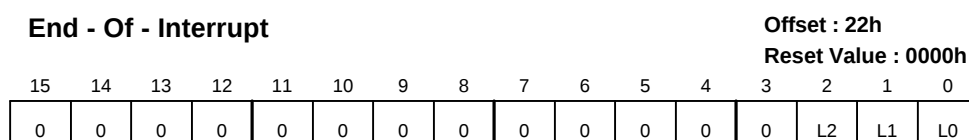
**Bit 15 : NSPEC, Non-Specific EOI.**

Set 1: indicates non-specific EOI.

Set 0: indicates the specific EOI interrupt type in S4-S0.

**Bit 14-5 : Reserved.**

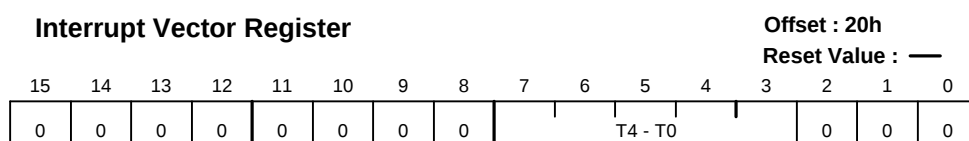
**Bit 4-0: S4-S0, Source EOI Type.** Specifies the EOI type of the interrupt that is currently being processed.



**(Slave Mode)**

**Bit 15-3 : Reserved.**

**Bit 2-0 : L2-L0, Interrupt Type.** Encoded value indicating the priority of the IS(interrupt service) bit to reset. Writes to these bits cause an EOI to be issued for the interrupt type in slave mode.



**(Slave Mode)**

**Bit 15-8 : Reserved**

**Bit 7-3 : T4-T0, Interrupt Type.**

The following interrupt type of slave mode can be programmed.

Timer 2 interrupt controller : (T4,T3,T2,T1,T0, 1, 0, 1)b

Timer 1 interrupt controller : (T4,T3,T2,T1,T0, 1, 0, 0)b

DMA 1 interrupt controller : (T4,T3,T2,T1,T0, 0, 1, 1)b

DMA 0 interrupt controller : (T4,T3,T2,T1,T0, 0, 1, 0)b

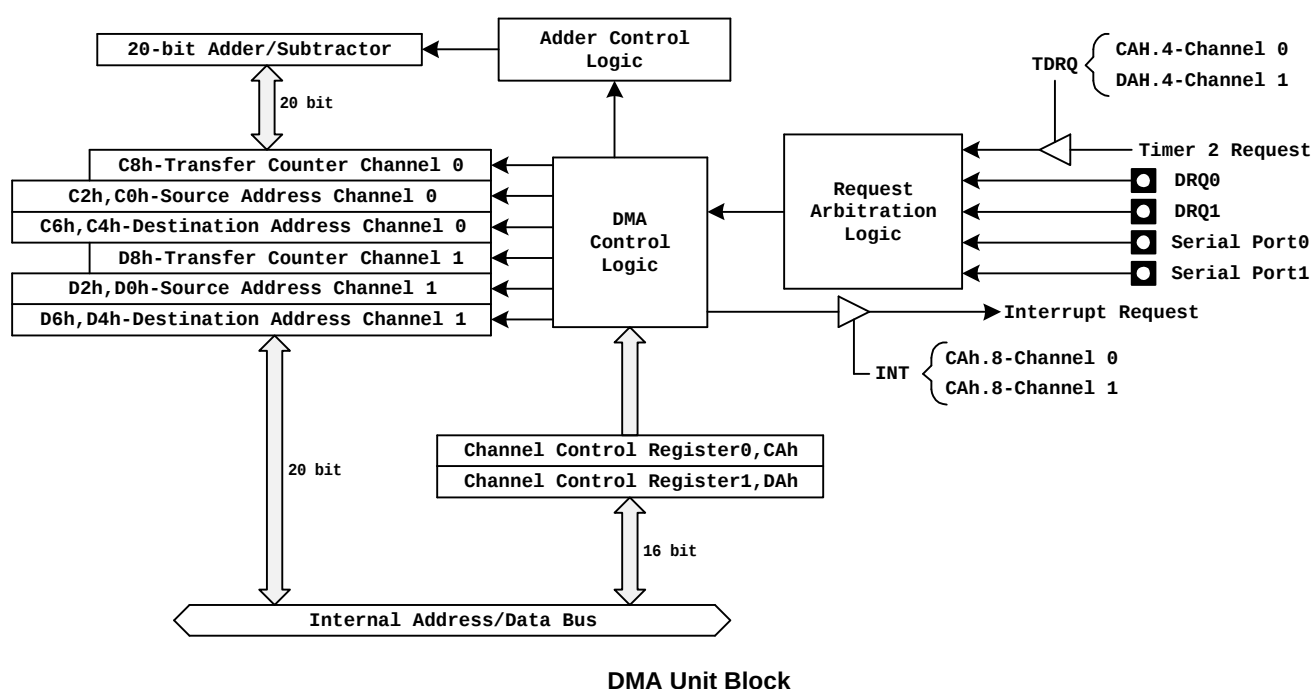
Timer 0 interrupt controller : (T4,T3,T2,T1,T0, 0, 0, 0)b

**Bit 2-0** :Reserved

## 14. DMA Unit

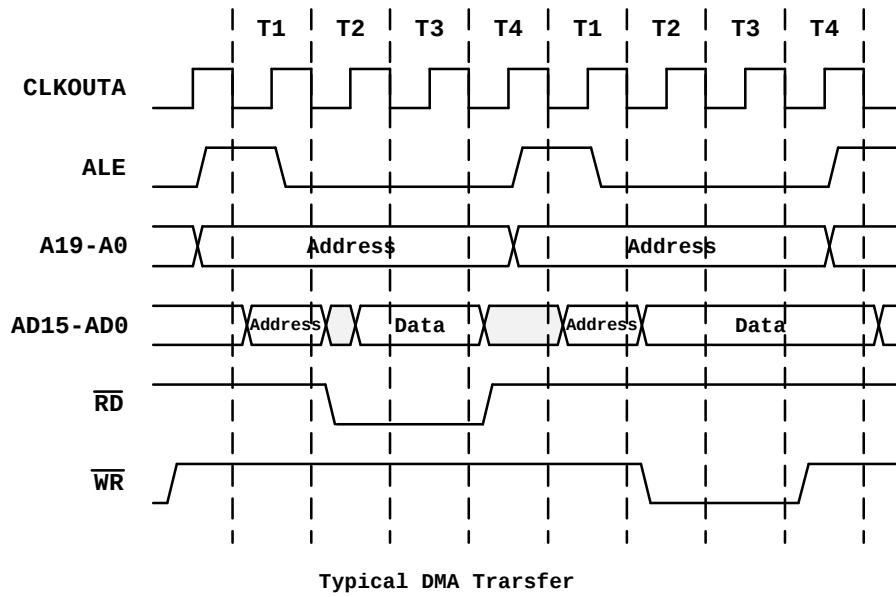
The DMA controller provides the data transfer between the memory and peripherals without the intervention of the CPU.

There are two DMA channels in the DMA unit. Each channel can accept DMA request from one of three sources: external pin (DRQ0 for channel 0 or DRQ1 for channel 1) or serial port (port 0 or port 1) or Timer 2 overflow. The data transfer from source to destination can be memory to memory, or memory to I/O, or I/O to I/O, or I/O to memory. Either bytes or words can be transferred to or from even or odd addresses and two bus cycles are necessary (read from source and write to destination) for each data transfer.



### 14.1 DMA Operation

Every DMA transfer consists of two bus cycles (figure of Typical DMA Transfer) and the two bus cycles can not be separated by a bus hold request, a refresh request or another DMA request. The registers ( CAh, C8h, C6h, C4h, C2h, C0h, DAh, D8h, D6h, D4h, D2h, D0h) are used to configure and operate the two DMA channels.



### DMA Control Registers

Offset : CAh (DMA0)

Reset Value : FFF9h

| 15                  | 14   | 13   | 12                  | 11   | 10   | 9  | 8   | 7    | 6    | 5 | 4    | 3   | 2   | 1  | 0   |
|---------------------|------|------|---------------------|------|------|----|-----|------|------|---|------|-----|-----|----|-----|
| DM/I $\overline{O}$ | DDEC | DINC | SM/I $\overline{O}$ | SDEC | SINC | TC | INT | SYN1 | SYN0 | P | TDRQ | Res | CHG | ST | B/W |

The definition of Bits 15-0 for DMA0 are same as the Bits 15-0 of register DAh for DMA1.

### DMA Transfer Count Register

Offset : C8h (DMA0)

Reset Value : —

| 15         | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| TC15 - TC0 |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

**Bit 15-0:** TC15-TC0, DMA 0 transfer Count. The value of this register is decremented by 1 after each transfer.

### DMA Destination Address High Register

Offset : C6h (DMA0)

Reset Value : —

| 15       | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3             | 2 | 1 | 0 |
|----------|----|----|----|----|----|---|---|---|---|---|---|---------------|---|---|---|
| Reserved |    |    |    |    |    |   |   |   |   |   |   | DDA19 - DDA16 |   |   |   |

**Bit 15-4:** Reserved

**Bit 3-0: DDA19-DDA16**, High DMA 0 Destination Address. These bits are map to A19- A16 during a DMA transfer when the destination address is in memory space or I/O space. If the destination address is in I/O space (64Kbytes), these bits must be programmed to 0000b.

**DMA Destination Address Low Register**
**Offset : C4h (DMA0)**
**Reset Value : —**

|              |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|--------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15           | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| DDA15 - DDA0 |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

**Bit 15-0: DDA15-DDA0**, Low DMA 0 Destination Address. These bits are mapped to A15- A0 during a DMA transfer. The value of (DDA19-DDA0)b will increment or decrement by 2 after each DMA transfer.

**DMA Source Address High Register**
**Offset : C2h (DMA0)**
**Reset Value : —**

|    |    |    |    |    |    |   |   |   |   |   |   |               |   |   |   |
|----|----|----|----|----|----|---|---|---|---|---|---|---------------|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3             | 2 | 1 | 0 |
|    |    |    |    |    |    |   |   |   |   |   |   | DSA19 - DSA16 |   |   |   |

**Bit 15-4:** Reserved

**Bit 3-0: DSA19-DSA16**, High DMA 0 Source Address. These bits are mapped to A19- A16 during a DMA transfer when the source address is in memory space or I/O space. If the source address is in I/O space (64Kbytes), these bits must be programmed to 0000b.

**DMA Source Address Low Register**
**Offset : C0h (DMA0)**
**Reset Value : —**

|              |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|--------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15           | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| DSA15 - DSA0 |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

**Bit 15-0: DSA15-DSA0**, Low DMA 0 Source Address. These bits are mapped to A15- A0 during a DMA transfer. The value of (DSA19-DSA0)b will increment or decrement by 2 after each DMA transfer.

**DMA Control Registers**
**Offset : DAh (DMA1)**
**Reset Value : FFF9h**

|                     |      |      |                     |      |      |    |     |      |      |   |      |     |     |    |                  |
|---------------------|------|------|---------------------|------|------|----|-----|------|------|---|------|-----|-----|----|------------------|
| 15                  | 14   | 13   | 12                  | 11   | 10   | 9  | 8   | 7    | 6    | 5 | 4    | 3   | 2   | 1  | 0                |
| DM/I $\overline{O}$ | DDEC | DINC | SM/I $\overline{O}$ | SDEC | SINC | TC | INT | SYN1 | SYN0 | P | TDRQ | Res | CHG | ST | $\overline{B}/W$ |

**Bit 15:**  $\overline{\text{DM}} / \overline{\text{IO}}$ , Destination Address Space Select.

Set 1: The destination address is in memory space.

Set 0: The destination address is in I/O space.

**Bit 14:** **DDEC**, Destination Decrement.

Set 1: The destination address is automatically decrement after each transfer.

The  $\overline{\text{B}}/\text{W}$  (bit 0) bit determines the decrement value which is by 1 or 2 When both DDEC and DINC bits are set to 1, the address remains constant

Set 0 : Disable the decrement function.

**Bit 13:** **DINC**, Destination Increment.

Set 1: The destination address is automatically increment after each transfer.

The  $\overline{\text{B}}/\text{W}$  (bit 0) bit determines the increment value which is by 1 or 2

Set 0 : Disable the decrement function.

**Bit 12:**  $\text{SM} / \overline{\text{IO}}$ , Source Address Space Select.

Set 1: The Source address is in memory space.

Set 0: The Source address is in I/O space

**Bit 11:** **SDEC**, Source Decrement.

Set 1: The Source address is automatically decrement after each transfer.

The  $\overline{\text{B}}/\text{W}$  (bit 0) bit determines the decrement value which is by 1 or 2 When both SDEC and SINC bits are set to 1, the address remains constant

Set 0 : Disable the decrement function.

**Bit 10:** **SINC**, Source Increment.

Set 1: The Source address is automatically increment after each transfer.

The  $\overline{\text{B}}/\text{W}$  (bit 0) bit determines the increment value which is by 1 or 2

Set 0 : Disable the decrement function

**Bit 9 :** **TC**, Terminal Count.

Set 1: The synchronized DMA transfer is terminated when the DMA transfer count register reaches 0.

Set 0: The synchronized DMA transfer is terminated when the DMA transfer count register reaches 0.

Unsynchronized DMA transfer is always terminated when the DMA transfer count register reaches 0, regardless the setting of this bit.

**Bit 8 :** **INT**, Interrupt.

Set 1: DMA unit generates an interrupt request when complete the transfer count .

The TC bit must set to 1 to generate an interrupt.

**Bit 7-6:** **SYN1-SYN0**, Synchronization Type Selection.

**SYN1 , SYN0** -- **Synchronization Type**

0 , 0 -- Unsynchronized

0 , 1 -- Source synchronized

1 , 0 -- Destination synchronized

1 , 1 -- Reserved

**Bit 5: P**, Priority.

Set 1: It selects high priority for this channel when both DMA 0 and DMA 1 are transfer in same time.

**Bit 4: TDRQ**, Timer Enable/Disable Request

Set 1: Enable the DMA requests from timer 2.

Set 0: Disable the DMA requests from timer 2.

**Bit 3:** Reserved

**Bit 2: CHG**, Changed Start Bit. This bit must set to 1 when will modify the ST bit.

**Bit 1: ST**, Start/Stop DMA channel.

Set 1: Start the DMA channel

Set 0: Stop the DMA channel

**Bit 0 :  $\overline{B}/W$** , Byte/Word Select.

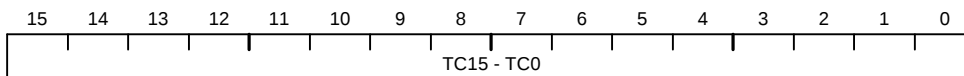
Set 1: The address is incremented or decremented by 2 after each transfer.

Set 0 :The address is incremented or decremented by 1 after each transfer.

#### DMA Transfer Count Register

Offset : D8h (DMA1)

Reset Value : —

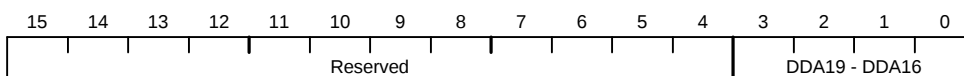


**Bit 15-0:** TC15-TC0, DMA 1 transfer Count. The value of this register is decremented by 1 after each transfer.

#### DMA Destination Address High Register

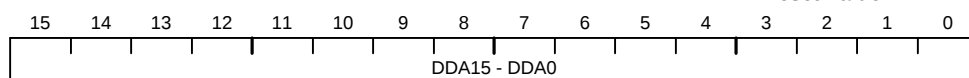
Offset : D6h (DMA1)

Reset Value : —



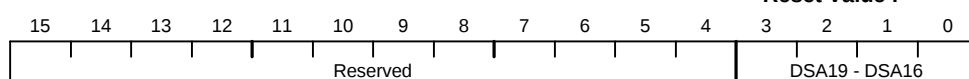
**Bit 15-4:** Reserved

**Bit 3-0:** **DDA19-DDA16**, High DMA 1 Destination Address. These bits are map to A19- A16 during a DMA transfer when the destination address is in memory space or I/O space. If the destination address is in I/O space (64Kbytes), these bits must be programmed to 0000b.

**DMA Destination Address Low Register**
**Offset : D4h (DMA1)**
**Reset Value : —**


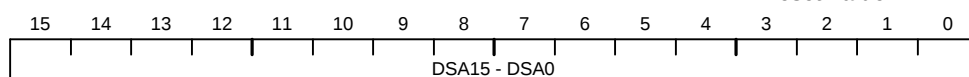
**Bit 15-0: DDA15-DDA0**, Low DMA 1 Destination Address. These bits are mapped to A15- A0 during a DMA transfer.

The value of (DDA19-DDA0)b will increment or decrement by 2 after each DMA transfer.

**DMA Source Address High Register**
**Offset : D2h (DMA1)**
**Reset Value : —**


**Bit 15-4:** Reserved

**Bit 3-0: DSA19-DSA16**, High DMA 1 Source Address. These bits are mapped to A19- A16 during a DMA transfer when the source address is in memory space or I/O space. If the source address is in I/O space (64Kbytes), these bits must be programmed to 0000b.

**DMA Source Address Low Register**
**Offset : D0h (DMA1)**
**Reset Value : —**


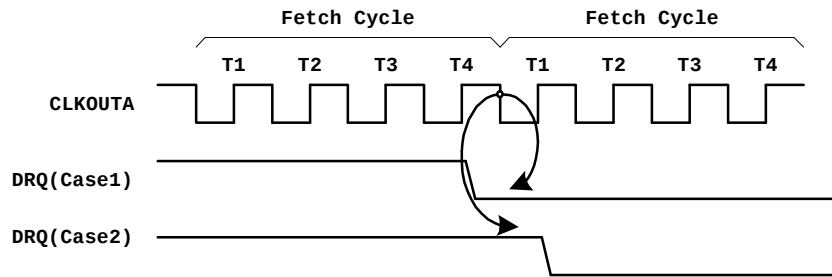
**Bit 15-0: DSA15-DSA0**, Low DMA 1 Source Address. These bits are map to A15- A0 during a DMA transfer.

The value of (DSA19-DSA0)b will increment or decrement by 2 after each DMA transfer.

## 14.2 External Requests

External DMA requests are asserted on the DRQ pins. The DRQ pins are sampled on the falling edge of CLKOUTA. It takes a minimum of four clocks before the DMA cycle is initiated by the Bus Interface. The DMA request is cleared four clocks before the end of the DMA cycle. And no DMA acknowledge is provided, since the chip-selects (MCSx, PCSx) can be programmed to be active for a given block of memory or I/O space, and the DMA source and destination address registers can be programmed to point to the same given block.

DMA transfer can be either source or destination synchronized, and it can also be unsynchronized. The Source-Synchronized Transfer figure shows the typical source-synchronized transfer which provides the source device at least three clock cycles from the time it is acknowledged to deassert its DRQ line.



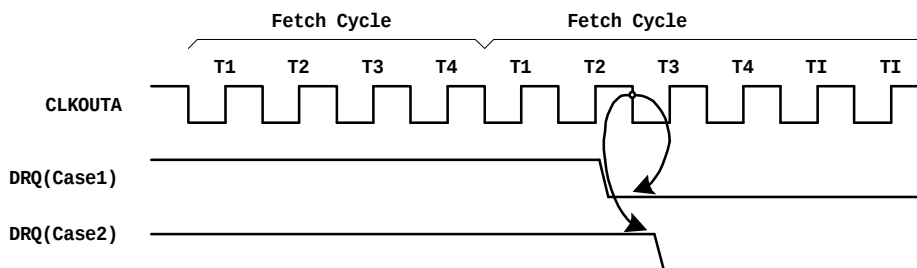
**NOTES:**

**Case1 :** Current source synchronized transfer will not be immediately followed by another DMA transfer.

**Case2 :** Current source synchronized transfer will be immediately followed by another DMA transfer.

### Source-Synchronized Transfers

The Destination-Synchronized Transfer figure shows the typical destination-synchronized transfer which differs from a source-synchronized transfer in that two idle states are added to the end of the deposit cycle. The two idle states extend the DMA cycle to allow the destination device to deassert its DRQ pin four clocks before the end of the cycle. If the two idle states were not inserted, the destination device would not have time to deassert its DRQ signal.



**NETES:**

**Case1 :** Current destination synchronized transfer will not be immediately followed by another DMA transfer.

**Case2 :** Current destination synchronized transfer will be immediately followed by another DMA transfer.

### Destination-Synchronized Transfers

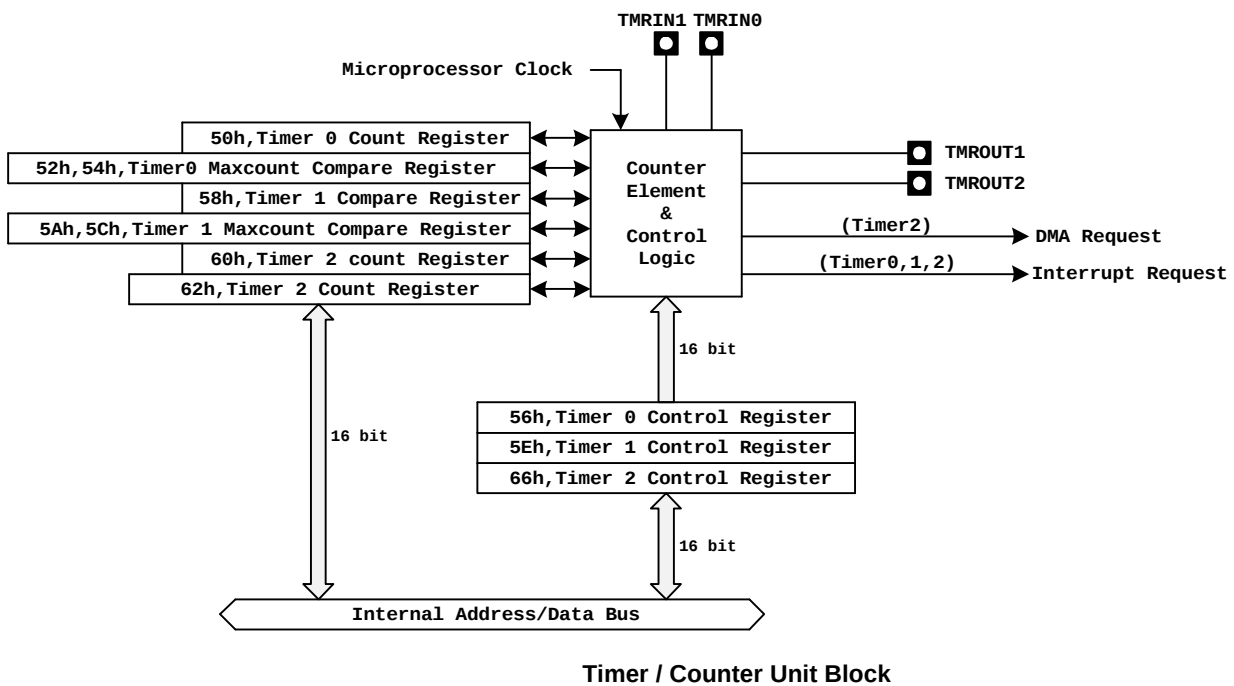
### 14.3 Serial Port/DMA Transfer

The serial port data can be DMA transfer to or from memory( or IO) space. And the  $\overline{B}/W$  bit of DMA control Register must be set 1 for byte transfer. The map address of Transmit Data Register is written to the DMA Destination Address Register and the memory (or I/O) address is written to the DMA Source Address Register, when transmit data. The map address of Receive Data Register is written to the DMA Source Address Register and the memory (or I/O) address is written to the DMA Destination Address Register, when receive data.

The software is programmed through the Serial Port Control Register to perform the serial port/ DMA transfer.

When a DMA channel is in use by a serial port, the corresponding external DMA request signal is deactivated. For DMA to the serial port, the DMA channel should be configured as destination synchronized. For DMA from the serial port, the DMA channel should be configured as source synchronized.

## 15. Timer Control Unit



There are three 16-bit programmable timers in the R8820LV. The timer operation is independent of the CPU. The three timers can be programmed as a timer element or as a counter element. Timers 0 and 1 are each connect to two external pins (TMRIN0, TMR0UT0, TMRIN1, TMR0UT1) which can be used to count or time external events, or they can be used to generate a variable-duty-cycle waveforms. Timer 2 is not connected any external pins. It can be used as a prescale to timer 0 and timer 1 or as a DMA request source.

### Timer 0 Mode / Control Register

Offset : 56h  
Reset Value : 0000h

| 15 | 14                      | 13  | 12  | 11 | 10 | 9 | 8 | 7 | 6 | 5  | 4   | 3 | 2   | 1   | 0    |
|----|-------------------------|-----|-----|----|----|---|---|---|---|----|-----|---|-----|-----|------|
| EN | $\overline{\text{INH}}$ | INT | RIU | 0  | 0  | 0 | 0 | 0 | 0 | MC | RTG | P | EXT | ALT | CONT |

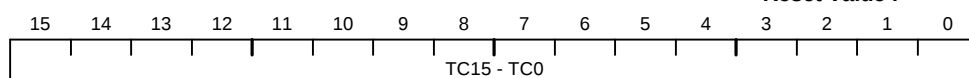
These bits definition for timer 0 are same as the bits of register 5Eh for timer 1.

### Timer 0 Count Register

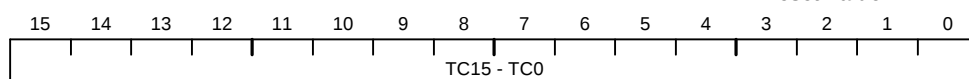
Offset : 50h  
Reset Value : —

| 15         | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| TC15 - TC0 |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

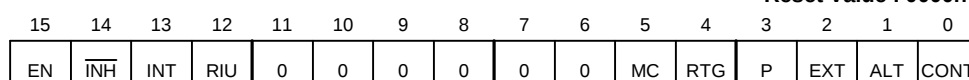
**Bit 15 – 0: TC15-TC0**, Timer 0 Count Value. This register contains the current count of timer 0. The count is incremented by one every four internal processor clocks or by prescaled the timer 2, or by one every four external clock which is configured the external clock select bit to refer the TMRIN1 signal.

**Timer 0 Maxcount Compare A Register**
**Offset : 52h**
**Reset Value : —**


**Bit 15-0 : TC15 – TC0**, Timer 0 Compare A Value.

**Timer 0 Maxcount Compare B Register**
**Offset : 54h**
**Reset Value : —**


**Bit 15-0 : TC15 – TC0**, Timer 0 Compare B Value.

**Timer 1 Mode / Control Register**
**Offset : 5Eh**
**Reset Value : 0000h**


**Bit 15: EN**, Enable Bit.

Set 1: The timer 1 is enable.

Set 0: The timer 1 is inhibited from counting.

The  $\overline{\text{INH}}$  bit must be set 1 during writing the EN bit, and the  $\overline{\text{INH}}$  bit and EN bit must be in the same write.

**Bit 14:  $\overline{\text{INH}}$** , Inhibit Bit. This bit is allows selective updating the EN bit. The  $\overline{\text{INH}}$  bit must be set 1 during writing the EN bit, and both the  $\overline{\text{INH}}$  bit and EN bit must be in the same write. This bit is not stored and is always read as 0.

**Bit 13: INT**, Interrupt Bit.

Set 1: A interrupt request is generated when the count register equals a maximum count. If the timer is configured in dual max-count mode, an interrupt is generated each time the count reaches max-count A or max-count B

Set 0: Timer 1 will not issue interrupt request.

**Bit 12: RIU**, Register in Use Bit.

Set 1: The Maxcount Compare B register of timer 1 is being used

Set 0: The Maxcount Compare A register of timer 1 is being used

**Bit 11-6 :** Reserved.

**Bit 5: MC**, Maximum Count Bit. When the timer reaches its maximum count, the MC bit will set to 1 by H/W. In dual maxcount mode, this bit is set each time either Maxcount Compare A or Maxcount Compare B register is reached. This bit is set regardless of the EN bit (66h.15).

**Bit 4: RTG**, Re-trigger Bit. This bit define the control function by the input signal of TMRIN1 pin. When EXT=1 (5Eh.2), this bit is ignored.

Set 1: Timer1 Count Register (58h) counts internal events; Reset the counting on every TMRIN1 input signal from low go high (rising edge trigger).

Set 0: Low input holds the timer 1 Count Register (58h) value; High input enables the counting which counts internal events.

**The definition of setting the (EXT , RTG )**

( 0 , 0 ) – Timer1 counts the internal events. if the TMRIN1 pin remains high.

( 0 , 1 ) -- Timer1 counts the internal events; count register reset on every rising transition on the TMRIN1 pin

( 1 , x ) -- TMRIN1 pin input acts as clock source and timer1 count register increase one every four external clock.

**Bit 3: P**, Prescaler Bit. This bit and EXT(5Eh.2) define the timer 1 clock source.

**The definition of setting the (EXT , P )**

( 0 , 0 ) – Timer1 Count Register increase one every four internal processor clock.

( 0 , 1 ) – Timer1 count register increase one which prescal by timer 2.

( 1 , x ) -- TMRIN1 pin input acts as clock source and Timer1 Count Register increase one every four external clock.

**Bit 2: EXT**, External Clock Bit.

Set 1: Timer 1 clock source from external

Set 0: Timer 1 clock source from internal

**Bit 1 : ALT**, Alternate Compare Bit. This bit controls whether the timer runs in single or dual maximum count mode.

Set 1: Specify dual maximum count mode. In this mode the timer counts to Maxcount Compare A, then resets the count register to 0. Then the timer counts to Maxcount Compare B, then resets the count register to 0 again, and starts over with Maxcount Compare A.

Set 0: Specify single maximum count mode. In this mode the timer will count to the value contained in Maxcount Compare A and reset to 0, and then the timer counts to Maxcount Compare A again. Maxcount Compare B is not used in this mode.

**Bit 0: CONT**, Continuous Mode Bit.

Set 1: The timer to run continuously.

Set 0: The timer will halt after each counting to the maximum count and the EN bit will be cleared.

**Timer 1 Count Register**

Offset : 58h

Reset Value : —

|            |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15         | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TC15 - TC0 |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

**Bit 15 – 0: TC15-TC0**, Timer 1 Count Value. This register contains the current count of timer 1. The count is incremented by one every four internal processor clocks or by prescaled the timer 2, or by one every four external clock which is configured the external clock select bit to refer the TMRIN1 signal.

**Timer 1 Maxcount Compare A Register**

Offset : 5Ah

Reset Value : —

|            |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15         | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TC15 - TC0 |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

**3Bit 15-0 : TC15 – TC0**, Timer 1 Compare A Value.

**Timer 1 Maxcount Compare B Register**

Offset : 5Ch

Reset Value : —

|            |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15         | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TC15 - TC0 |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

**Bit 15-0 : TC15 – TC0**, Timer 1 Compare B Value.

**Timer 2 Mode / Control Register**

Offset : 66h

Reset Value : 0000h

|    |     |     |    |    |    |   |   |   |   |    |   |   |   |   |      |
|----|-----|-----|----|----|----|---|---|---|---|----|---|---|---|---|------|
| 15 | 14  | 13  | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5  | 4 | 3 | 2 | 1 | 0    |
| EN | INH | INT | 0  | 0  | 0  | 0 | 0 | 0 | 0 | MC | 0 | 0 | 0 | 0 | CONT |

**Bit 15: EN**, Enable Bit.

Set 1: The timer 2 is enable.

Set 0: The timer 2 is inhibited from counting.

The  $\overline{\text{INH}}$  bit must be set 1 during writing the EN bit, and the  $\overline{\text{INH}}$  bit and EN bit must be in the same write.

**Bit 14:  $\overline{\text{INH}}$** , Inhibit Bit. This bit is allows selective updating the EN bit. The  $\overline{\text{INH}}$  bit must be set 1 during writing the EN bit, and both the  $\overline{\text{INH}}$  bit and EN bit must be in the same write. This bit is not stored and is always read as 0.

**Bit 13: INT**, Interrupt Bit.

Set 1: A interrupt request is generated when the count register equals a maximum count.

Set 0: Timer 2 will not issue interrupt request.

**Bit 12-6** : Reserved.

**Bit 5: MC**, Maximum Count Bit. When the timer reaches its maximum count, the MC bit will set to 1 by H/W. This bit is set regardless of the EN bit (66h.15).

**Bit 4-1**: Reserved.

**Bit 0: COUNT**, Continuous Mode Bit.

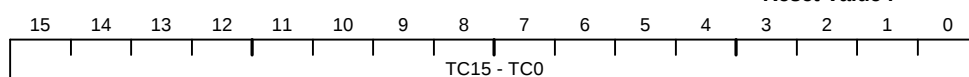
Set 1: Timer is continuously running when timer reaches the maximum count.

Set 0: The EN bit (66h.15) is cleared and the timer is hold after each timer count reaches the maximum count.

### Timer 2 Count Register

Offset : 60h

Reset Value : —

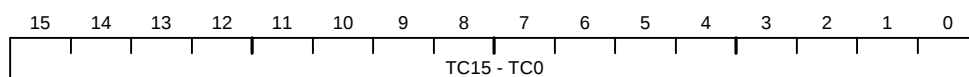


**Bit 15 – 0: TC15-TC0**, Timer 2 Count Value. This register contains the current count of timer 2. The count is incremented by one every four internal processor clocks.

### Timer 2 Maxcount Compare A Register

Offset : 62h

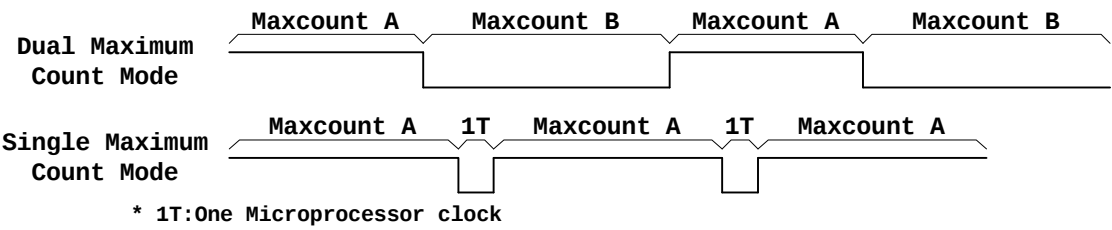
Reset Value : —



**Bit 15-0 : TC15 – TC0**, Timer 2 Compare A Value.

## 15.1 Timer/Counter Unit Output Mode

Timers 0 and 1 can use one maximum count value or two maximum count value. Timer 2 can use only one maximum count value. Timer 0 and timer1 can be configured to single or dual Maximum Compare count mode, the TMROUT0 or TMROUT1 signals can be used to generated waveform of various duty cycle.

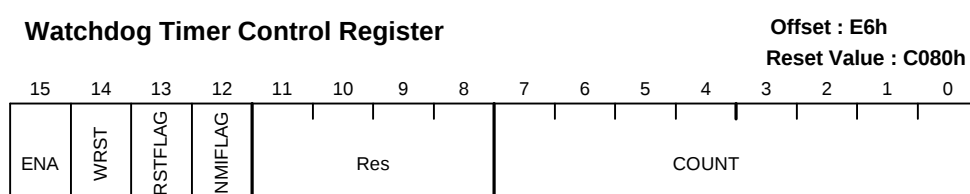


**Timer/Counter Unit Output Modes**

## 16. Watchdog Timer

R8820LV has one independent watchdog timer, which is programmable. **The watchdog timer is active after reset** and the timeout count with a maximum count value. The keyed sequence ( 3333h, CCCCh ) must be written to the register (E6h) first then writing new configuration to the Watchdog Timer Control Register. It is a single write so every one writing to Watchdog Timer Control Register must follow the rule.

To read the Watchdog Timer Control Register, the keyed sequence (5555h, AAAAh) must be written to the register (E6h) first. The current count should be reset before modifying the Watchdog Timer timeout period to ensure that an immediate timeout dose not occur.



**Bit 15: ENA**, Enable Watchdog Timer.

Set 1 : Enable Watchdog Timer.

Set 0 : Disable Watchdog Timer.

**Bit 14: WRST**, Watchdog Reset.

Set 1: WDT generates a system reset when WDT timeout count is reached.

Set 0 : WDT generates a NMI interrupt when WDT timeout count is reached if the NMIFLAG bit is 0. If the NMIFLAG bit is 1, the WDT will generate a system reset when timeout.

**Bit 13: RSTFLAG**, Reset Flag. When watchdog timer reset event has occurred, hardware will set this bit to 1. This bit will be cleared by any keyed sequence write to this register or external reset. This bit is 0 after an external reset or 1 after watchdog timer reset.

**Bit 12: NMIFLAG**, NMI Flag. After WDT generates a NMI interrupt, this bit will be set to 1 by H/W. This bit will be cleared by any keyed sequence write to this register.

Bit 11-8 : Reserved.

**Bit 7-0 : COUNT**, Timeout Count. The COUNT setting determines the duration of the watchdog timer timeout interval.

a. The duration equation : **Duration** =  $2^{\text{Exponent}}$  / **Frequency**

b. The Exponent of the COUNT setting:

(Bit 7, Bit 6, Bit 5, Bit 4, Bit 3, Bit 2, Bit 1, Bit 0) = ( Exponent)

( 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 ) = ( N/A )

( x , x , x , x , x , x , x , x ) = ( 10 )

( x , x , x , x , x , x , 1 , 0 ) = ( 20 )

( x , x , x , x , x , 1 , 0 , 0 ) = ( 21 )

(x, x, x, x, 1, 0, 0, 0) = (22)

(x, x, x, 1, 0, 0, 0, 0) = (23)

(x, x, 1, 0, 0, 0, 0, 0) = (24)

(x, 1, 0, 0, 0, 0, 0, 0) = (25)

(1, 0, 0, 0, 0, 0, 0, 0) = (26)

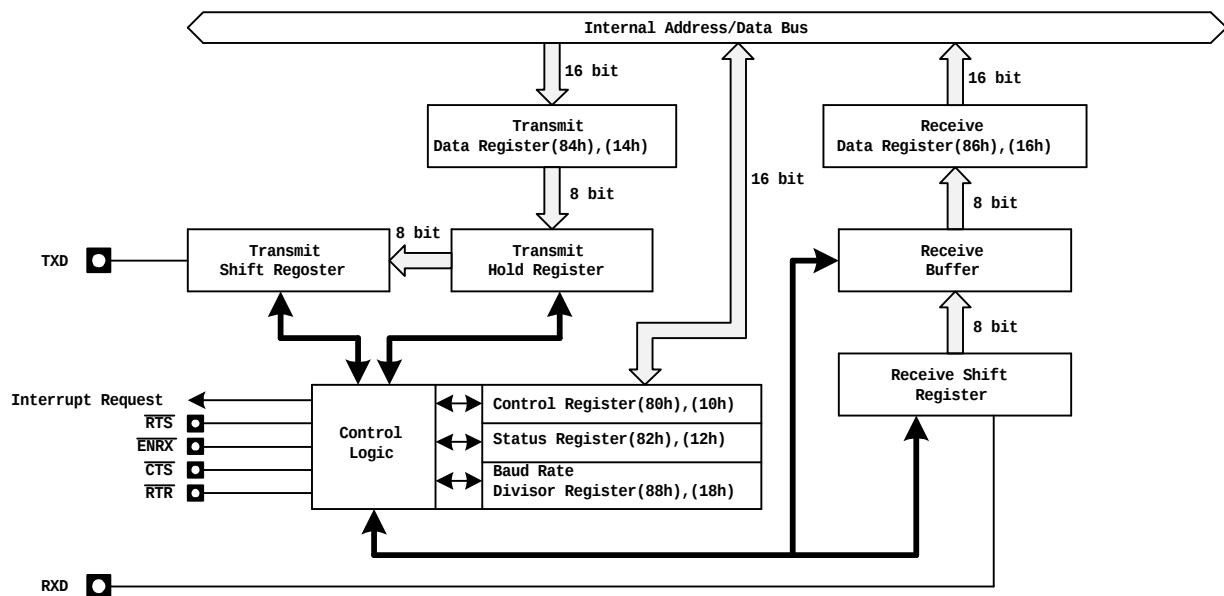
c. Watchdog timer Duration reference table:

| Frequency\Exponent | 10      | 20    | 21      | 22     | 23       | 24       | 25     | 26     |
|--------------------|---------|-------|---------|--------|----------|----------|--------|--------|
| <b>20 MHz</b>      | 51 us   | 52 ms | 104 ms  | 209 ms | 419 ms   | 838 ms   | 1.67 s | 3.35 s |
| <b>25 MHz</b>      | 40 us   | 41 ms | 83 ms   | 167 ms | 335 ms   | 671 ms   | 1.34 s | 2.68 s |
| <b>33 MHz</b>      | 30 us   | 31 ms | 62 ms   | 125 ms | 251 ms   | 503 ms   | 1.00 s | 2.01 s |
| <b>40 MHz</b>      | 25 us   | 26 ms | 52 ms   | 104 ms | 209 ms   | 419 ms   | 838 ms | 1.67 s |
| <b>50 MHz</b>      | 20.5 us | 21 ms | 41.9 ms | 83.9ms | 167.8 ms | 335.5 ms | 671 ms | 1.34 s |

## 17. Asynchronous Serial Port

R8820LV has two asynchronous serial ports, which provide the TXD, RXD pins for the full duplex bi-directional data transfer and with handshaking signals  $\overline{\text{CTS}}$ ,  $\overline{\text{ENRX}}$ ,  $\overline{\text{RTS}}$ ,  $\overline{\text{RTR}}$ . The serial ports support : 9-bit, 8-bit or 7-bit data transfer; odd parity, even parity, or no parity; 1 stop bits; Error detection; DMA transfers through the serial port; Multi-drop protocol (9-bit) support; Double buffers for transmit and receive.

The receive/transmit clock is based on the microprocessor clock. The serial port can be used in power-saved mode, but the transfer rate must be adjusted to correctly reflect the new internal operating frequency. Software is programmed through the registers ,(80h, 82h, 84h, 86h, 88h – for port 0), ( 10h,12h,14h,16h,18h – for port 1) to configure the asynchronous serial port.



**Serial Port Block Diagram**

### 17.1 Serial Port Flow Control

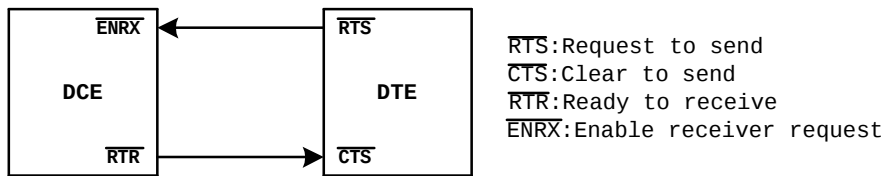
The two serial ports provided with two data pins (RXD and TXD) and two flow control signals ( $\overline{\text{RTS}}$ ,  $\overline{\text{RTR}}$ ). Hardware flow control is enabled when the FC bit in the Serial Port control Register is set. And the flow control signals are configured by software to support several different protocols.

#### 17.1.1 DCE/DTE Protocol

The R8820LV can be as a DCE (Data Communication Equipment) or as a DTE (Data Terminal Equipment). This protocol provides flow control where one serial port is receiving data and other serial port is sending data. To implement the DCE device, the ENRX bit should be set and the RTS bit should be cleared for the associated serial port. To implement the DTE device, the ENRX bit should be cleared and the RTS bit should be set for the associated serial port. The ENRX bit and RTS bit are in the register F2h.

The DCE/DTE protocol is asymmetric interface since the DTE device can not signal the DCE device that is ready to receive

data, and the DCE can not send the request to send signal.



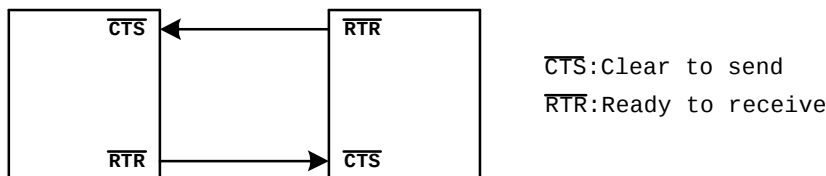
### DCE/DTE Protocol Connection

The DCE/DTE protocol communication step:

- DTE send data to DCE
- $\overline{\text{RTS}}$  signal is asserted by DTE when data is available.
- The  $\overline{\text{RTS}}$  signal is interpreted by the DCE device as a request to enable its receiver.
- The DCE asserts the  $\overline{\text{RTR}}$  signal to response that DCE is ready to receive data.

#### 17.1.2 CTS/RTR Protocol

The serial port can be programmed as a CTS/RTS protocol by clearing both  $\overline{\text{ENRX}}$  bit and  $\text{RTS}$  bit. This protocol is a symmetric interface, which provides flow control when both ports are sending and receiving data.



### CTS/RTR Protocol Connection

#### 17.2 DMA Transfer to/from a serial port function

DMA transfers to the serial port function as destination-synchronized DMA transfers. A new transfer is requested when the Transmit Holding Register is empty. When the port is configured for DMA transmits, the corresponding transmit interrupt is disabled regardless of the  $\text{TXIE}$  bit setting.

DMA transfers from the serial port function as source-synchronized DMA transfers. A new transfer is requested when the Receive Buffer contains valid data. When the port is configured for DMA receives, the corresponding receive interrupt is disabled regardless of the  $\text{RXIE}$  bit setting.

**The DMA request is generated internally when a DMA channel is being used for serial port transfers. And the  $\text{DRQ0}$  or  $\text{DRQ1}$  are not active when a serial port DMA transfers.**

Hardware handshaking may be used in conjunction with serial port DMA transfers.

### 17.3 The Asynchronous Modes description

There are 4 modes operation in the asynchronous serial port.

**Mode1:** Mode 1 is the 8-bit asynchronous communications mode. Each frame consists of a start bit, eight data bits and a stop bit. when parity is used, the eighth data bit becomes the parity bit.

**Mode 2:** Mode 2 is used together with Mode 3 for multiprocessor communications over a common serial link. In mode 2, the RX machine will not complete a reception unless the ninth data bit is a one. Any character received with the ninth bit equal to zero is ignored. No flags are set, no interrupts occur and no data is transferred to Receive Data Register. In mode 3, characters are received regardless of the state of the ninth data bit.

**Mode 3:** Mode 3 is the 9-bit asynchronous communications mode. Mode 3 is the same as mode 1 except that a frame contains nine data bits. The ninth data bit becomes the parity bit when the parity feature is enabled.

**Mode 4:** Mode 4 is the 7-bit asynchronous communications mode. Each frame consists of a start bit, seven data bits and a stop bit. Parity bit is not available in mode 4.

**Serial Port 0 Contrl Register**

Offset : 80h  
Reset Value : 0000h

| 15  | 14 | 13 | 12   | 11  | 10  | 9  | 8    | 7    | 6     | 5     | 4   | 3  | 2    | 1 | 0 |
|-----|----|----|------|-----|-----|----|------|------|-------|-------|-----|----|------|---|---|
| DMA |    |    | RISE | BRK | TB8 | FC | TXIE | RXIE | TMODE | RMODE | EVN | PE | MODE |   |   |

**Bit 15-13: DMA**, DMA Control Field. These bits configure the serial port for use with DMA transfers.

**DMA control bits**

| (Bit 15, bit 14, bit 13)b | --- | <u>Receive</u> | --- | <u>Transmit</u> |
|---------------------------|-----|----------------|-----|-----------------|
| ( 0, 0, 0 )               | --- | No DMA         | --- | No DMA          |
| ( 0, 0, 1 )               | --- | DMA 0          | --- | DMA 1           |
| ( 0, 1, 0 )               | --- | DMA 1          | --- | DMA 0           |
| ( 0, 1, 1 )               | --- | N/A            | --- | N/A             |
| ( 1, 0, 0 )               | --- | DMA 0          | --- | No DMA          |
| ( 1, 0, 1 )               | --- | DMA 1          | --- | No DMA          |
| ( 1, 1, 0 )               | --- | No DMA         | --- | DMA 0           |
| ( 1, 1, 1 )               | --- | No DMA         | --- | DMA 1           |

**Bit 12: RSIE**, Receive Status Interrupt Enable. An exception occurs during data reception or error detection occur will generate an interrupt.

Set 1: Enable the serial port 0 to generate an interrupt request.

**Bit 11: BRK**, Send Break.

Set this bit to 1 , the TXD pin always drives low.

Long Break : The TXD is driven low for grater than (2M+3) bit times;

Short break : The TXD is driven low for grater than M bit times;

\* M= start bit + data bits number + parity bit + stop bit

**Bit 10 : TB8**, Transmit Bit 8. This bit is transmitted as ninth data bit in mode 2 and mode 3. This bit is cleared after every transmission.

**Bit 9: FC**, Flow Control Enable.

Set 1: Enable the hardware flow control for serial port 0.

Set 0 : Disable the hardware flow control for serial port 0.

**Bit 8 : TXIE**, Transmitter Ready Interrupt Enable. When the Transmit Holding Register is empty ( THRE bit in Status Register is set ),it will have an interrupt occurs.

Set 1: Enable the Interrupt.

Set 0 : Disable the interrupt.

**Bit 7: RXIE**, Receive Data Ready Interrupt Enable. When the receiver buffer contains valid data ( RDR bit in Status Register is set) , it will generate an interrupt.

Set 1: Enable the Interrupt.

Set 0 : Disable the interrupt.

**Bit 6 : TMODE**, Transmit Mode.

Set 1: Enable the TX machines.

Set 1: Disable the TX machines.

**Bit 5: RMODE**, Received Mode.

Set 1: Enable the RX machines.

Set 1: Disable the RX machines.

**Bit 4: EVN**, Even Parity. This bit is valid only when the PE bit is set.

Set 1: the even parity checking is enforced (even number of 1s in frame).

Set 0: odd parity checking is enforced (odd number of 1s in frame).

**Bit 3: PE**, Parity Enable.

Set 1 : Enable the parity checking.

Set 0 : Disable the parity checking.

**Bit 2-0: MODE**, Mode of Operation.

| ( bit 2, bit 1, bit 0) | MODE   | Data Bits | Parity Bits | Stop Bits |
|------------------------|--------|-----------|-------------|-----------|
| ( 0 , 0 , 1)           | Mode 1 | 7 or 8    | 1 or 0      | 1         |
| ( 0 , 1 , 0)           | Mode 2 | 9         | N/A         | 1         |
| ( 0 , 1 , 1)           | Mode 3 | 8 or 9    | 1 or 0      | 1         |
| ( 1 , 0 , 0)           | Mode 4 | 7         | N/A         | 1         |

**Serial Port 0 Status Register**

Offset : 82h

Reset Value : —

|          |    |    |    |    |      |      |     |     |      |     |     |     |      |     |     |
|----------|----|----|----|----|------|------|-----|-----|------|-----|-----|-----|------|-----|-----|
| 15       | 14 | 13 | 12 | 11 | 10   | 9    | 8   | 7   | 6    | 5   | 4   | 3   | 2    | 1   | 0   |
| Reserved |    |    |    |    | BRK1 | BRK0 | RB8 | RDR | THRE | FER | OER | PER | TEMT | HS0 | Res |

The Serial Port 0 Status Register provides information about the current status of the serial port 0.

**Bit 15-11:** Reserved.

**Bit 10: BRK1**, Long Break Detected. This bit should be reset by software.

When a long break is detected, this bit will be set high.

**Bit 9 : BRK0**, Short Break Detected. This bit should be reset by software.

When a short break is detected, this bit will be set high

**Bit 8: RB8**, Received Bit 8. This bit should be reset by software.

This bit contains the ninth data bit received in mode 2 and mode 3.

**Bit 7: RDR**, Received Data Ready. Read only.

The Received Data Register contains valid data, this bit is set high. This bit can only be reset by reading the Serial Port 0 Receive Register.

**Bit 6: THRE**, Transmit Hold Register Empty. Read only.

When the Transmit Hold Register is ready to accept data, this bit will be set. This bit will be reset when writing data to the Transmit Hold Register.

**Bit 5: FER**, Framing Error detected. This bit should be reset by software.

This bit is set when a framing error is detected.

**Bit 4: OER**, Overrun Error Detected. This bit should be reset by software.

This bit is set when an overrun error is detected.

**Bit 3: PER**, Parity Error Detected. This bit should be reset by software.

This bit is set when a parity error ( for mode 1 and mode 3) is detected.

**Bit 2: TEMT**, Transmitter Empty. This bit is read only.

When the Transmit Shift Register is empty, this bit will be set.

**Bit 1: HS0**, Handshake Signal 0. This bit is read only.

This bit reflects the inverted value of the external  $\overline{\text{CTS0}}$  pin.

**Bit 0 :** Reserved.

**Serial Port 0 Transmit Register**

Offset : 84h

Reset Value : —

|          |    |    |    |    |    |   |   |       |   |   |   |   |   |   |   |
|----------|----|----|----|----|----|---|---|-------|---|---|---|---|---|---|---|
| 15       | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7     | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Reserved |    |    |    |    |    |   |   | TDATA |   |   |   |   |   |   |   |

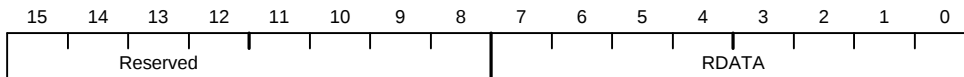
**Bit 15-8:** Reserved

**Bit 7-0 :** TDATA, Transmit Data. Software writes this register with data to be transmitted on the serial port 0.

**Serial Port 0 Receive Register**

Offset : 86h

Reset Value : —



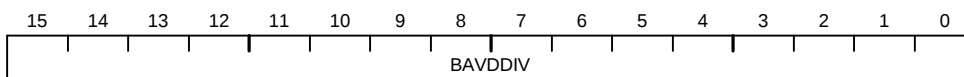
**Bit 15-8:** Reserved

**Bit 7-0: RDATA,** Received DATA. The RDR bit should be read as 1 before read the RDATA register to avoid reading invalid data.

**Serial Port 0 Baud Rate Divisor Register**

Offset : 88h

Reset Value : 0000h



**Bit 15-0: BAUDDIV,** Baud Rate Divisor.

The general formula for baud rate divisor is **Baud Rate = Microprocessor Clock / (16 x BAUDIV)**

For example, The Microprocessor clock is 22.1184MHz and the BBDIV=5 (Decimal), the baud rate of serial port is 115.2k.

**Serial Port 1 Contrl Register**

Offset : 10h

Reset Value : 0000h

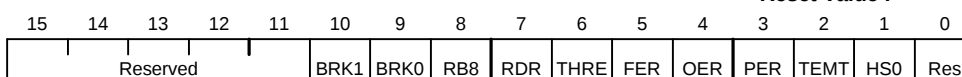


These bits definition are same as the bits definition of Register 80h

**Serial Port 1 Status Register**

Offset : 12h

Reset Value : —

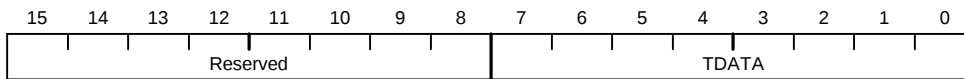


These bits definition are same as the bits definition of Register 82h

**Serial Port 1 Transmit Register**

Offset : 14h

Reset Value : —

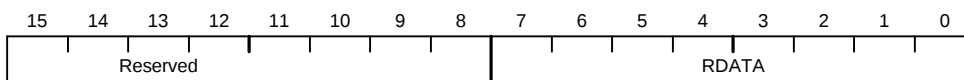


These bits definition are same as the bits definition of Register 84h

**Serial Port 1 Receive Register**

Offset : 16h

Reset Value : —

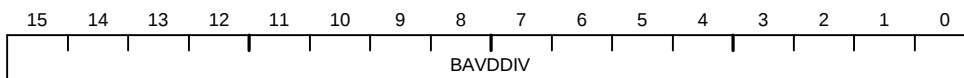


These bits definition are same as the bits definition of Register 86h

**Serial Port 1 Baud Rate Divisor Register**

Offset : 18h

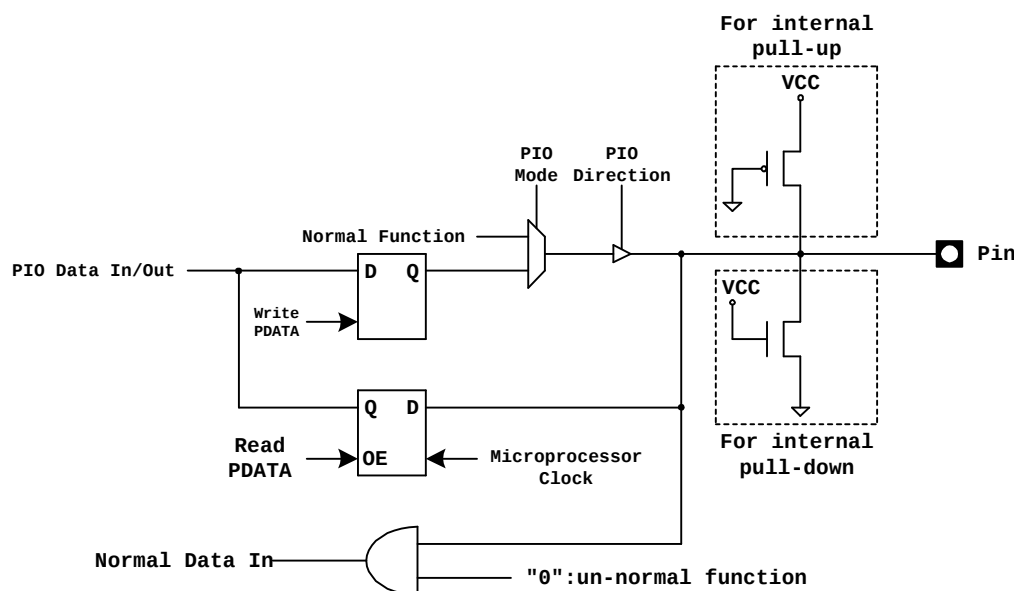
Reset Value : 0000h



These bits definition are same as the bits definition of Register 88h

## 18. PIO Unit

R8820LV provides 32 programmable I/O signals, which are multi-function pins with others normal function signals. Software is programmed through the registers ( 7Ah, 78h, 76h, 74h, 72h, 70h) to configure the multi-function pins for PIO or normal function.



**PIO pin Operation Diagram**

### 18.1 PIO multi-function Pin list table

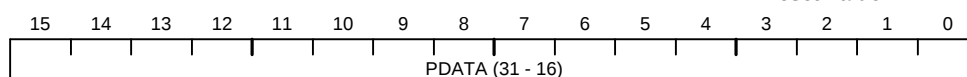
| PIO No. | Pin No. | Multi Function     | Reset status/PIO internal resister         |
|---------|---------|--------------------|--------------------------------------------|
| 0       | 72      | TMRIN1             | Input with 10k pull-up                     |
| 1       | 73      | TMROUT1            | Input with 10k pull-down                   |
| 2       | 59      | PCS6 /A2           | Input with 10k pull-up                     |
| 3       | 60      | PCS5 /A1           | Input with 10k pull-up                     |
| 4       | 48      | DT/ $\overline{R}$ | Normal operation/ Input with 10k pull-up   |
| 5       | 49      | $\overline{DEN}$   | Normal operation/ Input with 10k pull-up   |
| 6       | 46      | SRDY               | Normal operation/ Input with 10k pull-down |
| 7       | 22      | A17                | Normal operation/ Input with 10k pull-up   |
| 8       | 20      | A18                | Normal operation/ Input with 10k pull-up   |
| 9       | 19      | A19                | Normal operation/ Input with 10k pull-up   |
| 10      | 74      | TMROUT0            | Input with 10k pull-down                   |
| 11      | 75      | TMRIN0             | Input with 10k pull-up                     |
| 12      | 77      | DRQ0/INT5          | Input with 10k pull-up                     |
| 13      | 76      | DRQ1/INT6          | Input with 10k pull-up                     |
| 14      | 50      | $\overline{MCS0}$  | Input with 10k pull-up                     |
| 15      | 51      | $\overline{MCS1}$  | Input with 10k pull-up                     |
| 16      | 66      | $\overline{PCS0}$  | Input with 10k pull-up                     |
| 17      | 65      | $\overline{PCS1}$  | Input with 10k pull-up                     |

|    |     |                                                                             |                          |
|----|-----|-----------------------------------------------------------------------------|--------------------------|
| 18 | 63  | $\overline{\text{PCS2}} / \overline{\text{CTS1}} / \overline{\text{ENRX1}}$ | Input with 10k pull-up   |
| 19 | 62  | $\overline{\text{PCS3}} / \overline{\text{RTS1}} / \overline{\text{RTR1}}$  | Input with 10k pull-up   |
| 20 | 3   | $\overline{\text{RTS0}} / \overline{\text{RTR0}}$                           | Input with 10k pull-up   |
| 21 | 100 | $\overline{\text{CTS0}} / \overline{\text{ENRX0}}$                          | Input with 10k pull-up   |
| 22 | 2   | TXD0                                                                        | Input with 10k pull-down |
| 23 | 1   | RXD0                                                                        | Input with 10k pull-down |
| 24 | 68  | $\overline{\text{MCS2}}$                                                    | Input with 10k pull-up   |
| 25 | 69  | $\overline{\text{MCS3}} / \overline{\text{RFSH}}$                           | Input with 10k pull-up   |
| 26 | 97  | $\overline{\text{UZI}}$                                                     | Input with 10k pull-up   |
| 27 | 98  | TXD1                                                                        | Input with 10k pull-up   |
| 28 | 99  | RXD1                                                                        | Input with 10k pull-up   |
| 29 | 96  | S6/ $\overline{\text{CLKDIV}}$                                              | Input with 10k pull-up   |
| 30 | 52  | INT4                                                                        | Input with 10k pull-up   |
| 31 | 54  | INT2                                                                        | Input with 10k pull-up   |

**PIO Data 1 Register**

Offset : 7Ah

Reset Value : —

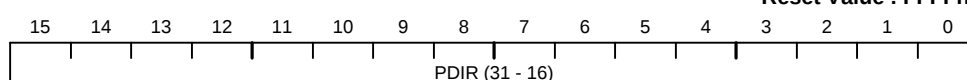

**Bit 15- 0 : PDATA31-PDATA16, PIO Data Bits.**

These bits PDATA31- PDATA16 map to the PIO31 –PIO16 which indicate the driven level when the PIO pin as an output or reflects the external level when the PIO pin as an input .

**PIO Direction 1 Register**

Offset : 78h

Reset Value : FFFFh


**Bit 15-0 : PDIR 31- PDIR16, PIO Direction Register.**

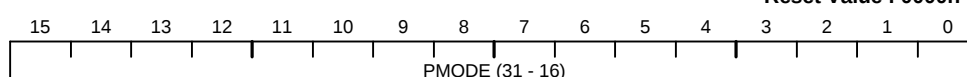
Set 1: Configure the PIO pin as an input.

Set 0: Configure the PIO pin as an output or as normal pin function.

**PIO Mode 1 Register**

Offset : 76h

Reset Value : 0000h


**Bit 15-0: PMODE31-PMODE16, PIO Mode Bit.**

The definition of PIO pins are configured by the combination of PIO Mode and PIO Direction. And the PIO pin is programmed individual.

The definition (PIO Mode, PIO Direction) for PIO pin function:

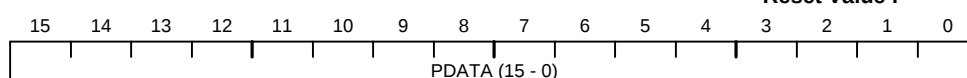
( 0 , 0 ) – Normal operation , ( 0 , 1 ) – PIO input with pull-up/pull-down

( 1 , 0 ) – PIO output , ( 1 , 1 ) -- PIO input without pull-up/pull-down

### PIO Data 0 Register

Offset : 74h

Reset Value : —



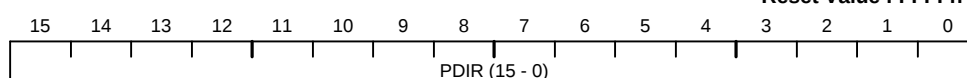
**Bit 15-0 : PDATA15- PDATA0** : PIO Data Bus.

These bits PDATA15- PDATA0 map to the PIO15 –PIO0 which indicate the driven level when the PIO pin as an output or reflects the external level when the PIO pin as an input.

### PIO Direction 0 Register

Offset : 72h

Reset Value : FFFFh



**Bit 15-0 : PDIR 15- PDIR0**, PIO Direction Register.

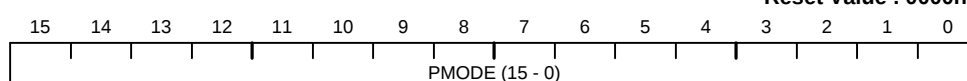
Set 1: Configure the PIO pin as an input.

Set 0: Configure the PIO pin as an output or as normal pin function.

### PIO Mode 0 Register

Offset : 70h

Reset Value : 0000h



**Bit 15-0: PMODE15-PMODE0**, PIO Mode Bit.

## 19. PSRAM Control Unit

The PSRAM interface is provided by the R8820LV and the refresh control unit automatically generates refresh bus cycles. The refresh control unit uses the internal microprocessor clock as a operating source clock. if the power-saved mode is enabled, the refresh control unit must be programmed to reflect the new clock rate. Software programs the registers (E0, E2, E4) to control the refresh control unit operation.

### Memory Partition Register

Offset : E0h

Reset Value : 0000h

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
|    |    |    |    |    |    |   | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

M6 - M0

**Bit 15-9:** M6-M0, Refresh Base. M6-M0 map to A19-A13 of the 20-bit memory refresh address.

**Bit 8-0 :** Reserved.

### Clock Prescaler Register

Offset : E2h

Reset Value : —

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0  | 0  | 0  | 0  | 0  | 0  | 0 |   |   |   |   |   |   |   |   |   |

RC8 - RC0

**Bit 15-9 :** Reserved

**Bit 8-0:** RC8-RC0, Refresh Counter Reload Value.

### Enable RCU Register

Offset : E4h

Reset Value : 0000h

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| E  | 0  | 0  | 0  | 0  | 0  | 0 |   |   |   |   |   |   |   |   |   |

T8 - T0

**Bit 15:** E, Enable RCU.

Set 1: Enable the refresh counter unit

Set 0 : Disable the refresh counter unit.

**Bit 14-9 :** Reserved

**Bit 8-0:** T8-T0, Refresh Count. Read only bits and these bits present value of the down counter which triggers refresh requests.

## 20. INSTRUCTION SET OPCODES AND CLOCK CYCLES

| Function                            | Format      |               |             |               | Clocks     | Notes |
|-------------------------------------|-------------|---------------|-------------|---------------|------------|-------|
| DATA TRANSFER INSTRUCTIONS          |             |               |             |               |            |       |
| MOV = Move                          |             |               |             |               |            |       |
| register to register/memory         | 1000100w    | mod reg r/m   |             |               | 1/1        |       |
| register/memory to register         | 1000101w    | mod reg r/m   |             |               | 1/6        |       |
| immediate to register/memory        | 1100011w    | mod 000 r/m   | data        | data if w=1   | 1/1        |       |
| immediate to register               | 1011w reg   | data          | data if w=1 |               | 1          |       |
| memory to accumulator               | 1010000w    | addr-low      | addr-high   |               | 6          |       |
| accumulator to memory               | 1010001w    | addr-low      | addr-high   |               | 1          |       |
| register/memory to segment register | 10001110    | mod 0 reg r/m |             |               | 3/8        |       |
| segment register to register/memory | 10001100    | mod 0 reg r/m |             |               | 2/2        |       |
| PUSH = Push                         |             |               |             |               |            |       |
| memory                              | 11111111    | mod 110 r/m   |             |               | 8          |       |
| register                            | 01010 reg   |               |             |               | 3          |       |
| segment register                    | 000reg110   |               |             |               | 2          |       |
| immediate                           | 011010s0    | data          | data if s=0 |               | 1          |       |
| POP = Pop                           |             |               |             |               |            |       |
| memory                              | 10001111    | mod 000 r/m   |             |               | 8          |       |
| register                            | 01011 reg   |               |             |               | 6          |       |
| segment register                    | 000 reg 111 | (reg: 011)    |             |               | 8          |       |
| PUSHA = Push all                    |             |               |             |               |            |       |
| POPA = Pop all                      |             |               |             |               |            |       |
| XCHG = Exchange                     |             |               |             |               |            |       |
| register/memory                     | 1000011w    | mod reg r/m   |             |               | 3/8        |       |
| register with accumulator           | 10010 reg   |               |             |               | 3          |       |
| XTAL = Translate byte to AL         |             |               |             |               |            |       |
| IN = Input from                     |             |               |             |               |            |       |
| fixed port                          | 1110010w    | port          |             |               | 12         |       |
| variable port                       | 1110110w    |               |             |               | 12         |       |
| OUT = Output from                   |             |               |             |               |            |       |
| fixed port                          | 1110010w    | port          |             |               | 12         |       |
| variable port                       | 1110110w    |               |             |               | 12         |       |
| LEA = Load EA to register           | 10001101    | mod reg r/m   |             |               | 1          |       |
| LDS = Load pointer to DS            | 11000101    | mod reg r/m   | (mod: 11)   |               | 14         |       |
| LES = Load pointer to ES            | 11000100    | mod reg r/m   | (mod: 11)   |               | 14         |       |
| ENTER = Build stack frame           |             |               |             |               |            |       |
| L = 0                               | 11001000    | data-low      | data-high   | L             | 7          |       |
| L = 1                               |             |               |             |               | 11         |       |
| L > 1                               |             |               |             |               | 11+10(L-1) |       |
| LEAVE = Tear down stack frame       |             |               |             |               |            |       |
| LAHF = Load AH with flags           | 11001001    |               |             |               | 7          |       |
| SAHF = Store AH into flags          | 10011111    |               |             |               | 2          |       |
| PUSHF = Push flags                  | 10011110    |               |             |               | 2          |       |
| POPF = Pop flags                    | 10011100    |               |             |               | 2          |       |
|                                     | 10011101    |               |             |               | 11         |       |
| ARITHMETIC INSTRUCTIONS             |             |               |             |               |            |       |
| ADD = Add                           |             |               |             |               |            |       |
| reg/memory with register to either  | 000000dw    | mod reg r/m   |             |               | 1/7        |       |
| immediate to register/memory        | 100000sw    | mod 000 r/m   | data        | data if sw=01 | 1/8        |       |
| immediate to accumulator            | 0000010w    | data          | data if w=1 |               | 1          |       |

| Function                                    | Format    |             |             |               | Clocks | Notes |
|---------------------------------------------|-----------|-------------|-------------|---------------|--------|-------|
| <b>ADC</b> = Add with carry                 |           |             |             |               |        |       |
| reg/memory with register to either          | 000100dw  | mod reg r/m |             |               | 1/7    |       |
| immediate to register/memory                | 100000sw  | mod 010 r/m | data        | data if sw=01 | 1/8    |       |
| immediate to accumulator                    | 0001010w  | data        | data if w=1 |               | 1      |       |
| <b>INC</b> = Increment                      |           |             |             |               |        |       |
| register/memory                             | 111111w   | mod 000 r/m |             |               | 1/8    |       |
| register                                    | 01000 reg |             |             |               | 1      |       |
| <b>SUB</b> = Subtract                       |           |             |             |               |        |       |
| reg/memory with register to either          | 0001010dw | mod reg r/m |             |               | 1/7    |       |
| immediate from register/memory              | 100000sw  | mod 101 r/m | data        | data if sw=01 | 1/8    |       |
| immediate from accumulator                  | 0001110w  | data        | data if w=1 |               | 1      |       |
| <b>SBB</b> = Subtract with borrow           |           |             |             |               |        |       |
| reg/memory with register to either          | 000110dw  | mod reg r/m |             |               | 1/7    |       |
| immediate from register/memory              | 100000sw  | mod 011 r/m |             |               | 1/8    |       |
| immediate from accumulator                  | 0001110w  | data        | data if w=1 |               | 1      |       |
| <b>DEC</b> = Decrement                      |           |             |             |               |        |       |
| register/memory                             | 111111w   | mod 001 r/m |             |               | 1/8    |       |
| register                                    | 01001 reg |             |             |               | 1      |       |
| <b>NEG</b> = Change sign                    |           |             |             |               |        |       |
| register/memory                             | 1111011w  | mod reg r/m |             |               | 1/8    |       |
| <b>CMP</b> = Compare                        |           |             |             |               |        |       |
| register/memory with register               | 0011101w  | mod reg r/m |             |               | 1/7    |       |
| register with register/memory               | 0011100w  | mod reg r/m |             |               | 1/7    |       |
| immediate with register/memory              | 100000sw  | mod 111 r/m | data        | data if sw=01 | 1/7    |       |
| immediate with accumulator                  | 0011110w  | data        | data if w=1 |               | 1      |       |
| <b>MUL</b> = multiply (unsigned)            |           |             |             |               |        |       |
| register-byte                               | 1111011w  | mod 100 r/m |             |               | 13     |       |
| register-word                               |           |             |             |               | 21     |       |
| memory-byte                                 |           |             |             |               | 18     |       |
| memory-word                                 |           |             |             |               | 26     |       |
| <b>IMUL</b> = Integer multiply (signed)     |           |             |             |               |        |       |
| register-byte                               | 1111011w  | mod 101 r/m |             |               | 16     |       |
| register-word                               |           |             |             |               | 24     |       |
| memory-byte                                 |           |             |             |               | 21     |       |
| memory-word                                 |           |             |             |               | 29     |       |
| register/memory multiply immediate (signed) | 011010s1  | mod reg r/m | data        | data if s=0   | 23/28  |       |
| <b>DIV</b> = Divide (unsigned)              |           |             |             |               |        |       |
| register-byte                               | 1111011W  | mod 110 r/m |             |               | 18     |       |
| register-word                               |           |             |             |               | 26     |       |
| memory-byte                                 |           |             |             |               | 23     |       |
| memory-word                                 |           |             |             |               | 31     |       |
| <b>IDIV</b> = Integer divide (signed)       |           |             |             |               |        |       |
| register-byte                               | 1111011w  | mod 111 r/m |             |               | 18     |       |
| register-word                               |           |             |             |               | 26     |       |
| memory-byte                                 |           |             |             |               | 23     |       |
| memory-word                                 |           |             |             |               | 31     |       |
| <b>AAS</b> = ASCII adjust for subtraction   | 00111111  |             |             |               | 3      |       |
| <b>DAS</b> = Decimal adjust for subtraction | 00101111  |             |             |               | 2      |       |
| <b>AAA</b> = ASCII adjust for addition      | 00110111  |             |             |               | 3      |       |
| <b>DAA</b> = Decimal adjust for addition    | 00100111  |             |             |               | 2      |       |
| <b>AAD</b> = ASCII adjust for divide        | 11010101  | 00001010    |             |               | 14     |       |
| <b>AAM</b> = ASCII adjust for multiply      | 11010100  | 00001010    |             |               | 15     |       |
| <b>CBW</b> = Convert byte to word           | 10011000  |             |             |               | 2      |       |
| <b>CWD</b> = Convert word to double-word    | 10011001  |             |             |               | 2      |       |

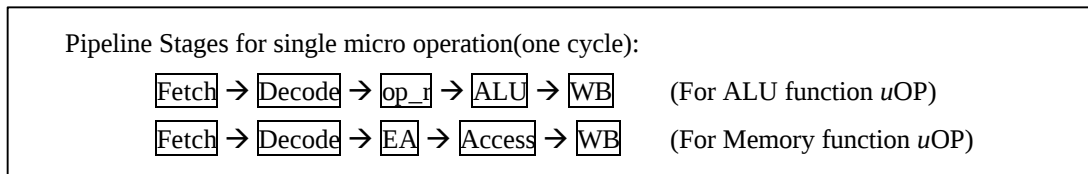
| Function                                           | Format   |             |             |             | Clocks    | Notes |
|----------------------------------------------------|----------|-------------|-------------|-------------|-----------|-------|
| <b>BIT MANIPULATION INSTRUCTUIONS</b>              |          |             |             |             |           |       |
| <b>NOT</b> = Invert register/memory                | 1111011w | mod 010 r/m |             |             | 1/7       |       |
| <b>AND</b> = And                                   |          |             |             |             |           |       |
| reg/memory and register to either                  | 001000dw | mod reg r/m |             |             | 1/7       |       |
| immediate to register/memory                       | 1000000w | mod 100 r/m | data        | data if w=1 | 1/8       |       |
| immediate to accumulator                           | 0010010w | data        | data if w=1 |             | 1         |       |
| <b>OR</b> = Or                                     |          |             |             |             |           |       |
| reg/memory and register to either                  | 000010dw | mod reg r/m |             |             | 1/7       |       |
| immediate to register/memory                       | 1000000w | mod 001 r/m | data        | data if w=1 | 1/8       |       |
| immediate to accumulator                           | 0000110w | data        | data if w=1 |             | 1         |       |
| <b>XOR</b> = Exclusive or                          |          |             |             |             |           |       |
| reg/memory and register to either                  | 001100dw | mod reg r/m |             |             | 1/7       |       |
| immediate to register/memory                       | 1000000w | mod 110 r/m | data        | data if w=1 | 1/8       |       |
| immediate to accumulator                           | 0011010w | data        | data if w=1 |             | 1         |       |
| <b>TEST</b> = And function to flags , no result    |          |             |             |             |           |       |
| register/memory and register                       | 1000010w | mod reg r/m |             |             | 1/7       |       |
| immediate data and register/memory                 | 1111011w | mod 000 r/m | data        | data if w=1 | 1/8       |       |
| immediate data and accumulator                     | 1010100w | data        | data if w=1 |             | 1         |       |
| <b>Sifts/Rotates</b>                               |          |             |             |             |           |       |
| register/memory by 1                               | 1101000w | mod TTT r/m |             |             | 2/8       |       |
| register/memory by CL                              | 1101001w | mod TTT r/m |             |             | 1+n / 7+n |       |
| register/memory by Count                           | 1100000w | mod TTT r/m | count       |             | 1+n / 7+n |       |
| <b>STRING MANIPULATION INSTRUCTIONS</b>            |          |             |             |             |           |       |
| <b>MOVS</b> = Move byte/word                       | 1010010w |             |             |             | 13        |       |
| <b>INS</b> = Input byte/word from DX port          | 0110110w |             |             |             | 13        |       |
| <b>OUTS</b> = Output byte/word to DX port          | 0110111w |             |             |             | 13        |       |
| <b>CMPS</b> = Compare byte/word                    | 1010011w |             |             |             | 18        |       |
| <b>SCAS</b> = Scan byte/word                       | 101011w  |             |             |             | 13        |       |
| <b>LODS</b> = Load byte/word to AL/AX              | 1010110w |             |             |             | 13        |       |
| <b>STOS</b> = Store byte/word from AL/AX           | 1010101w |             |             |             | 7         |       |
| <b>Repeated by count in CX:</b>                    |          |             |             |             |           |       |
| <b>MOVS</b> = Move byte/word                       | 11110010 | 1010010w    |             |             | 4+9n      |       |
| <b>INS</b> = Input byte/word from DX port          | 11110010 | 0110110w    |             |             | 5+9n      |       |
| <b>OUTS</b> = Output byte/word to DX port          | 11110010 | 0110111w    |             |             | 5+9n      |       |
| <b>CMPS</b> = Compare byte/word                    | 1111011z | 1010011w    |             |             | 4+18n     |       |
| <b>SCAS</b> = Scan byte/word                       | 1111001z | 1010111w    |             |             | 4+13n     |       |
| <b>LODS</b> = Load byte/word to AL/AX              | 11110010 | 0101001w    |             |             | 3+9n      |       |
| <b>STOS</b> = Store byte/word from AL/AX           | 11110100 | 0101001w    |             |             | 4+3n      |       |
| <b>PROGRAM TRANSFER INSTRUCTIONS</b>               |          |             |             |             |           |       |
| <b>Conditional Transfers ; Jump if:</b>            |          |             |             |             |           |       |
| <b>JE/JZ</b> = equal/zero                          | 01110100 | disp        |             |             | 1/9       |       |
| <b>JL/JNGE</b> = less/not greater or equal         | 01111100 | disp        |             |             | 1/9       |       |
| <b>JLE/JNG</b> = less or equal/not greater         | 01111110 | disp        |             |             | 1/9       |       |
| <b>JC/JB/JNAE</b> = carry/below/not above or equal | 01110010 | disp        |             |             | 1/9       |       |
| <b>JBE/JNA</b> = below or equal/not above          | 01110110 | disp        |             |             | 1/9       |       |
| <b>JP/JPE</b> = parity/parity even                 | 01111010 | disp        |             |             | 1/9       |       |
| <b>JO</b> = overflow                               | 01110000 | disp        |             |             | 1/9       |       |
| <b>JS</b> = sign                                   | 01111000 | disp        |             |             | 1/9       |       |
| <b>JNE/JNZ</b> = not equal/not zero                | 01110101 | disp        |             |             | 1/9       |       |
| <b>JNL/JGE</b> = not less/greater or equal         | 01111101 | disp        |             |             | 1/9       |       |
| <b>JNLE/JG</b> = not less or equal/greater         | 01111111 | disp        |             |             | 1/9       |       |
| <b>JNC/JNB/JAE</b> = not carry/not below           | 01110011 | disp        |             |             | 1/9       |       |
| /above or equal                                    |          |             |             |             |           |       |
| <b>JNBE/JA</b> = not below or equal/above          | 01110111 | disp        |             |             | 1/9       |       |
| <b>JNP/JPO</b> = not parity/parity odd             | 01111011 | disp        |             |             | 1/9       |       |
| <b>JNO</b> = not overflow                          | 01110001 | disp        |             |             | 1/9       |       |
| <b>JNS</b> = not sign                              | 01111001 | disp        |             |             | 1/9       |       |

| Function                                         | Format                           | Clocks | Notes |
|--------------------------------------------------|----------------------------------|--------|-------|
| <b>Unconditional Transfers</b>                   |                                  |        |       |
| <b>CALL</b> = Call procedure                     |                                  |        |       |
| direct within segment                            | 11101000 disp-low disp-high      | 11     |       |
| reg/memory indirect within segment               | 11111111 mod 010 r/m             | 12/17  |       |
| indirect intersegment                            | 11111111 mod 011 r/m (mod ? 11)  | 25     |       |
| direct intersegment                              | 10011010 segment offset selector | 18     |       |
| <b>RET = Return from procedure</b>               |                                  |        |       |
| within segment                                   | 11000011                         | 16     |       |
| within segment adding immed to SP                | 11000010 data-low data-high      | 16     |       |
| intersegment                                     | 11001011                         | 23     |       |
| intersegment adding immed to SP                  | 1001010 data-low data-high       | 23     |       |
| <b>JMP = Unconditional jump</b>                  |                                  |        |       |
| short/long                                       | 11101011 disp-low                | 9/9    |       |
| direct within segment                            | 11101001 disp-low disp-high      | 9      |       |
| reg/memory indirect within segment               | 11111111 mod 100 r/m             | 11/16  |       |
| indirect intersegment                            | 11111111 mod 101 r/m (mod ? 11)  | 18     |       |
| direct intersegment                              | 11101010 segment offset selector | 11     |       |
| <b>Iteration Control</b>                         |                                  |        |       |
| <b>LOOP</b> = Loop CX times                      | 11100010 disp                    | 7/16   |       |
| <b>LOOPZ/LOOPE</b> = Loop while zero/equal       | 11100001 disp                    | 7/16   |       |
| <b>LOOPNZ/LOOPNE</b> = Loop while not zero/equal | 11100000 disp                    | 7/16   |       |
| <b>JCXZ</b> = Jump if CX = zero                  | 11100011 disp                    | 7/15   |       |
| <b>Interrupt</b>                                 |                                  |        |       |
| <b>INT</b> = Interrupt                           |                                  |        |       |
| Type specified                                   | 11001101 type                    | 41     |       |
| Type 3                                           | 11001100                         | 41     |       |
| <b>INTO</b> = Interrupt on overflow              | 11001110                         | 43/4   |       |
| <b>BOUND</b> = Detect value out of range         | 01100010 mod reg r/m             | 21-60  |       |
| <b>IRET</b> = Interrupt return                   | 11001111                         | 31     |       |
| <b>PROCESSOR CONTROL INSTRUCTIONS</b>            |                                  |        |       |
| <b>CLC</b> = clear carry                         | 11111000                         | 2      |       |
| <b>CMC</b> = Complement carry                    | 11110101                         | 2      |       |
| <b>STC</b> = Set carry                           | 11111001                         | 2      |       |
| <b>CLD</b> = Clear direction                     | 11111100                         | 2      |       |
| <b>STD</b> = Set direction                       | 11111101                         | 2      |       |
| <b>CLI</b> = Clear interrupt                     | 11111010                         | 5      |       |
| <b>STI</b> = Set interrupt                       | 11111011                         | 5      |       |
| <b>HLT</b> = Halt                                | 11110100                         | 1      |       |
| <b>WAIT</b> = Wait                               | 10011011                         | 1      |       |
| <b>LOCK</b> = Bus lock prefix                    | 11110000                         | 1      |       |
| <b>ESC</b> = Math coprocessor escape             | 11011MMM mod PPP r/m             | 1      |       |
| <b>NOP</b> = No operation                        | 10010000                         | 1      |       |
| <b>SEGMENT OVERRIDE PREFIX</b>                   |                                  |        |       |
| CS                                               | 00101110                         | 2      |       |
| SS                                               | 00110110                         | 2      |       |
| DS                                               | 00111110                         | 2      |       |
| ES                                               | 00100110                         | 2      |       |

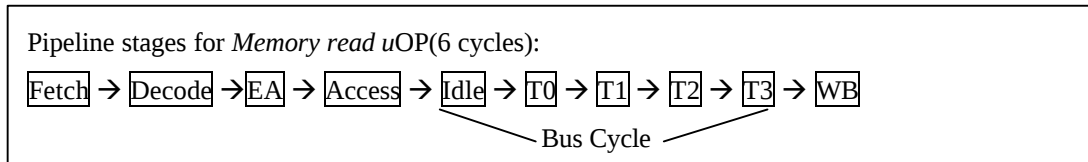
## 20.1 R8820LV Execution Timings

The above instruction timing represent the minimum execution time in clock cycles for each instruction. The timings given are based on the following assumptions:

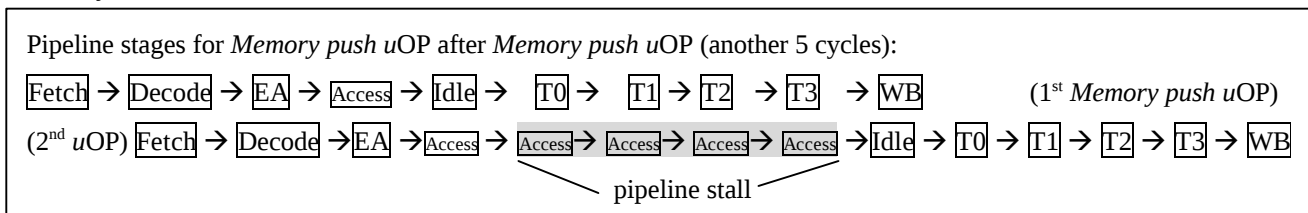
1. The opcode, along with and data or displacement required for execution, has been prefetched and resides in the instruction queue at the time is needed.
2. No wait states or bus HOLDS occur.
3. All word -data is located on even-address boundaries.
4. One RISC micro operation(uOP) maps one cycle(according the pipeline stages described below) , except the following case:



4.1 Memory read uOP need 6 cycles for bus.

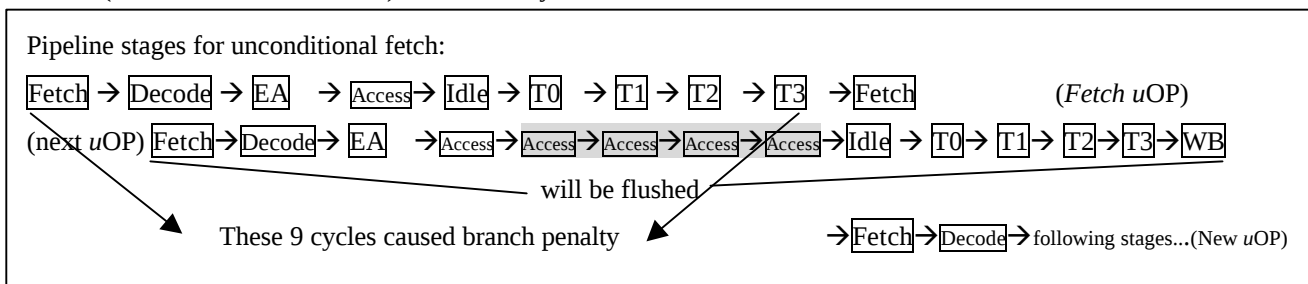


4.2 Memory push uOP need 1 cycle if it has no previous Memory push uOP, and 5 cycles if it has previous Memory push or Memory Write uOP.



4.3 MUL uOP and DIV of ALU function uOP for 8 bits operation need both 8 cycles, for 16 bits operation need both 16 cycles.

4.4 All jumps, calls, ret and loopXX instructions required to fetch the next instruction for the destination address(Unconditional Fetch uOP) will need 9 cycles.



**Note:** op\_r: operand read stage, EA: Calculate Effective Address stage, Idle: Bus Idle stage, T0..T3: Bus T0..T3 stage, Access: Access data from cache memory stage.

## 21. DC Characteristics

### Absolute Maximum Rating

| Symbol | Rating                               | Commercial        | Unit              | Note |
|--------|--------------------------------------|-------------------|-------------------|------|
| Vterm  | Terminal Voltage with Respect To GND | -0.5 to Vcc+0.5 V | V                 |      |
| Ta     | Operating Temperature                | 0 to +70          | <b>Centigrade</b> |      |
| Pt     | Power Dissipation                    | 1.5               | W                 |      |

### Recommended DC Operating Conditions

| Symbol | Parameter               | Min. | Typ. | Max.    | Unit |
|--------|-------------------------|------|------|---------|------|
| Vcc    | Supply Voltage          | 3.0  | 3.3  | 3.6     | V    |
| GND    | Ground                  | 0    | 0    | 0       | V    |
| Vih    | Input High Voltage(1)   | 2.0  | ---  | Vcc+0.5 | V    |
| Vih1   | Input High Voltage(RES) | 2.5  |      | Vcc+0.5 | V    |
| Vih2   | Input High Voltage (X1) | 2.5  |      | Vcc+0.5 | V    |
| Vil    | Input Low voltage       | -0.5 | 0    | 0.8     | V    |

Note 1: RST ,X1 pins not included

### DC Electrical Characteristics

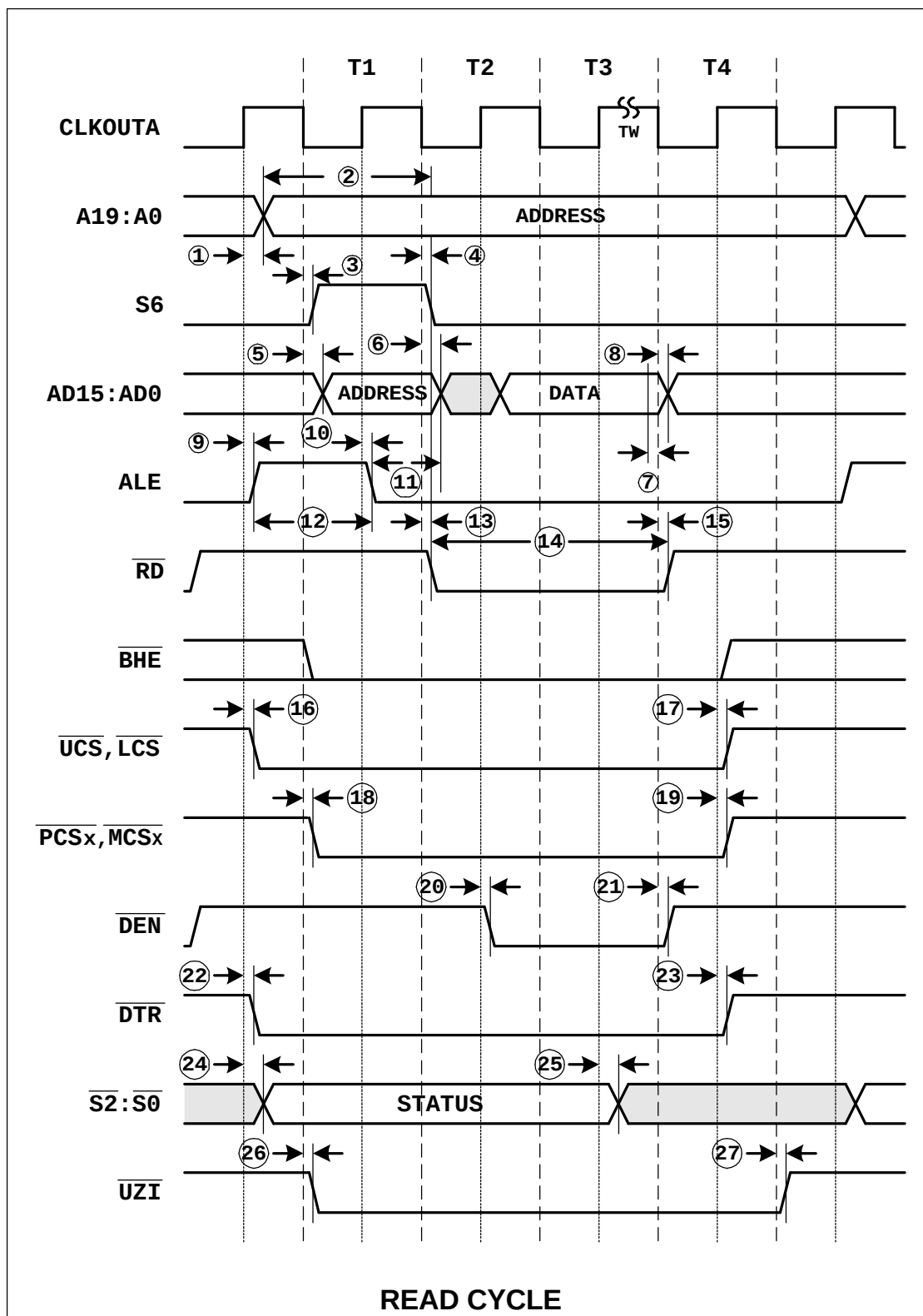
| Symbol | Parameter                               | Test Condition             | Min | Max | Unit |
|--------|-----------------------------------------|----------------------------|-----|-----|------|
| Ili    | Input Leakage Current (for 32 Pio Pins) | Vcc=Vmax<br>Vin=GND to Vcc |     | 300 | uA   |
| Ili    | Input Leakage Current (Others)          | Vcc=Vmax<br>Vin=GND to Vcc |     | 80  | uA   |
| Ilo    | Output Leakage Current                  | Vcc=Vmax<br>Vin=GND to Vcc |     | 300 | uA   |
|        |                                         |                            |     |     |      |
| VOL    | Output Low Voltage                      | Iol=2mA,<br>Vcc=Min.       | —   | 0.4 | V    |
| VOH    | Output High Voltage                     | Ioh=-2.4mA,<br>Vcc=Min.    | 2.4 | —   | V    |

Note1: Vmax=3.6V Vmin=3.0V

**DC Electrical Characteristics**

| Symbol | Parameter                     | Test condition     | Min | Max | Unit | Note     |
|--------|-------------------------------|--------------------|-----|-----|------|----------|
| Icc    | Max Operating Current         | Vcc=3.6V,<br>33MHz | --- | 85  | mA   |          |
| Fmax   | Max operation clock frequency |                    | 5   | 33  | Mhz  | Vcc+-5%  |
| Fmax   | Max operation clock frequency |                    | 5   | 25  | Mhz  | Vcc+-10% |

## 22.AC Characteristics

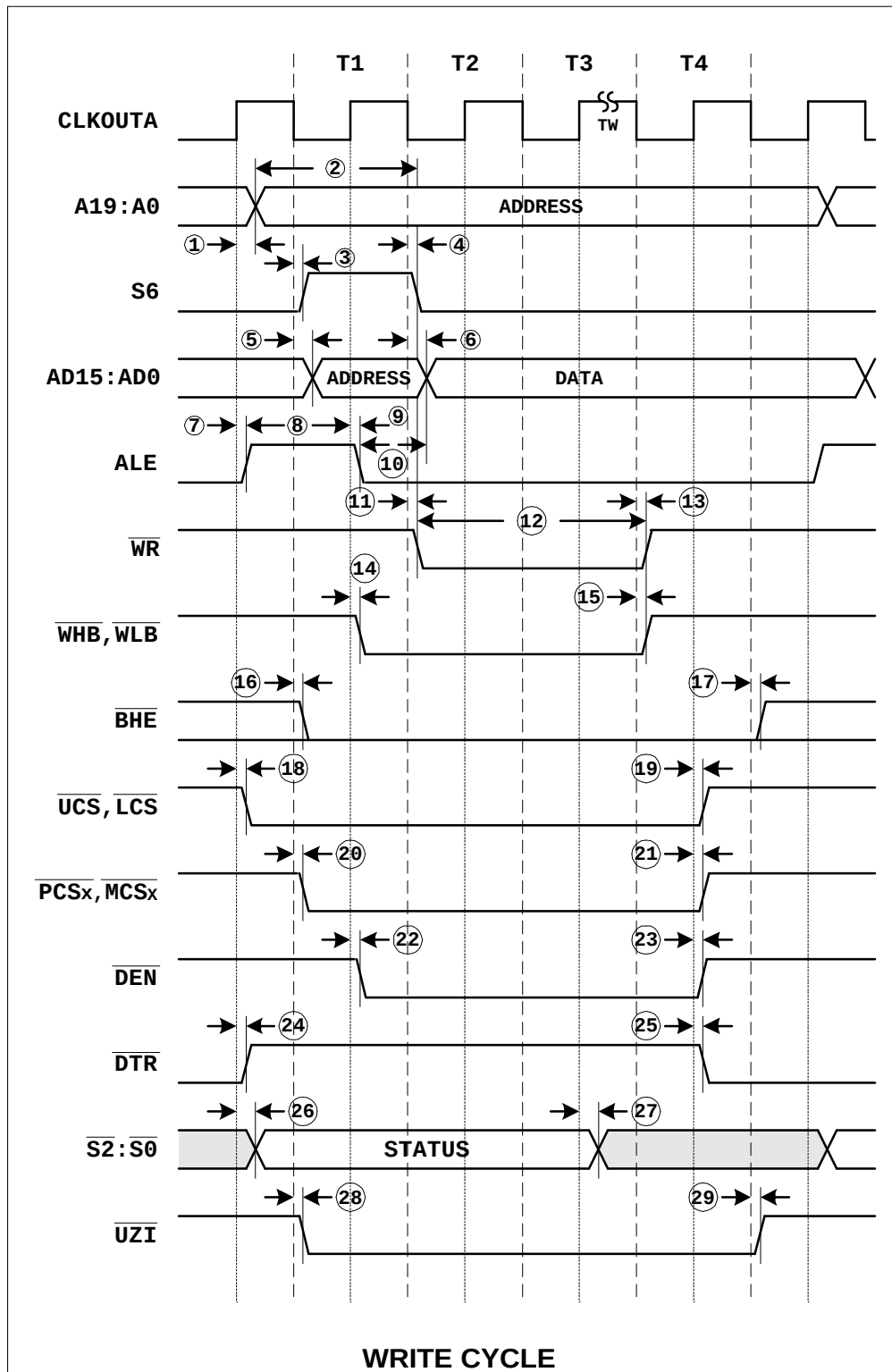


| No. | Description                                                           | MIN     | MAX | Unit |
|-----|-----------------------------------------------------------------------|---------|-----|------|
| 1   | CLKOUTA high to A Address Valid                                       | 0       | 15  | ns   |
| 2   | A address valid to $\overline{\text{RD}}$ low                         | 1.5T-12 |     | ns   |
| 3   | S6 active delay                                                       | 0       | 20  | ns   |
| 4   | S6 inactive delay                                                     | 0       | 20  | ns   |
| 5   | AD address Valid Delay                                                | 0       | 20  | ns   |
| 6   | Address Hold                                                          | 0       | 12  | ns   |
| 7   | Data in setup                                                         | 10      |     | ns   |
| 8   | Data in Hold                                                          | 3       |     | ns   |
| 9   | ALE active delay                                                      | 0       | 20  | ns   |
| 10  | ALE inactive delay                                                    | 0       | 20  | ns   |
| 11  | Address Valid after ALE inactive                                      | 1/2T-10 |     | ns   |
| 12  | ALE width                                                             | T-10    |     | ns   |
| 13  | $\overline{\text{RD}}$ active delay                                   | 0       | 15  | ns   |
| 14  | $\overline{\text{RD}}$ Pulse Width                                    | 2T-15   |     | ns   |
| 15  | $\overline{\text{RD}}$ inactive delay                                 | 0       | 20  | ns   |
| 16  | CLKOUTA HIGH to $\overline{\text{LCS}}$ $\overline{\text{UCS}}$ valid | 0       | 20  | ns   |
| 17  | $\overline{\text{UCS}}$ , $\overline{\text{LCS}}$ inactive delay      | 0       | 20  | ns   |
| 18  | $\overline{\text{PCS}}$ , $\overline{\text{MCS}}$ active delay        | 0       | 20  | ns   |
| 19  | $\overline{\text{PCS}}$ , $\overline{\text{MCS}}$ inactive delay      | 0       | 20  | ns   |
| 20  | $\overline{\text{DEN}}$ active delay                                  | 0       | 20  | ns   |
| 21  | $\overline{\text{DEN}}$ inactive delay                                | 0       | 20  | ns   |
| 22  | $\overline{\text{DTR}}$ active delay                                  | 0       | 20  | ns   |
| 23  | $\overline{\text{DTR}}$ inactive delay                                | 0       | 20  | ns   |
| 24  | Status active delay                                                   | 0       | 20  | ns   |
| 25  | Status inactive delay                                                 | 0       | 20  | ns   |
| 26  | $\overline{\text{UZI}}$ active delay                                  | 0       | 20  | ns   |
| 27  | $\overline{\text{UZI}}$ inactive delay                                | 0       | 20  | ns   |

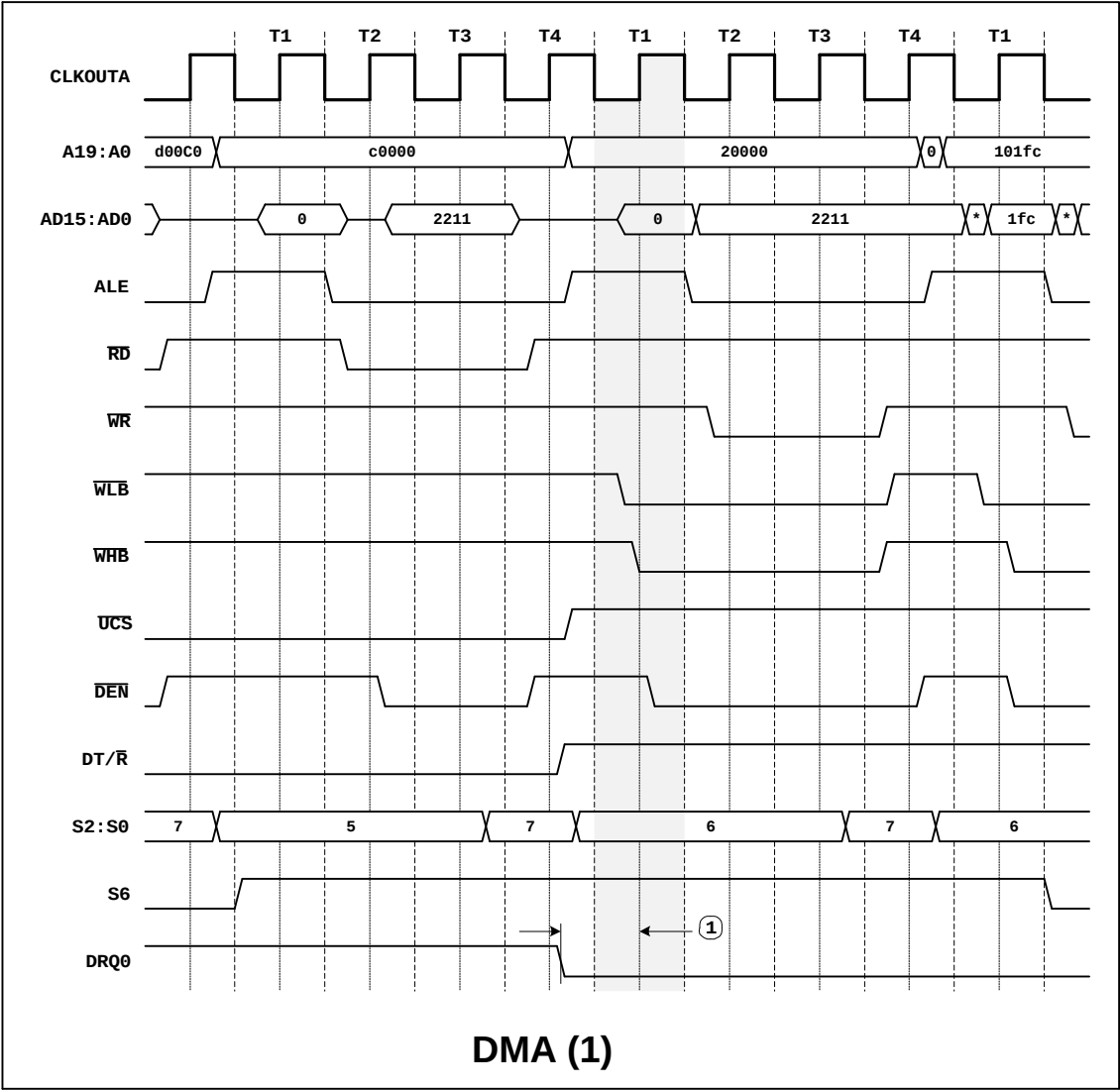
1. T means a clock period time

2. All timing parameters are measured at 1.5V with 50 PF loading on CLKOUTA

All output test conditions are with CL=50 pF

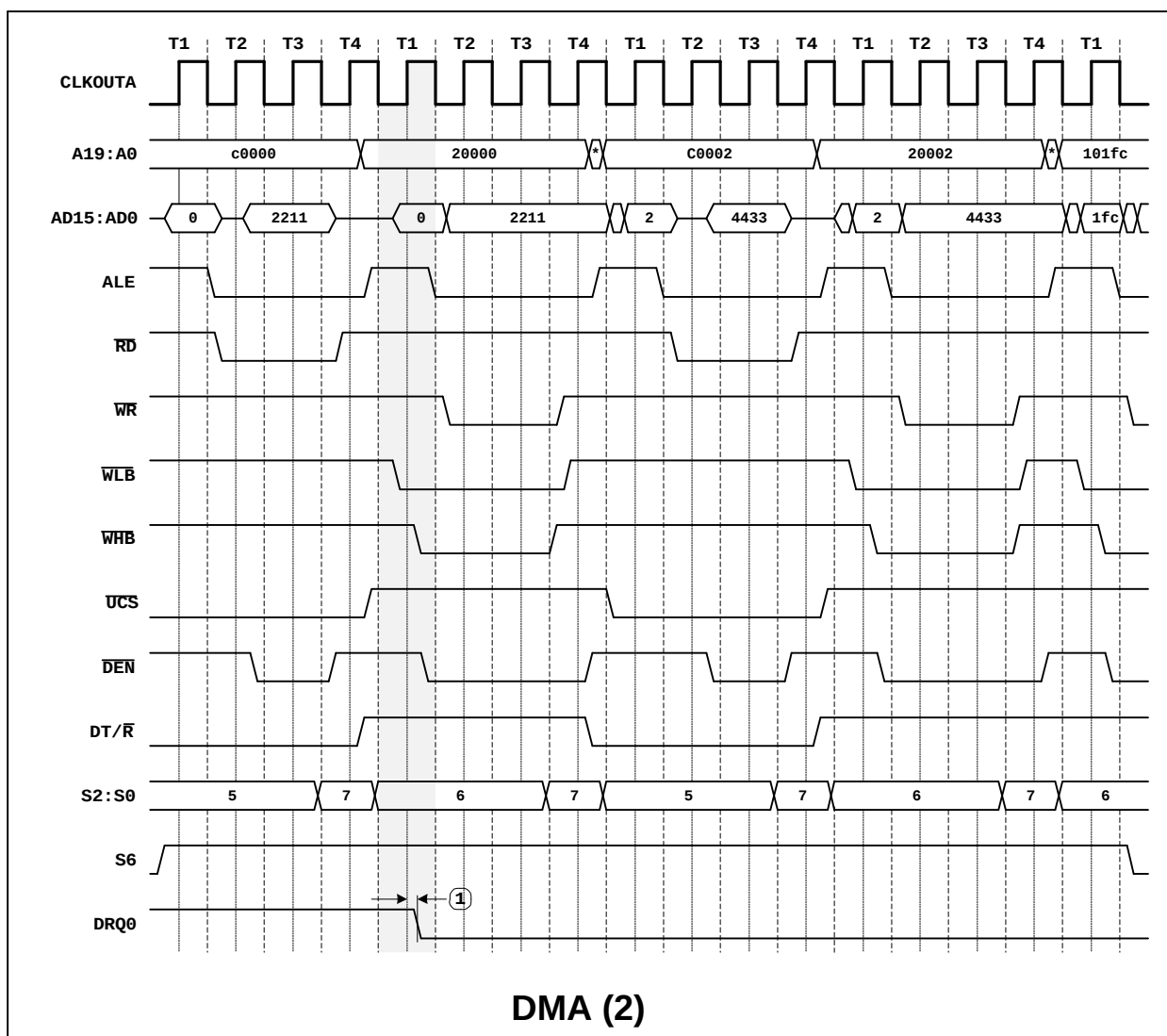


| No. | Description                                                      | MIN     | MAX | Unit |
|-----|------------------------------------------------------------------|---------|-----|------|
| 1   | CLKOUTA high to A Address Valid                                  | 0       | 15  | ns   |
| 2   | A address valid to $\overline{\text{WR}}$ low                    | 1.5T-12 |     | ns   |
| 3   | S6 active delay                                                  | 0       | 20  | ns   |
| 4   | S6 inactive delay                                                | 0       | 20  | ns   |
| 5   | AD address Valid Delay                                           | 0       | 15  | ns   |
| 6   | Address Hold                                                     |         |     | ns   |
| 7   | ALE active delay                                                 | 0       | 20  | ns   |
| 8   | ALE width                                                        | T-10    |     | ns   |
| 9   | ALE inactive delay                                               | 0       | 20  | ns   |
| 10  | Address valid after ALE inactive                                 | 1/2T-10 |     | ns   |
| 11  | $\overline{\text{WR}}$ active delay                              | 0       | 15  | ns   |
| 12  | $\overline{\text{WR}}$ pulse width                               | 2T-15   |     | ns   |
| 13  | $\overline{\text{WR}}$ inactive delay                            | 0       | 15  | ns   |
| 14  | WHB , $\overline{\text{WLB}}$ active delay                       | 0       | 20  | ns   |
| 15  | WHB , $\overline{\text{WLB}}$ inactive delay                     | 0       | 20  | ns   |
| 16  | BHE active delay                                                 | 0       | 20  | ns   |
| 17  | BHE inactive delay                                               | 0       | 20  | ns   |
| 18  | CLKOUTA high to UCS , $\overline{\text{LCS}}$ valid              | 0       | 20  | ns   |
| 19  | UCS , $\overline{\text{LCS}}$ inactive delay                     | 0       | 20  | ns   |
| 20  | $\overline{\text{PCS}}$ , $\overline{\text{MCS}}$ active delay   | 0       | 20  | ns   |
| 21  | $\overline{\text{PCS}}$ , $\overline{\text{MCS}}$ inactive delay | 0       | 20  | ns   |
| 22  | $\overline{\text{DEN}}$ active delay                             | 0       | 20  | ns   |
| 23  | $\overline{\text{DEN}}$ inactive delay                           | 0       | 20  | ns   |
| 24  | $\overline{\text{DTR}}$ active delay                             | 0       | 20  | ns   |
| 25  | $\overline{\text{DTR}}$ inactive delay                           | 0       | 20  | ns   |
| 26  | Status active delay                                              | 0       | 20  | ns   |
| 27  | Status inactive delay                                            | 0       | 20  | ns   |
| 28  | $\overline{\text{UZI}}$ active delay                             | 0       | 20  | ns   |
| 29  | $\overline{\text{UZI}}$ inactive delay                           | 0       | 20  | ns   |



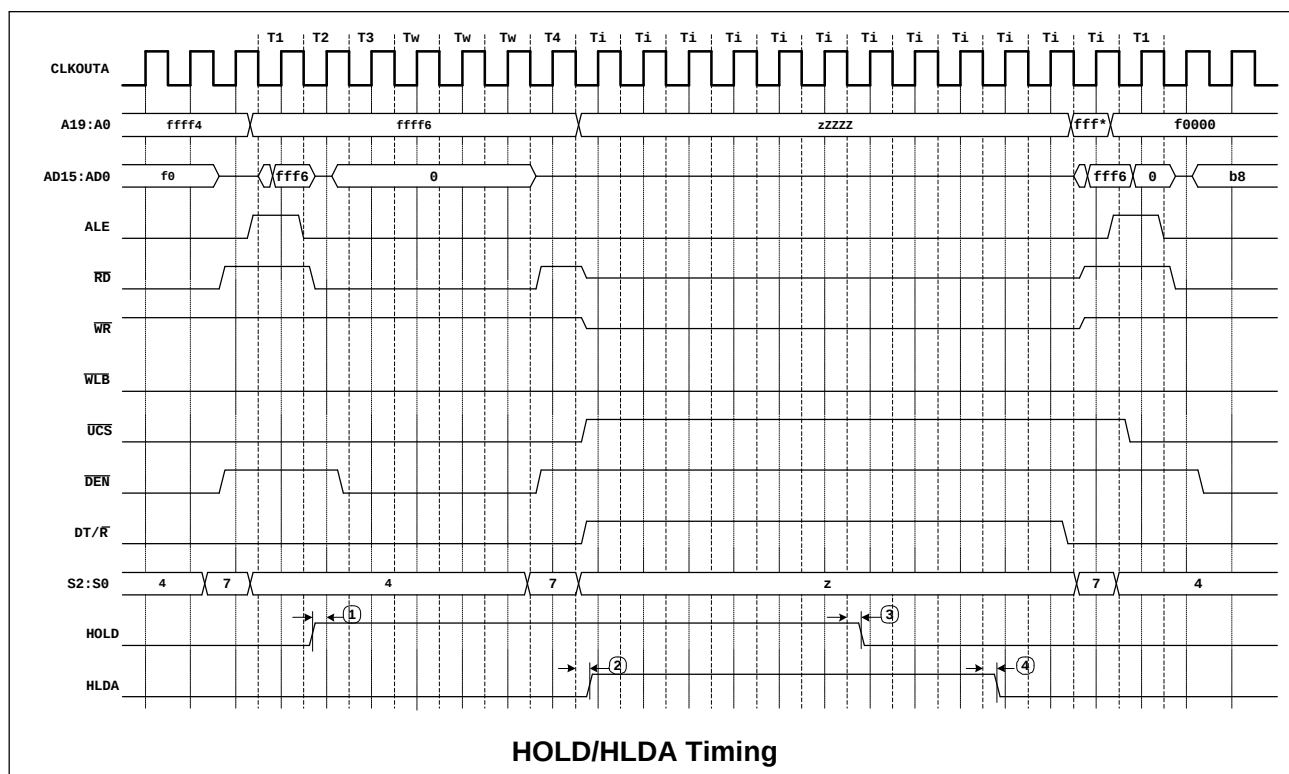
\* The source-synchronized transfer is not followed immediately by another DMA transfer

| No. | Description           | MIN | MAX | Unit |
|-----|-----------------------|-----|-----|------|
| 1   | DRQ is confirmed time | 0   | 10  | ns   |

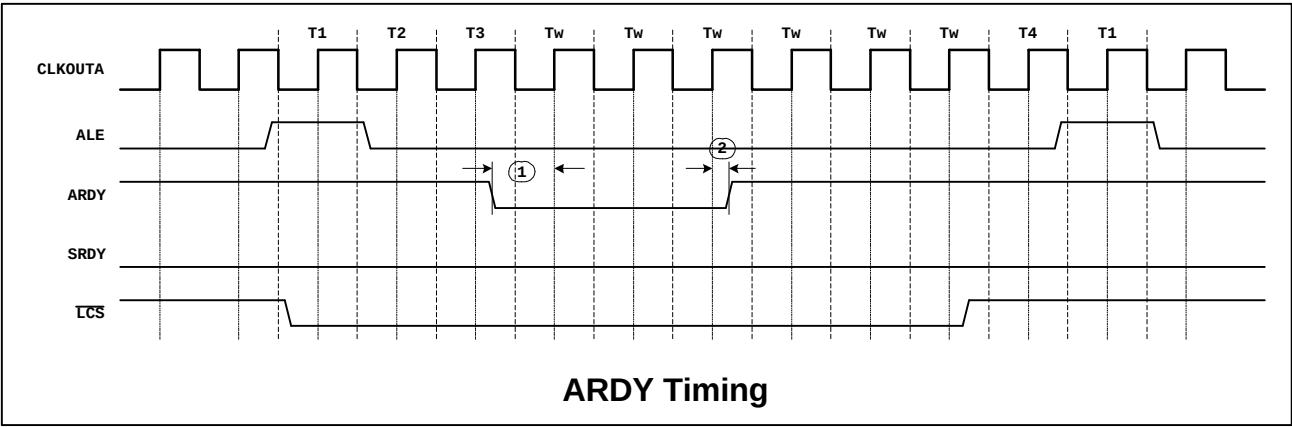


\* The source-synchronized transfer is followed immediately by another DMA transfer

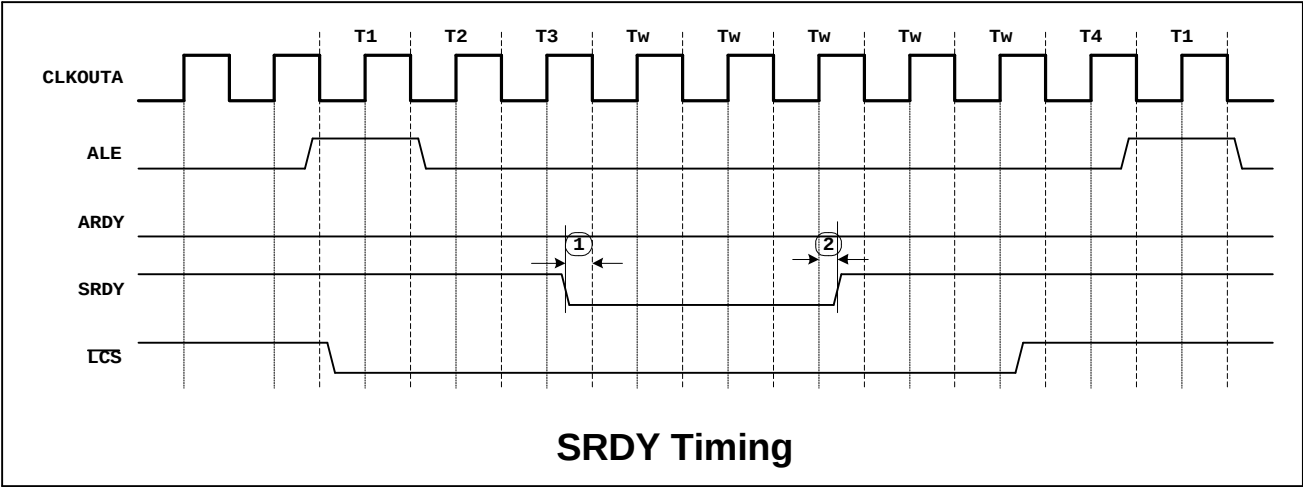
| No. | Description           | MIN | MAX | Unit |
|-----|-----------------------|-----|-----|------|
| 1   | DRQ is confirmed time | 0   | 3   | ns   |



| No. | Description      | MIN | MAX | Unit |
|-----|------------------|-----|-----|------|
| 1   | HOLD setup time  | 0   | 10  | ns   |
| 2   | HLDA Valid Delay | 0   | 20  | ns   |
| 3   | HOLD hold time   | 0   | 3   | ns   |
| 4   | HLDA Valid Delay | 0   | 20  | ns   |

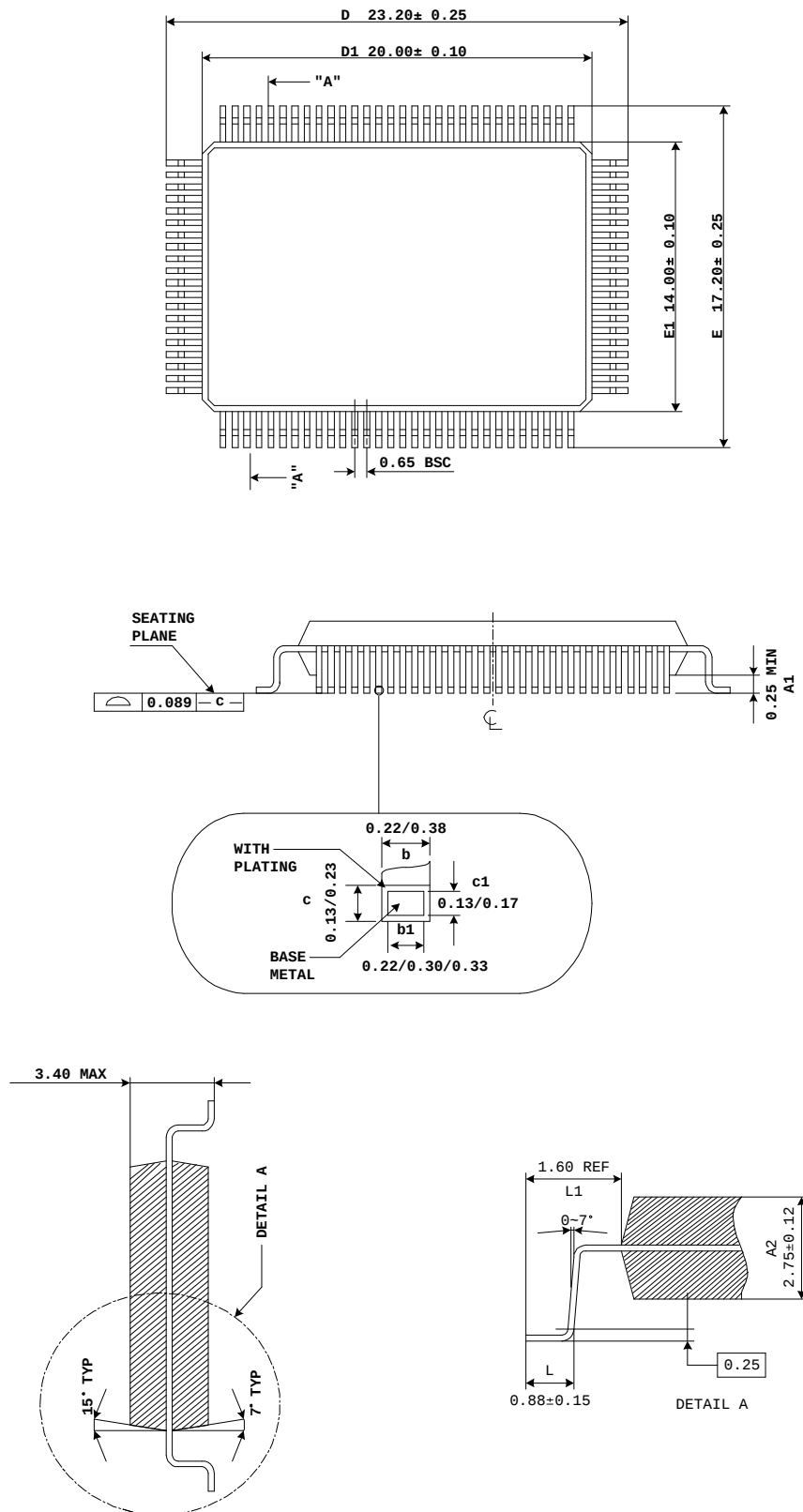


| No. | Description                           | MIN | MAX | Unit |
|-----|---------------------------------------|-----|-----|------|
| 1   | ARDY Resolution Transition setup time | 0   | 10  | ns   |
| 2   | ARDY active hold time                 | 0   | 10  | ns   |

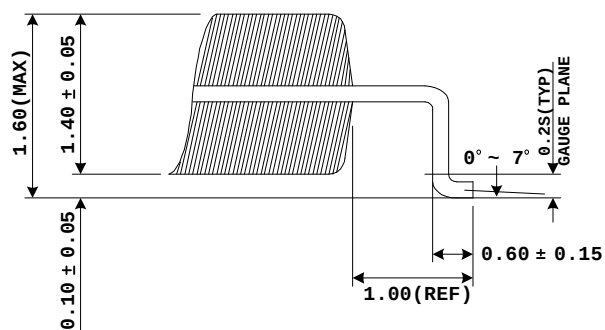
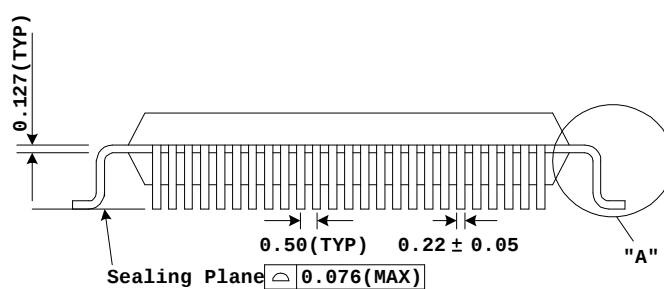
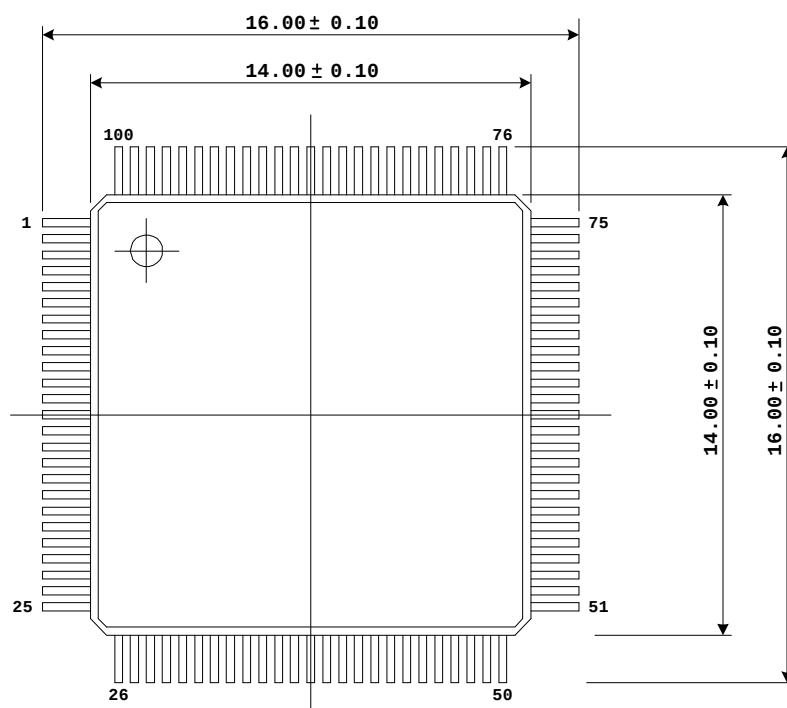


| No. | Description                | MIN | MAX | Unit |
|-----|----------------------------|-----|-----|------|
| 1   | SRDY transition setup time | 0   | 10  | ns   |
| 2   | SRDY transition hold time  | 0   | 3   | ns   |

## 23. PACKAGE INFORMATION (PQFP)



**(LQFP)**



**UNIT : mm**