Features

- On board Enhanced Super I/O Controller
- Provide Configurable area for each device on
  - Address mapping
  - IRQ channel routing
  - DMA channel routing
- Support the floppy disk upto 2.88MB
- FDD re-route to Parallel port
- Support 3-mode FDD
- Two high speed serial ports with the IrDA and ASKIR Supporting
- MIDI bit rate supporting on serial port
- Multi-mode parallel port supporting on ECP/EPP/SPP
- IDE/Game port interface decoder output
- Power Management supporting
- 100 PQFP

General Description

HT8669IR is a high integrated I/O device. It supports a floppy disk controller, a multi-mode printer port, two high speed serial communication ports, one of which is enriched to support IrDA SIR and ASKIR transmission. By setting the different configuration, HT8669IR can also support IDE and game port interfaces.

There are some configuration register sets to reconfigure the ISA address, IRQ access channel, and DRQ channel for each device in order to support compatible function. The floppy disk controller supports the disk capacity up to 2.88MB with 3-mode floppy disk hardware interface. The disk interface can be re-routed to printer port for some specific applications. In the print port interface, it supports standard mode, PC/AT or PS/2 mode, Enhanced Parallel Port (EPP) 1.7/1.9, or Enhanced Capabilities Port (ECP). For the serial communication interface, there are two high speed ports for serial communication with the MIDI rate supports. One of which is expanded to support IrDA SIR or ASKIR transmission. By using the infrared interface, this device can support wireless communication easily.
HT8669IR

Pin Assignment

DDEN0 1
nMOE0 2
nDS1 3
nDS0 4
nMOE1 5
GND 6
nDIR 7
nSTEP 8
nWD0 9
nWG 10
nHSEL 11
nDX 12
nTRK0 13
nWP 14
VCC 15
nRDG 16
nDFG 17
nDERST:DDEN 18
IRQ_A 19
CLK24 20
DRQ_A:DE7 21
nDACK_A:IRQ_IN 22
nDEEN:IRQ_H 24
nDECS0:IRSO 25
nDECS1:IRSI 26
nCS 27
HA0 28
HAI 29
HA2 30

HT8669IR

3rd Jan '97
## Host Interface

<table>
<thead>
<tr>
<th>Pin No.</th>
<th>Pin Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>97, [43:41], [34:28]</td>
<td>HA[10:0]</td>
<td>I</td>
<td>Host I/O Address: For internal decoder use. The contents are latched internally by the leading edge of nIOR or nIOW.</td>
</tr>
<tr>
<td>[56:53], [51:48]</td>
<td>HD[7:0]</td>
<td>I/O24</td>
<td>Host I/O Data: For data accesses. These pins are Hi-Z when no output.</td>
</tr>
<tr>
<td>46</td>
<td>HAEN</td>
<td>I</td>
<td>Host Address Enable: For indicating DMA operation and internal address decode qualification.</td>
</tr>
<tr>
<td>44</td>
<td>nIOR</td>
<td>I</td>
<td>I/O Read: For host read operation.</td>
</tr>
<tr>
<td>45</td>
<td>nIOW</td>
<td>I</td>
<td>I/O Write: For host write operation.</td>
</tr>
<tr>
<td>100</td>
<td>IOCHRDY</td>
<td>OD8U</td>
<td>I/O Channel Ready: It is used to extend the host command in EPP mode. It is internal pull-up.</td>
</tr>
<tr>
<td>22</td>
<td>nDACK_A</td>
<td>I</td>
<td>DMA Acknowledgement: Host acknowledge the DMA request for transferring.</td>
</tr>
<tr>
<td>36</td>
<td>nDACK_B</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>96</td>
<td>nDACK_C</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>DRQ_A/</td>
<td>O24ID</td>
<td>DMA Request: This pin is used to request host a DMA transferring. It will be cleared on the last data transfer by the nDACK/nIOR being low.</td>
</tr>
<tr>
<td>52</td>
<td>DRQ_B/</td>
<td>O24ID</td>
<td></td>
</tr>
<tr>
<td>99</td>
<td>DRQ_C</td>
<td>O24</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>TC</td>
<td>I</td>
<td>Terminal Count: It indicates the DMA transfer is complete.</td>
</tr>
<tr>
<td>23</td>
<td>IRQ_IN</td>
<td>I</td>
<td>IRQ Input: An external IRQ input to the chip for IRQ router.</td>
</tr>
<tr>
<td>19</td>
<td>IRQ_A</td>
<td>O24</td>
<td></td>
</tr>
<tr>
<td>94</td>
<td>IRQ_B</td>
<td>O24</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>IRQ_C</td>
<td>O24</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>IRQ_D</td>
<td>O24</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>IRQ_E</td>
<td>O24</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>IRQ_F</td>
<td>O24</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>IRQ_H</td>
<td>O24</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>nCS</td>
<td>I</td>
<td>Chip Select: External decoder input for selecting this device.</td>
</tr>
<tr>
<td>57</td>
<td>RESET</td>
<td>IS</td>
<td>System Reset: It is a reset input with a 500ns minimum active pulse for internal egisters reset. The configuration registers are unaffected.</td>
</tr>
<tr>
<td>Pin No.</td>
<td>Pin Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>16</td>
<td>nRDD</td>
<td>IS</td>
<td>Read Disk Data: Raw serial disk data coming from disk presents a flux transition on each falling edge.</td>
</tr>
<tr>
<td>9</td>
<td>nWDD</td>
<td>OD48</td>
<td>Write Disk Data: Encoded disk data stream for disk write.</td>
</tr>
<tr>
<td>10</td>
<td>nWG</td>
<td>OD48</td>
<td>Write Gate: For disk write head operation.</td>
</tr>
<tr>
<td>17</td>
<td>nDCHG</td>
<td>IS</td>
<td>Disk Changed: Indicate drive door is open.</td>
</tr>
<tr>
<td>14</td>
<td>nWP</td>
<td>IS</td>
<td>Write Protect: For disk status indication on write protection.</td>
</tr>
<tr>
<td>13</td>
<td>nTRK0</td>
<td>IS</td>
<td>Track 00: For disk status indication on track 0 being sensed.</td>
</tr>
<tr>
<td>12</td>
<td>nIDX</td>
<td>IS</td>
<td>Index Hole: For disk status indication on index hole being sensed.</td>
</tr>
<tr>
<td>11</td>
<td>nHSEL</td>
<td>OD48</td>
<td>Head Select: For disk head selection. A logic &quot;1&quot; means side 0 and a logic &quot;0&quot; means side 1.</td>
</tr>
<tr>
<td>7</td>
<td>nDIR</td>
<td>OD48</td>
<td>Direction Control: For disk head direction control. A logic &quot;1&quot; means inward motion and a logic &quot;0&quot; means outward motion.</td>
</tr>
<tr>
<td>8</td>
<td>nSTEP</td>
<td>OD48</td>
<td>Step Pulse: A pulse sequence output for track-to-track operation.</td>
</tr>
<tr>
<td>3, 4</td>
<td>nDS[1:0]</td>
<td>OD48</td>
<td>Drive Selects: For disk driver selection.</td>
</tr>
<tr>
<td>5, 2</td>
<td>nMOE[1:0]</td>
<td>OD48</td>
<td>Motor On: For disk motor control.</td>
</tr>
<tr>
<td>18</td>
<td>DDEN1</td>
<td>OD48</td>
<td>Driver Density(Reduce Write Current): Select drive and media. Refer to CR03, CR0B, and CR1F.</td>
</tr>
<tr>
<td>1</td>
<td>DDEN0</td>
<td>OD48</td>
<td></td>
</tr>
</tbody>
</table>

5  3rd Jan '97
<table>
<thead>
<tr>
<th>Pin No.</th>
<th>Pin Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>78</td>
<td>S11</td>
<td>I</td>
<td>Serial Data In: Received serial data input.</td>
</tr>
<tr>
<td></td>
<td>S12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>79</td>
<td>S01</td>
<td>O4</td>
<td>Serial Data Out: Transmit serial data output.</td>
</tr>
<tr>
<td></td>
<td>S02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>81</td>
<td>nRTS1</td>
<td>O4</td>
<td>Request To Send: Handshake output signals notify modem that the UARTn is ready to transmit data. It can be programmed by writing to rts.CMn_MCR. It will be reset to inactive mode during hardware reset or forced to inactive during loop mode operation.</td>
</tr>
<tr>
<td></td>
<td>nRTS2</td>
<td>OT4</td>
<td></td>
</tr>
<tr>
<td>83</td>
<td>nDTR1</td>
<td>O4</td>
<td>Data Terminal Ready: Handshake output signals notify modem that the UARTn is ready to setup data communication link. It can be programmed by writing to dtr.CMn_MCR. It will be reset to inactive mode during hardware reset or forced to inactive during loop mode operation.</td>
</tr>
<tr>
<td></td>
<td>nDTR2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>82</td>
<td>nCTS1</td>
<td>I</td>
<td>Clear To Send: Handshake input signals notify UARTn that the modem is ready to receive data. An nCTSn signal state change from low to high after the last CMn_MSR read will set dcts.CMn_MSR to ’1’. If emsi.CMn_IER is set, it will generate an interrupt when nCTSn changes state. The CPU can monitor the status of nCTSn by reading cts.CMn_MSR. The bit is the complement of nCTSn.</td>
</tr>
<tr>
<td></td>
<td>nCTS2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>nDSR1</td>
<td>I</td>
<td>Data Send Ready: Handshake input signals notify UARTn that the modem is ready to setup the data communication link. An nDSRn signal state change from low to high after the last CMn_MSR read will set ddsr.CMn_MSR to ’1’. If emsi.CMn_IER is set, it will generate an interrupt when nDSRn changes state. The CPU can monitor the status of nDSRn by reading dsr.CMn_MSR. The bit is the complement onDSRn.</td>
</tr>
<tr>
<td></td>
<td>nDSR2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>85</td>
<td>nDCD1</td>
<td>I</td>
<td>Data Carrier Detect: Handshake input signals notify UARTn that carrier signal is detected by the modem. An nDCDn signal state change from low to high after the last CMn_MSR read will set dcd.CMn_MSR to ’1’. If emsi.CMn_IER is set, it will generate an interrupt when nDCDn changes state. The CPU can monitor the status of nDCDn by reading dcd.CMn_MSR. The bit is the complement of nDCDn.</td>
</tr>
<tr>
<td></td>
<td>nDCD2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>84</td>
<td>nRI1</td>
<td>I</td>
<td>Ring Indicator: Handshake input signals notify UARTn that the telephone ring signal is detected by the modem. An nRIn signal state change from low to high after the last CMn_MSR read will set teri.CMn_MSR to ’1’. If emsi.CMn_IER is set, it will generate an interrupt when nRIn changes state. The CPU can monitor the status of nRIn by reading rj.CMn_MSR. The bit is the complement of nRIn.</td>
</tr>
<tr>
<td></td>
<td>nRI2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Parallel Port

<table>
<thead>
<tr>
<th>Pin No.</th>
<th>Pin Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[63:66], [68:71]</td>
<td>PD[7:0]</td>
<td>I/O24</td>
<td>Parallel Port Data I/O: The bi-directional parallel port data for data transfer between HOST and peripherals. It contains either address or data in EPP or ECP mode, the data may include RLE data in ECP mode.</td>
</tr>
<tr>
<td>61</td>
<td>BUSY</td>
<td>I</td>
<td>Line Busy: A busy signal from printer to indicate printer is not available to receive the new data. The bit busy. SPP_SPR is the complement of this input. nWAIT(Wait): In EPP mode, it is active low to indicate the device is ready for the next transfer. BUSY/nPACK(Line Busy/Peripheral Acknowledge): In ECP mode, it is inactive low to indicate the peripheral is ready for the next transfer in the forward direction. It indicates the data line is ECP command or data in the reverse direction.</td>
</tr>
<tr>
<td>62</td>
<td>nACK</td>
<td>I</td>
<td>Acknowledgment: A acknowledge signal from printer to indicate printer has received data and is ready to accept a new data. The bit nack. SPP_SPR directly reflects this signal. INTR(Interrupt): In EPP mode, it is active high with the positive edge triggered for the interrupt signal. nPACK(Peripheral Acknowledgment): In ECP mode, it is active low to indicate valid data being driven by peripheral.</td>
</tr>
<tr>
<td>60</td>
<td>PE</td>
<td>I</td>
<td>Paper End: A status signal from printer to indicate the printer is out of paper. The bit pe. SPP_SPR directly reflects this signal. (Same definition as SPP in EPP mode) PERRROR/nACKR(PError/nAckReverse): In ECP mode, peripheral uses it to acknowledge a transfer direction change for nRREQ. The direction is forward when asserted, host is then permitted to drive the bus.</td>
</tr>
<tr>
<td>59</td>
<td>SLCT</td>
<td>I</td>
<td>Printer Selected Status: A status signal from printer to indicate the printer has powered on. The bit slct. SPP_SPR directly reflects this signal. (Same definition as SPP in EPP mode) SLCT(Printer Selected Status): In ECP mode, a status signal from printer to indicate it is on-line.</td>
</tr>
<tr>
<td>75</td>
<td>nERROR</td>
<td>I</td>
<td>Printer Port Error: A status signal from printer to indicate an error status at the printer. The bit nerr. SPP_SPR directly reflects this signal. (Same definition as SPP in EPP mode) nFAULT/nPREQ(Fault/Peripheral Request): In ECP mode, peripheral uses it to indicate an error interrupt. It is valid only in forward mode. Occasionally, it can be used as a request for reverse transfer.</td>
</tr>
<tr>
<td>Pin No.</td>
<td>Pin Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>73</td>
<td>nSLCTIN</td>
<td>OD24 O24</td>
<td>Printer Select Input: This output is the complement of the bit slctin. SPP_CPR to select the printer. nASTB(Address Strobe): This output is used to indicate an address port access in EPP mode. nSLCTIN(Printer Select Input): In ECP mode, it is always deasserted.</td>
</tr>
<tr>
<td>74</td>
<td>nINIT</td>
<td>OD24 O24</td>
<td>Printer Initial Output: This output reflects the bit ninit. SPP_CPR to initiate the printer. (Same definition as SPP in EPP mode) nINIT/nRREQ(Initial Output/Reverse Request): In ECP mode, it sets the transfer direction. The transfer direction is reversed when it is asserted.</td>
</tr>
<tr>
<td>76</td>
<td>nAFD</td>
<td>OD24 O24</td>
<td>Printer Autofeed Output: This output is the complement of the bit autofd.SPP_CPR to control the printer for the auto line feed after each line is printed. nDSTB(Data Strobe): This output is used to indicate a data port access in EPP mode. nAFD/HACK(Autofeed Output/Host Acknowledge): In ECP mode, it is asserted to request a byte from the peripheral by the handshaking with nPACK in the reverse direction. In the forward direction, it indicates the data contents is address or data.</td>
</tr>
<tr>
<td>77</td>
<td>nSTB</td>
<td>OD24 O24</td>
<td>Printer Strobe Output: This output is the complement of the bit stb.SPP_CPR to strobe the data into printer. nWRITE(Write): In EPP mode, this output is used to indicate a write operation. nSTB(Strobe Output): In ECP mode, it is used to strobe the address or data into the peripheral on the asserting edge during write operation.</td>
</tr>
</tbody>
</table>

**Infra-red interface**

<table>
<thead>
<tr>
<th>Pin No.</th>
<th>Pin Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>88</td>
<td>IRSI1</td>
<td>I</td>
<td>IR Receive Data In 1: IR Receive data input.</td>
</tr>
<tr>
<td>89</td>
<td>IRSO1</td>
<td>O4</td>
<td>IR Transmit Data Out 1: IR Transmit data output.</td>
</tr>
<tr>
<td>25</td>
<td>IRSI2</td>
<td>I</td>
<td>IR Receive Data In 2: An alternative IR Receive data input.</td>
</tr>
<tr>
<td>26</td>
<td>IRSO2</td>
<td>O24</td>
<td>IR Transmit Data Out 2: An alternative IR Transmit data output.</td>
</tr>
</tbody>
</table>

**Game port interface**

<table>
<thead>
<tr>
<th>Pin No.</th>
<th>Pin Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>58</td>
<td>nGAMECS</td>
<td>O4</td>
<td>Game Port Select: This is a select signal for game port I/O address corresponding to the setup of CR1E when game port is enabled.</td>
</tr>
</tbody>
</table>

8     3rd Jan ‘97
### IDE interface

<table>
<thead>
<tr>
<th>Pin No.</th>
<th>Pin Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>nIDERST</td>
<td>OD48</td>
<td>IDE Reset Output: An inverted RESET output for IDE interface.</td>
</tr>
<tr>
<td>24</td>
<td>nIDEEN</td>
<td>O24</td>
<td>IDE Enable: This signal is active when the IDE port is enabled and the system is accessing an IDE register.</td>
</tr>
<tr>
<td>25</td>
<td>nIDECS0</td>
<td>O24</td>
<td>IDE Chip Select 0: This is a select signal for IDE base address corresponding to the setup of CR21 when IDE port is enabled.</td>
</tr>
<tr>
<td>26</td>
<td>nIDECS1</td>
<td>O24P</td>
<td>IDE Chip Select 1: This is a select signal for IDE alternate base address corresponding to the setup of CR22 when IDE port is enabled.</td>
</tr>
</tbody>
</table>

### Misc

<table>
<thead>
<tr>
<th>Pin No.</th>
<th>Pin Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>58</td>
<td>PWRGD</td>
<td>I</td>
<td>Power Good: This signal indicates the power (Vcc) is valid. When it is inactive, all inputs are disconnected, all outputs are tri-stated, and the contents of registers are kept if the Vcc is valid. It sets system into standby mode.</td>
</tr>
<tr>
<td>20</td>
<td>CLK24</td>
<td>ICLK</td>
<td>Clock 24MHz: A clock input for whole chip.</td>
</tr>
<tr>
<td>94</td>
<td>nPIOCS</td>
<td>OD24U</td>
<td>Programmable I/O Address Decode: This is a select signal for a 1, 8, or 16 byte I/O address corresponding to the setup of CR08 and CR09 when p94s[1:0].CR03 is set to decode mode.</td>
</tr>
<tr>
<td>52</td>
<td>DE6</td>
<td>ID</td>
<td>DE6: HT8669 supports an internal pull down input for ISA mode power on setup. System can use an external pull-up resistor to determine the operation mode. At the rising edge of PWRGD, the DE6 input is latched for the mode selection: 0: Normal mode, On-board with no device being active after hardware reset. 1: ISA mode, Adapter based design with default active value after hardware reset.</td>
</tr>
<tr>
<td>21</td>
<td>DE7</td>
<td>ID</td>
<td>DE7: HT8669 supports an internal pull down input for Ir mode power on setup. System can use an external pull-up resistor to determine the operation mode. At the rising edge of PWRGD, the DE7 input is latched for the mode selection: 0: Normal mode, polarity of IR receive signal is normal. 1: Inverted mode, polarity of IR receive signal is inverted.</td>
</tr>
<tr>
<td>91</td>
<td>SYSOPT</td>
<td>I</td>
<td>Index Base I/O Address Selection: HT8669 supports an input for configuration access setup. System can use an external pull-up/down resistor to determine the address. At the trailing edge of hardware reset, the SYSOPT input is latched for the address selection: 0: Index base I/O address is 3F0h. 1: Index base I/O address is 370h.</td>
</tr>
</tbody>
</table>
### Power

<table>
<thead>
<tr>
<th>Pin No.</th>
<th>Pin Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>15,72</td>
<td>VCC</td>
<td>Power</td>
<td>Vcc Power:</td>
</tr>
<tr>
<td>6,47,67,95</td>
<td>GND</td>
<td>Power</td>
<td>Ground:</td>
</tr>
</tbody>
</table>

Note: Pin type definition:

- **Input:** ISU: Normal, U: With pull-up R, D: With pull-down R

### Register Definition

**FDC register set**

There are status registers, data register, and control registers being built in the FDC subsystem. The address map and the short form of these registers are shown below:

#### Base I/O Address

<table>
<thead>
<tr>
<th>Base I/O Address</th>
<th>Attribute</th>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fdc+00h</td>
<td></td>
<td>Reserved</td>
<td></td>
</tr>
<tr>
<td>fdc+01h</td>
<td></td>
<td>Reserved</td>
<td></td>
</tr>
<tr>
<td>fdc+02h</td>
<td>W/R</td>
<td>FDC_DOR</td>
<td>Digital output register</td>
</tr>
<tr>
<td>fdc+03h</td>
<td></td>
<td>Reserved</td>
<td></td>
</tr>
<tr>
<td>fdc+04h</td>
<td>W R</td>
<td>FDC_DSR</td>
<td>Data rate select register</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FDC_MSR</td>
<td>Main status register</td>
</tr>
<tr>
<td>fdc+05h</td>
<td>W/R</td>
<td>FDC_MDR</td>
<td>Main data register</td>
</tr>
<tr>
<td>fdc+06h</td>
<td></td>
<td>Reserved</td>
<td></td>
</tr>
<tr>
<td>fdc+07h</td>
<td>W R</td>
<td>FDC_CCR</td>
<td>Configuration control register</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FDC_DIR</td>
<td>Digital input register</td>
</tr>
</tbody>
</table>

#### Default Reg bit

<table>
<thead>
<tr>
<th>Default</th>
<th>Reg</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>DOR</td>
<td>0</td>
<td>0</td>
<td>moten[1:0]</td>
<td>dmaen</td>
<td>nreset</td>
<td>dvsel[1:0]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>DSR</td>
<td>sreset</td>
<td>fchpd</td>
<td>0</td>
<td>pcomp[2:0]</td>
<td>drsel[1:0]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MSR</td>
<td>rqm</td>
<td>dio</td>
<td>nondma</td>
<td>cmdbsy</td>
<td>dubsy[3:0]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MDR</td>
<td></td>
<td></td>
<td></td>
<td>hd[7:0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10b</td>
<td>CCR</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>drsel[1:0]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DIR</td>
<td>dskchg</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
• Digital output register (FDC_DOR)

This register is used to control the driver Interface. It can not be affected by a software reset. The definition of the bits are:

<table>
<thead>
<tr>
<th>FDC_DOR: Digital Output Register (fdc+02h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>fdc+02h</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:6</td>
<td>0</td>
<td>Reserved.</td>
</tr>
<tr>
<td>5:4</td>
<td>moten[1:0]</td>
<td>Motor Enable [1:0]: These bits control the related nMOE disk interface. A logic &quot;1&quot; will cause the related output pin to go active.</td>
</tr>
<tr>
<td>3</td>
<td>dmaen</td>
<td>FDC DRQ Enable: Writing &quot;0&quot; can disable nDACK and TC inputs, and hold DRQ and IRQ outputs to Hi-Z state. Writing &quot;1&quot; will enable nDACK, TC, DRQ, and IRQ for DMA function.</td>
</tr>
<tr>
<td>2</td>
<td>nreset</td>
<td>FDC DOR Reset: A logic &quot;0&quot; written to this bit resets FDC. It will remain active until a logic &quot;1&quot; is written to this bit. The minimum reset duration for the software reset is 100ns. It does not affect FDC_DSR, FDC_CCR, and other bits of this register.</td>
</tr>
<tr>
<td>1:0</td>
<td>dvsel[1:0]</td>
<td>Drive Select: These bits are encoded selection bits to select DS0 - DS3. Therefore, only one drive can be accessed at one time.</td>
</tr>
</tbody>
</table>

The access of disk drive can be configured by programming FDC_DOR with the value as below:

<table>
<thead>
<tr>
<th>Drive</th>
<th>DOR_FDC Value</th>
<th>Drive</th>
<th>DOR_FDC Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1Ch</td>
<td>2</td>
<td>4Eh</td>
</tr>
<tr>
<td>1</td>
<td>2Dh</td>
<td>3</td>
<td>8Ph</td>
</tr>
</tbody>
</table>
• Data rate select register (FDC_DSR)

This register is a write-only register. It is used to program the write precompensation, low power mode, software reset, and data rate selection. In PC-AT system, data rate is programmed by using FDC_CCR instead of this register. But, the data rate is set by the recent programming of the FDC_CCR or FDC_DSR. This register is not affected by a software reset. The definition of the bits are:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>sreset</td>
<td>FDC Software Reset: This bit has the same function as nreset.FDC_DOR except the polarity. By the way, this bit is self clearing.</td>
</tr>
<tr>
<td>6</td>
<td>fdclpd</td>
<td>FDC Low Power Mode: Writing “1” can put FDC into Manual Low Power mode. In this mode, the FDC clock and data separator circuit will be turned off. FDC will leave this mode after software reset or access of the FDC_DR or FEC_MSR (and clear this bit).</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>4:2</td>
<td>pcomp[2:0]</td>
<td>FDC Write Precompensation Select: These bits select the value of write precompensation for WDATA. Track 0 is the default starting track for precompensation and can be changed by the configuration command.</td>
</tr>
<tr>
<td>1:0</td>
<td>drsel[1:0]</td>
<td>Data Rate Select: These bits control the data rate of the FDC.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>fdc+04h</td>
<td>W</td>
<td>sreset</td>
<td>fdclpd</td>
<td>0</td>
<td>pcomp2</td>
<td>pcomp1</td>
<td>pcomp0</td>
<td>drsel1</td>
<td>drsel0</td>
</tr>
<tr>
<td>Reset Default</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Main status register (FDC_MSR)

This register is a read-only register. It reports the FDC status for the handshaking with system for FDC access. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>fdc+04h</td>
<td>R</td>
<td>rqm</td>
<td>dio</td>
<td>nondma</td>
<td>cmdbsy</td>
<td>dvbsy3</td>
<td>dvbsy2</td>
<td>dvbsy1</td>
<td>dvbsy0</td>
</tr>
<tr>
<td>Reset Default</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>rqm</td>
<td>FDC Ready for Access: FDC can be accessed if this bit is &quot;1&quot;, otherwise the access is not allowed.</td>
</tr>
<tr>
<td>6</td>
<td>dio</td>
<td>FDC Data Transfer Direction: It indicates the data transfer direction when rqm is set. Reading &quot;1&quot; indicates a read operation. &quot;0&quot; is a write.</td>
</tr>
<tr>
<td>5</td>
<td>nondma</td>
<td>FDC Non-DMA Operation: It reflects the DMA setup in SPECIFY command and will be set &quot;1&quot; during execution phase of a command. It is for polled transfer and helps to distinguish between the data transfer phase and the reading of result bytes.</td>
</tr>
<tr>
<td>4</td>
<td>cmdbsy</td>
<td>FDC Command Progress: This bit indicates the command being processed. It is set after the command bytes being transfer and goes inactive at the end of result phase. If there is no result phase, it will return &quot;0&quot; after the last command byte being transferred.</td>
</tr>
<tr>
<td>3:0</td>
<td>dvbsy[3:0]</td>
<td>Drive x Busy: These bits are set to 1s when a driver is in the seek operation, including implied and overlapped seeks and recalibration.</td>
</tr>
</tbody>
</table>

Main data register (FDC_MDR)

This register is a data I/O register for FDC. All commands, data, and result status are accessed from this register.

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>fdc+05h</td>
<td>WR</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Reset Default</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

HT8669IR
13
3rd Jan '97
• Configuration control register (FDC_CCR)
This register is a write-only register. It is programmed for data rate selection as the function as drsel[1:0], FDC_DOR. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>fdc+07h</td>
<td>W</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>drsel1 drsel0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:2</td>
<td>—</td>
<td>Reserved and could not be accessed.</td>
</tr>
<tr>
<td>1:0</td>
<td>drsel[1:0]</td>
<td>Data Rate Select: These bits control the data rate of the FDC.</td>
</tr>
</tbody>
</table>

• Digital input register (FDC_DIR)
This register is a read-only register. It reports the FDC status for the disk changes. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>fdc+07h</td>
<td>R</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>dskchg</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>dskchg</td>
<td>FDD Disk Changed: This bit reflects the opposite value of the nDCHG input.</td>
</tr>
<tr>
<td>6:0</td>
<td>—</td>
<td>Reserved and tri-state during read access.</td>
</tr>
</tbody>
</table>
Serial port (UART) register set

There are status registers, data buffer registers, and control registers being built in the UART subsystem. The address map of these registers and the short form is shown below:

<table>
<thead>
<tr>
<th>Base I/O Address</th>
<th>dlab</th>
<th>Attribute</th>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+0h</td>
<td>0</td>
<td>W R</td>
<td>CMn_THR</td>
<td>Transmit Buffer Register</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CMn_RBR</td>
<td>Receive Buffer Register</td>
</tr>
<tr>
<td>cmn+0h</td>
<td>1</td>
<td>W/R</td>
<td>CMn_DLL</td>
<td>Divisor LSB (Baud Rate Generator)</td>
</tr>
<tr>
<td>cmn+1h</td>
<td>0</td>
<td>W/R</td>
<td>CMn_IER</td>
<td>Interrupt Enable Register</td>
</tr>
<tr>
<td>cmn+1h</td>
<td>1</td>
<td>W/R</td>
<td>CMn_DLH</td>
<td>Divisor MSB (Baud Rate Generator)</td>
</tr>
<tr>
<td>cmn+2h</td>
<td>x</td>
<td>W R</td>
<td>CMn_FCR</td>
<td>FIFO Control Register</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CMn_IIR</td>
<td>Interrupt Identification Register</td>
</tr>
<tr>
<td>cmn+3h</td>
<td>x</td>
<td>W/R</td>
<td>CMn_LCR</td>
<td>Line Control Register</td>
</tr>
<tr>
<td>cmn+4h</td>
<td>x</td>
<td>W/R</td>
<td>CMn_MCR</td>
<td>Modem Control Register</td>
</tr>
<tr>
<td>cmn+5h</td>
<td>x</td>
<td>W/R</td>
<td>CMn_LSR</td>
<td>Line Status Register</td>
</tr>
<tr>
<td>cmn+6h</td>
<td>x</td>
<td>W/R</td>
<td>CMn_MSR</td>
<td>Modem Status Register</td>
</tr>
<tr>
<td>cmn+7h</td>
<td>x</td>
<td>W/R</td>
<td>CMn_SCR</td>
<td>Scratchpad Register</td>
</tr>
</tbody>
</table>

Note: dlab is the 7th bit of CMn_LCR.

<table>
<thead>
<tr>
<th>Default</th>
<th>Reg</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>THR</td>
<td>Binary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>RBR</td>
<td>Binary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>IER</td>
<td>0 0 0 0 0 emsi elsi ethrei erdai</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>DLL</td>
<td>Binary (LSB)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>DLH</td>
<td>Binary (MSB)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>DLL</td>
<td>sreset fdclpd 0 0 pcomp[2:0] drsel[1:0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>FCR</td>
<td>thr[1:0] 0 0 0 xmtrst revrst fifoen</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>IIR</td>
<td>fifo 0 0 fifoto intid1 intid0 noint</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>LCR</td>
<td>dlab sbc spb eps pen stb wls1 wls0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>MCR</td>
<td>0 0 0 loop out2 out1 rts dtr</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>LSR</td>
<td>erfifo temt thre bi fe pe oe dr</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0</td>
<td>MSR</td>
<td>dcd ri dar cts dded teri ddar dcts</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>SCR</td>
<td>Binary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
• Transmit/Receive Buffer Register (CMn_THR/CMn_RBR)
These registers are used to buffer the transmitting or received data. Bit 0 is transmitted and received first. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+0h</td>
<td>W</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

• Interrupt enable register (CMn_IER)
This register is used to control the attribute of interrupt. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+1h</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:4</td>
<td>0</td>
<td>None.</td>
</tr>
<tr>
<td>3</td>
<td>emsi</td>
<td>Enable MODEM Status Interrupt: Writing “1” can enable the function. This is caused when one of CMn_MSR bits changes state.</td>
</tr>
<tr>
<td>2</td>
<td>elsi</td>
<td>Enable Receiver Line Status Interrupt: Writing “1” can enable the function. The error sources for the interrupt are overrun, parity, framing and break. The CMn_LSR must be read to determine the source.</td>
</tr>
<tr>
<td>1</td>
<td>ethrei</td>
<td>Enable Transmit Buffer Register Empty Interrupt: Writing “1” can enable the function.</td>
</tr>
<tr>
<td>0</td>
<td>erdai</td>
<td>Enable Received Dtat Available Interrupt: Writing “1” can enable the function and timeout interrupts in the FIFO mode.</td>
</tr>
</tbody>
</table>
• Divisor MSB/LSB register (CMn_DLH/CMn_DLL)

These registers program the 16-bit divisor for baud rate generator. The definition are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+1h</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cmn+0h</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Reset Default: 00000000

Below shown is the programming reference table for some specific baud rates, the input clock is 1.8462MHz which is output from 24MHz with a divisor circuit of 13:

<table>
<thead>
<tr>
<th>Desired Baud Rate</th>
<th>Divisor used to generate 16 x Clock</th>
<th>Percent Error Difference between Desired and Actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>2307</td>
<td>0.03</td>
</tr>
<tr>
<td>75</td>
<td>1538</td>
<td>0.03</td>
</tr>
<tr>
<td>110</td>
<td>1049</td>
<td>0.005</td>
</tr>
<tr>
<td>134.5</td>
<td>858</td>
<td>0.01</td>
</tr>
<tr>
<td>150</td>
<td>769</td>
<td>0.03</td>
</tr>
<tr>
<td>300</td>
<td>384</td>
<td>0.16</td>
</tr>
<tr>
<td>600</td>
<td>192</td>
<td>0.16</td>
</tr>
<tr>
<td>1200</td>
<td>96</td>
<td>0.16</td>
</tr>
<tr>
<td>1800</td>
<td>64</td>
<td>0.16</td>
</tr>
<tr>
<td>2000</td>
<td>58</td>
<td>0.5</td>
</tr>
<tr>
<td>2400</td>
<td>48</td>
<td>0.16</td>
</tr>
<tr>
<td>3600</td>
<td>32</td>
<td>0.16</td>
</tr>
<tr>
<td>4800</td>
<td>24</td>
<td>0.16</td>
</tr>
<tr>
<td>7200</td>
<td>16</td>
<td>0.16</td>
</tr>
<tr>
<td>9600</td>
<td>12</td>
<td>0.16</td>
</tr>
<tr>
<td>19200</td>
<td>6</td>
<td>0.16</td>
</tr>
<tr>
<td>38400</td>
<td>3</td>
<td>0.16</td>
</tr>
<tr>
<td>57600</td>
<td>2</td>
<td>1.6</td>
</tr>
<tr>
<td>115200</td>
<td>1</td>
<td>0.16</td>
</tr>
</tbody>
</table>
• FIFO control register (CMn_FCR)

This register is a write-only register and is used to control the FIFO operation. It is not supporting during the DMA operation. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+2h</td>
<td>thr1</td>
<td>thr0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>xmrst</td>
<td>rcvrs</td>
<td>fifoen</td>
</tr>
<tr>
<td>Reset Default</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:6</td>
<td>thr[1:0]</td>
<td>FIFO Threshold Level: These bits are used to set the trigger level for the RCVR FIFO interrupt. Programming with &quot;00&quot; set 1 byte threshold, &quot;01&quot; for 4 bytes, &quot;10&quot; for 8 bytes, and &quot;11&quot; for 14 bytes.</td>
</tr>
<tr>
<td>5:3</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>2</td>
<td>xmrst</td>
<td>Transmit FIFO Reset: Writing &quot;1&quot; can clear all bytes in the XMIT FIFO and can reset its counter logic to 0. By the way, The shift register is not cleared. It is a self-cleared bit.</td>
</tr>
<tr>
<td>1</td>
<td>rcvrs</td>
<td>Receive FIFO Reset: Writing &quot;1&quot; can clear all bytes in the RCVR FIFO and can reset its counter logic to 0. It is a self-cleared bit.</td>
</tr>
<tr>
<td>0</td>
<td>fifoen</td>
<td>FIFO Enable: Writing &quot;1&quot; can enable both XMIT and RCVR FIFOs. When write &quot;0&quot; to this bit, it will disable the FIFOs and clear the contents automatically. It should be set to &quot;1&quot; if other bits in this register are set.</td>
</tr>
</tbody>
</table>
• Interrupt identification register (CMn_IIR)

This register is a read-only register and is used to determine the interrupt source and its priority. During CPU accessing the CMn_IIR, UART will freeze all interrupts and keep current status and pending new interrupts until CPU complete the access. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+2h</td>
<td>R</td>
<td>fifo</td>
<td>fifo</td>
<td>0</td>
<td>0</td>
<td>fifoto</td>
<td>intid1</td>
<td>intid0</td>
<td>noint</td>
</tr>
<tr>
<td>Reset Default</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:6</td>
<td>fifo</td>
<td>FIFO is Enabled: These bits are set when fifoen.CMn_FCR is set.</td>
</tr>
<tr>
<td>5:4</td>
<td>0</td>
<td>None.</td>
</tr>
<tr>
<td>3</td>
<td>fifoto</td>
<td>FIFO Time-Out: When in non-FIFO mode, this bit is always &quot;0&quot;. In FIFO mode (fifoen.CMn_FCR is set), this bit is set along with bit 2 when a timeout interrupt is pending.</td>
</tr>
<tr>
<td>2:1</td>
<td>intid[1:0]</td>
<td>Interrupt Identification: These bits are used to identify the highest priority interrupt indicated below.</td>
</tr>
<tr>
<td>0</td>
<td>noint</td>
<td>No Interrupt Pending: There is no interrupt pending when this bit is &quot;1&quot;. Otherwise, an interrupt is pending if the bit is &quot;0&quot;. It can be used in either a hardwired prioritized or pulled environment.</td>
</tr>
</tbody>
</table>
Below shown is the Interrupt control table for the explanation of fifoto, intid[1:0], noint:

<table>
<thead>
<tr>
<th>IIR[3:0]</th>
<th>Interrupt set and reset functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>fifoto intid1 intid0 noint Priority</td>
<td>Interrupt type</td>
</tr>
<tr>
<td>0 0 0 1</td>
<td>—</td>
</tr>
<tr>
<td>0 1 1 0</td>
<td>Highest Receiver line status</td>
</tr>
<tr>
<td>0 1 0 0</td>
<td>Second Received Data Available</td>
</tr>
<tr>
<td>1 1 0 0</td>
<td>Second Character Timeout Indication</td>
</tr>
<tr>
<td>0 0 1 0</td>
<td>Third CMn_THR Empty CMn_THR empty.</td>
</tr>
<tr>
<td>0 0 0 0</td>
<td>Lowest MODEM Status Clear to Send, or Data Set Ready, or Ring Indicator, or Data Carrier Detect</td>
</tr>
</tbody>
</table>
**Line control register (CMn_LCR)**

This register is used to determine the format of serial line. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+3h</td>
<td>WR</td>
<td>dlab</td>
<td>sbc</td>
<td>spb</td>
<td>eps</td>
<td>pen</td>
<td>stb</td>
<td>wls1</td>
<td>wls0</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>dlab</td>
<td>Divisor Latch Access Bit: It must be set to '1' to access the CMn_DLH/DLL and set to '0' for the access of CMn_RBR/THR/IER.</td>
</tr>
<tr>
<td>6</td>
<td>sbc</td>
<td>Set Break Control Bit: When this bit is set &quot;1&quot;, the SO output is forced to the spacing or logic &quot;0&quot; state and remains there until the bit is reset. This feature enables the UART to alert a terminal in a communications system.</td>
</tr>
<tr>
<td>5</td>
<td>spb</td>
<td>Stick Parity Bit: When this bit is set &quot;1&quot; and pen is enabled, the parity bit is transmitted and then detected by the receiver in the opposite state indicated by eps.</td>
</tr>
<tr>
<td>4</td>
<td>eps</td>
<td>Even Parity Select Bit: When pen is set &quot;1&quot;, it will generate or check the serial data with odd number of logic &quot;1&quot; if this bit is set to &quot;0&quot;, or even parity check rule will be followed for the bit is set to &quot;1&quot;.</td>
</tr>
<tr>
<td>3</td>
<td>pen</td>
<td>Parity Enable Bit: When it is set, a parity bit is generated for transmit or checked during receiving between the last data word bit and the first stop bit of the serial data.</td>
</tr>
<tr>
<td>2</td>
<td>stb</td>
<td>Number of Stop Bit: This bit defines the number of stop bits for transmitting or receiving data. The number of stop bits is depended to wls[1:0] too.</td>
</tr>
<tr>
<td>1:0</td>
<td>wls[1:0]</td>
<td>Word Length Select Bits: These bits defines the number of bits in each transmitted or received serial character. Below shown is the different types of character format.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>stb</th>
<th>wls1</th>
<th>wls0</th>
<th>Word Length</th>
<th>Number of Stop Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5 Bits</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>6 Bits</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>7 Bits</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>8 Bits</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>5 Bits</td>
<td>1.5</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>6 Bits</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>7 Bits</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>8 Bits</td>
<td>2</td>
</tr>
</tbody>
</table>
• MODEM control register (CMn_MCR)

This register is used to control the interfaces of MODEM or data set or the emulated MODEM mode. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>Address</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+4h</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>loop</td>
<td>out2</td>
<td>out1</td>
<td>rts</td>
<td>dtr</td>
</tr>
</tbody>
</table>

Reset Default: 00000000

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-5</td>
<td>0</td>
<td>None.</td>
</tr>
<tr>
<td>4</td>
<td>loop</td>
<td>Loopback Control Bit: When this bit is set &quot;1&quot;, the UART will go into diagnostic test. First, SO is set to marking state (Logic &quot;1&quot;) and SI is disconnected. Then, the output of Transmitter Shift Register is looped back into the Receiver Shift Register input. In this mode, all MODEM inputs (nCTS, nDSR, nRI, and nDCD) are disconnected, and the four MODEM Control output (nDTR, nRTS, OUT1, and OUT2) are internally connected to the four MODEM Control inputs (nCTS, nDSR, nRI, and nDCD) respectively. In addition, the MODEM control output pins are forced Hi-Z and the transmitted data is received immediately.</td>
</tr>
<tr>
<td>3</td>
<td>out2</td>
<td>Output 2: This bit sets to &quot;1&quot; to enable the serial port interrupt. When this bit is &quot;0&quot;, the interrupt is disabled with a Hi-Z state.</td>
</tr>
<tr>
<td>2</td>
<td>out1</td>
<td>Output 1: For read/write access only.</td>
</tr>
<tr>
<td>1</td>
<td>rts</td>
<td>Request To Send: This bit controls the nRTS output. When it is set to &quot;1&quot;, the nRTS output is forced to a logic &quot;0&quot;, and nRTS will be &quot;1&quot; if the bit is &quot;0&quot;.</td>
</tr>
<tr>
<td>0</td>
<td>dtr</td>
<td>Data Terminal Ready: This bit controls the nDTR output. If it is set to &quot;1&quot;, the nDTR output is forced to a logic &quot;0&quot;, and nDTR will be &quot;1&quot; if the bit is &quot;0&quot;.</td>
</tr>
</tbody>
</table>

• Line status register (CMn_LSR):

This register reports the status of serial port interface. Bits 7 through 4 are the error conditions that produce a Receiver Line Status Interrupt when any of the corresponding conditions are detected and the interrupt is enabled. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>Address</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+5h</td>
<td>WR</td>
<td>erffi</td>
<td>temt</td>
<td>thre</td>
<td>bi</td>
<td>fe</td>
<td>pe</td>
<td>oe</td>
<td>dr</td>
</tr>
</tbody>
</table>

Reset Default: 01100000

HT8669IR
22  3rd Jan ‘97
<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>erfifo</td>
<td>Error in RCVR FIFO: When in non-FIFO mode, this bit is permanently set to &quot;0&quot;. In FIFO mode, this bit is set &quot;1&quot;, when there is at least one parity error, frame error or break indication in the FIFO. This bit is cleared when this register is read if there are no subsequent errors in the FIFO.</td>
</tr>
<tr>
<td>6</td>
<td>temt</td>
<td>Transmitter Empty: This bit is set when CMn_THR and CMn_TSR are both empty, and will be reset whenever either CMn_THR or CMn_TSR contains a data character. In the FIFO mode, this bit is set when the CMn_THR and CMn_TSR are both empty. This is a read-only bit.</td>
</tr>
<tr>
<td>5</td>
<td>thre</td>
<td>Transmitter Holding Register Empty: This bit indicates the UART is ready to accept a new character for transmission. It is set when a character is transferred from CMn_THR into the CMn_TSR (Transmitter Shift Register), and will be reset whenever Host loads the CMn_THR. In addition, this bit causes the UART to issue an interrupt when ethrei.CMn_IER is set. In the FIFO mode, this bit is set when the XMIT FIFO is empty. It is cleared when at least 1 byte is written to the XMIT FIFO. This is a read-only bit.</td>
</tr>
<tr>
<td>4</td>
<td>bi</td>
<td>Break Interrupt: This bit is set whenever the received data is held in the Spacing state (&quot;0&quot;) for longer than a full word transmission time (total time of the start bit + data bits + parity bit + stop bits). It is reset when this register is read. In FIFO mode, this error is associated with the particular character in the FIFO it applies to. This error is indicated when the associated character is at the top of the FIFO. When break occurs only one zero character is loaded into the FIFO. Restarting after a break is received requires the serial data to be &quot;1&quot; for at least 1/2 bit time.</td>
</tr>
<tr>
<td>3</td>
<td>fe</td>
<td>Frame Error: This bit is set whenever the received character did not have a valid stop bit, i.e. a spacing level. It is reset when this register is read. In FIFO mode, this error is associated with the particular character in the FIFO it applies to. This error is indicated when the associated character is at the top of the FIFO. The UART will try to re-synchronize after a frame error. To do this, it assumes that the framing error was due to the next start bit, so it samples this &quot;start&quot; bit twice and then takes in the &quot;data&quot; field.</td>
</tr>
<tr>
<td>2</td>
<td>pe</td>
<td>Parity Error: This bit is set when the parity of the receive data is detected error, and is reset when this register is read. In FIFO mode, this error is associated with the particular character in the FIFO it applies to. This error is indicated when the associated character is at the top of the FIFO.</td>
</tr>
<tr>
<td>1</td>
<td>oe</td>
<td>Overrun Error: This bit is set immediately to indicates the overrun condition that the data in the CMn_RBR was not read before the next character was transferred into the register. In FIFO mode, it will occur only when the FIFO is full and the next character has been completely received in the shift register, the character in the shift register is overwritten but not transferred into the FIFO. It is reset when this register is read.</td>
</tr>
<tr>
<td>0</td>
<td>dr</td>
<td>Data Ready: This bit is set to &quot;1&quot; when a complete incoming character has been received and transferred into the CMn_RBR or the FIFO. It is reset to &quot;0&quot; by reading all of the data in the CMn_RBR or the FIFO.</td>
</tr>
</tbody>
</table>
• MODEM status register (CMn_MSR)

This register indicates the current state of the control lines from the MODEM (or peripheral device). In addition, there are 4 bits of this register provide change information. These bits are set to "1" when the corresponding control input from the MODEM changes state. Whenever bit 0, 1, 2, or 3 is set to "1", a MODEM Status Interrupt is generated. They are reset to "0" whenever this register is read. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+6h</td>
<td>WR</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**CMn_MSR: MODEM Status Register (com+6h)**

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>dcd</td>
<td>Data Carrier Detect: This bit reflects the complement of nDCD input. If loop.CMn_MCR is set, this bit is equivalent to out2.CMn_MCR.</td>
</tr>
<tr>
<td>6</td>
<td>ri</td>
<td>Ring Indicator: This bit reflects the complement of nRI input. If loop.CMn_MCR is set, this bit is equivalent to out1.CMn_MCR.</td>
</tr>
<tr>
<td>5</td>
<td>dsr</td>
<td>Data Set Ready: This bit reflects the complement of nDSR input. If loop.CMn_MCR is set, this bit is equivalent to dtr.CMn_MCR.</td>
</tr>
<tr>
<td>4</td>
<td>cts</td>
<td>Clear To Send: This bit reflects the complement of nCTS input. If loop.CMn_MCR is set, this bit is equivalent to rts.CMn_MCR.</td>
</tr>
<tr>
<td>3</td>
<td>ddcd</td>
<td>Delta Data Carrier Detect: This bit is set when the nDCD input to the chip has changed state. It will be cleared when it is read.</td>
</tr>
<tr>
<td>2</td>
<td>teri</td>
<td>Trailing Edge Ring Indicator: This bit is set when the nRI input to the chip has changed from &quot;0&quot; to &quot;1&quot;. It will be cleared when it is read.</td>
</tr>
<tr>
<td>1</td>
<td>ddsr</td>
<td>Delta Data Set Ready: This bit is set when the nDSR input to the chip has changed state since the last time being read. It will be cleared when it is read.</td>
</tr>
<tr>
<td>0</td>
<td>dcts</td>
<td>Delta Clear To Send: This bit is set when the nCTS input to the chip has changed state since the last time being read. It will be cleared when it is read.</td>
</tr>
</tbody>
</table>

• Scratchpad register (CMn_SCR)

This register has no effect on the UART operation but just be a scratchpad register to be used by the programmer to hold data temporarily. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmn+7h</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**CMn_MCR: Scratchpad Register (com+7h)**
Parallel port (SPP/EPP) register set

There are status registers, data buffer registers, and control registers being built in the parallel port subsystem. The registers, SPP_DPR, SPP_SPR, and SPP_CPR, are available in all modes of parallel port. The others, EPP_ADR, and EPP_DP[0:3], are only available in EPP mode. The address map and the short form of these registers are shown below:

<table>
<thead>
<tr>
<th>Base I/O Address</th>
<th>Attribute</th>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+00h</td>
<td>W/R</td>
<td>SPP_DPR</td>
<td>Data Port Register - for all mode</td>
</tr>
<tr>
<td>lpt+01h</td>
<td>W/R</td>
<td>SPP_SPR</td>
<td>Status Port Register - for all mode</td>
</tr>
<tr>
<td>lpt+02h</td>
<td>W/R</td>
<td>SPP_CPR</td>
<td>Control Port Register - for all mode</td>
</tr>
<tr>
<td>lpt+03h</td>
<td>W/R</td>
<td>EPP_ADR</td>
<td>EPP Address Port Register - for EPP mode</td>
</tr>
<tr>
<td>lpt+04h</td>
<td>W/R</td>
<td>EPP_DP0</td>
<td>EPP Data Port 0 Register - for EPP mode</td>
</tr>
<tr>
<td>lpt+05h</td>
<td>W/R</td>
<td>EPP_DP1</td>
<td>EPP Data Port 1 Register - for EPP mode</td>
</tr>
<tr>
<td>lpt+06h</td>
<td>W/R</td>
<td>EPP.dp2</td>
<td>EPP Data Port 2 Register - for EPP mode</td>
</tr>
<tr>
<td>lpt+07h</td>
<td>W/R</td>
<td>EPP_DP3</td>
<td>EPP Data Port 3 Register - for EPP mode</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Default</th>
<th>Reg</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>DPR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>SPR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CPR</td>
<td>0</td>
<td>0</td>
<td>pcd</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>tmout</td>
</tr>
<tr>
<td>00</td>
<td>ADR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>DP0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>DP1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>DP2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>DP3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
In addition, there are more registers being defined for ECP operation. The address map and the short form of these registers are shown below:

<table>
<thead>
<tr>
<th>Base I/O Address</th>
<th>Attribute</th>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+00h</td>
<td>W/R</td>
<td>ECP_DPR</td>
<td>Data Port Register - for ECP 000-001 mode</td>
</tr>
<tr>
<td>lpt+01h</td>
<td>W/R</td>
<td>ECP_AFF</td>
<td>ECP Address FIFO - for ECP 011 mode</td>
</tr>
<tr>
<td>lpt+02h</td>
<td>W/R</td>
<td>ECP_SPR</td>
<td>Status Port Register - for ECP all mode</td>
</tr>
<tr>
<td>lpt+400h</td>
<td>W/R</td>
<td>ECP_PDF</td>
<td>ECP Parallel Port Data FIFO - for ECP 010 mode</td>
</tr>
<tr>
<td>lpt+400h</td>
<td>W/R</td>
<td>ECP_DFF</td>
<td>ECP Data FIFO - for ECP 011 mode</td>
</tr>
<tr>
<td>lpt+400h</td>
<td>R</td>
<td>ECP_TFF</td>
<td>ECP Test FIFO - for ECP 110 mode</td>
</tr>
<tr>
<td>lpt+400h</td>
<td>W/R</td>
<td>ECP_CAR</td>
<td>ECP Configuration Register A - for ECP 111 mode</td>
</tr>
<tr>
<td>lpt+401h</td>
<td>W/R</td>
<td>ECP_CBR</td>
<td>ECP Configuration Register B - for ECP 111 mode</td>
</tr>
<tr>
<td>lpt+402h</td>
<td>W/R</td>
<td>ECP_ECR</td>
<td>ECP Extended Control Register - for ECP all mode</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Default</th>
<th>Reg</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>DPR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>AFF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>SPR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CPR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>PDF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>DFF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>TFF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CAR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CBR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>ECR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

`HT8669IR`
By the way, the pinout assignment of the port connector for each mode is defined below:

<table>
<thead>
<tr>
<th>Host Connector</th>
<th>Chip Pin Number</th>
<th>Standard</th>
<th>EPP</th>
<th>ECP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>77</td>
<td>nStrobe</td>
<td>nWrite</td>
<td>nStrobe</td>
</tr>
<tr>
<td>10</td>
<td>62</td>
<td>nACK</td>
<td>Intr</td>
<td>nACK</td>
</tr>
<tr>
<td>11</td>
<td>61</td>
<td>Busy</td>
<td>nWait</td>
<td>Busy</td>
</tr>
<tr>
<td>12</td>
<td>60</td>
<td>PE</td>
<td>(NU)</td>
<td>PError</td>
</tr>
<tr>
<td>13</td>
<td>59</td>
<td>Select</td>
<td>(NU)</td>
<td>Select</td>
</tr>
<tr>
<td>14</td>
<td>76</td>
<td>nAutofd</td>
<td>nDatastb</td>
<td>nAutofd</td>
</tr>
<tr>
<td>15</td>
<td>75</td>
<td>nError</td>
<td>(NU)</td>
<td>nFault</td>
</tr>
<tr>
<td>16</td>
<td>74</td>
<td>nInit</td>
<td>(NU)</td>
<td>nInit</td>
</tr>
<tr>
<td>17</td>
<td>73</td>
<td>nSelectin</td>
<td>nAddrstb</td>
<td>nSelectin</td>
</tr>
</tbody>
</table>

- **Data port register (SPP_DPR)**
  
  This register is used to latch the contents of output data bus with the rising edge of nIOW during the write operation. The contents of this register are buffered and output onto the PD[7:0] port. During a read operation in SPP mode, PD[7:0] ports are buffered without latch and output to the HOST. The definition of the bits are:

### SPP_DPR: Data Port Register (lpt+0h)

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+0h</td>
<td>WR</td>
<td>Binary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reset Default</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
• Status port register (SPP_SPR)

This register is used to latch the status of printer port with the rising edge of nIOR during the read cycle. Most of the bits are read only, except the bit tmout can be written to clear the status in EPP mode. The definition of the bits are:

| SPP_DPR: Data Port Register (lpt+0h) |
|---|---|---|---|---|---|---|---|---|---|
| Address | Type | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| lpt+1h | WR | nbusy | nack | pe | slct | nerr | 0 | 0 | tmout |
| Reset Default | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>nbusy</td>
<td>BUSY Status: This bit reflects the complement of BUSY input. 0: The printer is busy and can not accept a new character. 1: It is ready to accept next new one.</td>
</tr>
<tr>
<td>6</td>
<td>nack</td>
<td>ACKNOWLEDGE Status: This bit reflects the nACK input. 0: The printer has received a character and can now accept a new one. 1: It is still processing the last character or has not received the data.</td>
</tr>
<tr>
<td>5</td>
<td>pe</td>
<td>Paper End Status: This bit reflects the PE input. 0: Paper present. 1: Paper end.</td>
</tr>
<tr>
<td>4</td>
<td>slct</td>
<td>Printer Selected Status: This bit reflects the SLCT input. 0: The printer is not selected. 1: The printer is on line.</td>
</tr>
<tr>
<td>3</td>
<td>nerr</td>
<td>ERROR Status: This bit reflects the nERROR input. 0: An error has been detected. 1: No error.</td>
</tr>
<tr>
<td>2:1</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>0</td>
<td>tmout</td>
<td>Time Out: This bit is valid in EPP mode only and is set to &quot;1&quot; when a 10us time out has occurred and detected on the EPP bus. It is cleared by a RESET or writing a &quot;1&quot; to this bit. On a write, this bit is cleared by itself with writing a &quot;1&quot;, but is no effect on writing a &quot;0&quot; to this bit.</td>
</tr>
</tbody>
</table>
• Control port register (SPP_CPR)

This register is used to control the printer port. The contents will be initialized by RESET. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+2h</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>pcd</td>
<td>irqe</td>
<td>slctin</td>
<td>ninit</td>
<td>autofd</td>
<td>stb</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:6</td>
<td>Reserved</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>5</td>
<td>pcd</td>
<td>Parallel Control Direction: This bit is valid in extended mode only (ppmode.CR01 = 0). In printer mode, the direction is always out regardless the state of this bit. In bi-directional mode, “0” means the printer port is in output mode (write), and “1” means the printer port is in input mode (read).</td>
</tr>
</tbody>
</table>
| 4   | irqe     | Interrupt Request Enable: This bit enables or disables the interrupt request. When it is high, an interrupt request is generated on the IRQ port by a positive going nACK input.  
0: Disable IRQ.  
1: Enable IRQ. |
| 3   | slctin   | Printer Select Input: This bit is inverted and output onto the nSLCTIN output.  
0: The printer is not selected.  
1: Select the printer. |
| 2   | ninit    | nInitiate Output: This bit is output onto the nINIT output.                  |
| 1   | autofd   | Autofeed: This bit is inverted and output onto the nAFD input.               0: No autofeed.  
1: The printer will generate a line feed after each line is printed. |
| 0   | stb      | Strobe: This bit is inverted and output onto the nSTB output.                |

• EPP address port register (EPP_ADR)

This register is used to buffer and output the contents onto PD[7:0] with no inverting, the leading edge of nIOW causes an EPP ADDRESS WRITE cycle to be performed, the trailing edge of nIOW latches the data for the duration of the EPP write cycle. During a read operation, PD[7:0] ports are read, the leading edge of nIOR causes an EPP ADDRESS READ cycle to be performed and the data output to the HOST, the deassertion of nAddrstb latches the data of PD[7:0] for the duration of the IOR cycle. This register is only available in EPP mode. It is cleared by RESET. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+3h</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
• EPP DATA Port n Register (EPP_DP[0:3])
These registers are used to buffer and output the contents onto PD[7:0] with no inverting, the leading edge of nIOW causes an EPP DATA WRITE cycle to be performed, the trailing edge of nIOW latches the data for the duration of the EPP write cycle. During a read operation, PD[7:0] ports are read, the leading edge of nIOR causes an EPP DATA READ cycle to be performed and the data output to the HOST, the deassertion of nDatastb latches the data of PD[7:0] for the duration of the IOR cycle. These registers are only available in EPP mode. They are cleared by RESET. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>pt+[4:7]h WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Reset Default: 00000000

• Data port register-ECP 000 and 001 mode (ECP_DPR)
This register is used to latch the contents of output data bus with the rising edge of nIOW during the write operation. The contents of this register are buffered and output onto the PD[7:0] port. During a read operation in ECP 000 and 001 mode, PD[7:0] ports are buffered without latch and output to the HOST. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+0h WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Reset Default: 00000000

• ECP address FIFO-ECP 011 mode (ECP_AFF)
This FIFO is used to store the ECP Address/RLE contents. It will be sent automatically. This register is used only in forward direction. This definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+0h WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Reset Default: 00000000

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>rle_a</td>
<td>RLE/Address: This bit defines the contents of ecpa[6:0] is the Run-Length count or ECP port Channel Address. 0: Run-Length Count in ecpa[6:0]. 1: Channel Address in ecpa[6:0].</td>
</tr>
<tr>
<td>6:0</td>
<td>ecpa[6:0]</td>
<td>RLE/Address Value: These bits present the RLE Count value or Channel Address Value for the ECP transmission.</td>
</tr>
</tbody>
</table>
• **ECP status port register (ECP_SPR)**

This register is used to latch the status of printer port with the rising edge of nIOR during the read cycle. Most of the bits are read only. The definition of the bits are:

<table>
<thead>
<tr>
<th>ECP_SPR: ECP Status Port Register (lpt+1h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
</tr>
<tr>
<td>lpt+1h</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>nbusy</td>
<td>nBUSY Status: This bit reflects the complement of BUSY input.</td>
</tr>
<tr>
<td>6</td>
<td>nack</td>
<td>nACKNOWLEDGE Status: This bit reflects the nACK input.</td>
</tr>
<tr>
<td>5</td>
<td>perror</td>
<td>Paper End Status: This bit reflects the PE input.</td>
</tr>
<tr>
<td>4</td>
<td>slct</td>
<td>Printer Selected Status: This bit reflects the SLCT input.</td>
</tr>
<tr>
<td>3</td>
<td>nfault</td>
<td>nERROR Status: This bit reflects the nERROR input.</td>
</tr>
<tr>
<td>2:0</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
</tbody>
</table>

• **Control port register (ECP_CPR)**

This register is used to control the printer port. The contents will be initialized by RESET. The definition of the bits are:

<table>
<thead>
<tr>
<th>ECP_CPR: ECP Status Port Register (lpt+2h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
</tr>
<tr>
<td>lpt+2h</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:6</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>5</td>
<td>pcd</td>
<td>Parallel Control Direction: This bit is no effect in ECP 000 or 010 mode, and the direction is always out. In all other modes, it is valid and a logic &quot;0&quot; means the printer port is in output mode (write), and &quot;1&quot; means the printer port is in input mode (read).</td>
</tr>
<tr>
<td>4</td>
<td>irqe</td>
<td>Interrupt Request Enable: This bit enables or disables the interrupt request. When it is high, an interrupt request is generated on the IRQ port by a positive going nACK input. 0: Disable IRQ. 1: Enable IRQ.</td>
</tr>
<tr>
<td>3</td>
<td>slctin</td>
<td>Printer Select Input: This bit is inverted and output onto the nSLCTIN output. 0: The printer is not selected. 1: Select the printer.</td>
</tr>
<tr>
<td>2</td>
<td>ninit</td>
<td>nInitiate Output: This bit is output onto the nINIT output.</td>
</tr>
<tr>
<td>1</td>
<td>autofd</td>
<td>Autofeed: This bit is inverted and output onto the nAFD input. 0: No autofeed. 1: The printer will generate a line feed after each line is printed.</td>
</tr>
<tr>
<td>0</td>
<td>stb</td>
<td>Strobe: This bit is inverted and output onto the nSTB output.</td>
</tr>
</tbody>
</table>
• ECP parallel port data FIFO-ECP 010 mode (ECP_PDF)
This FIFO is used to store the bytes written or DMAed from the system in ECP 010 mode. It is defined only in the forward direction. The FIFO is shared with other FIFO accessing. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+400h</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Reset Default: 0 0 0 0 0 0 0 0 0

• ECP data FIFO-ECP 011 mode (ECP_DFF):
This FIFO is used to store the bytes written or DMAed from the system or ECP port in ECP 011 mode. It is defined for bi-direction. The FIFO is shared with other FIFO accessing. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+400h</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Reset Default: 0 0 0 0 0 0 0 0 0

• ECP test FIFO-ECP 110 mode (ECP_TFF)
This FIFO is used to store the bytes accessed or DMAed from the system in ECP 110 mode. It is defined for any direction. Data may not be transferred to printer port. The FIFO is shared with other FIFO accessing. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+400h</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Reset Default: 0 0 0 0 0 0 0 0 0

• ECP configuration register A-ECP 111 mode (ECP_CAR)
This register is read only. It is used to indicate this device is an 8-bit implementation. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+400h</td>
<td>R</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Reset Default: 0 0 0 1 0 0 0 0 0
• ECP configuration register B-ECP 111 mode (ECP_CBR)

This register is used to indicate the compression mode and the IRQ status while system read it back. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+401h</td>
<td>WR</td>
<td>cpress</td>
<td>intvle</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>cpress</td>
<td>Compress: This bit is read only and response a low level to indicate this device can not support hardware RLE compression. By the way, hardware RLE decompression is supported.</td>
</tr>
<tr>
<td>6</td>
<td>intvle</td>
<td>Interrupt Value: This bit reflects the value on the ISA IRQ line to determine the possible conflicts.</td>
</tr>
<tr>
<td>5:0</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
</tbody>
</table>

• ECP extended control register (ECP_ECR)

This register is used to control the ECP extended functions. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpt+402h</td>
<td>WR</td>
<td>ecpm[2:0]</td>
<td>neiren</td>
<td>dmaen</td>
<td>sintr</td>
<td>full</td>
<td>empty</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:5</td>
<td>ecpm[2:0]</td>
<td>ECP Mode: These bits are used to select the operation mode in ECP mode. The definition is shown below.</td>
</tr>
<tr>
<td>4</td>
<td>neiren</td>
<td>Error Interrupt Enable: This bit control the function of error interrupt. An interrupt will be generated on the falling edge of the nFAULT signal when it is set to &quot;0&quot;. The interrupt will be disabled while it is set to &quot;1&quot;. This device will still generate a interrupt when nFAULT is asserted and this bit is written from &quot;1&quot; to &quot;0&quot;.</td>
</tr>
<tr>
<td>3</td>
<td>dmaen</td>
<td>DMA Enable: This bit control the function of DMA. It is set to &quot;1&quot; to enable the DMA function or is set to &quot;0&quot; to disable the DMA function.</td>
</tr>
<tr>
<td>Bit</td>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>2</td>
<td>sintr</td>
<td>Service Interrupt: This bit is set to &quot;1&quot; to disable the DMA and all of other interrupts, or to indicate the interrupt being serviced. When it is reset to &quot;0&quot;, one of three conditions will set it to &quot;1&quot;. One is terminal count being reached during DMA (dmaen is set). Second condition is Write Interrupt Threshold being reached or more bytes free in the FIFO when dmaen is &quot;0&quot; and the direction is &quot;0&quot;. Last one is Read Interrupt Threshold being reached or more valid bytes to be read from the FIFO when dmaen is &quot;0&quot; and the direction is &quot;1&quot;. It must be reset to service the next interrupt.</td>
</tr>
<tr>
<td>1</td>
<td>full</td>
<td>FIFO Full: This bit is read only. It indicates FIFO can not accept another data or the FIFO is full when it is &quot;1&quot;. It is &quot;0&quot; to indicate the FIFO can accept one or more bytes of data.</td>
</tr>
<tr>
<td>0</td>
<td>empty</td>
<td>FIFO Empty: This bit is read only. It indicates FIFO is empty when it is &quot;1&quot;. It is &quot;0&quot; to indicate the FIFO contains one or more bytes of data.</td>
</tr>
</tbody>
</table>

The following table shows the ECP mode being set by ecpm[2:0].ECP_ECR:

<table>
<thead>
<tr>
<th>ecpm[2:0]</th>
<th>Mode Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>SPP mode: In this mode, it works like the standard Parallel Port. FIFO is disabled.</td>
</tr>
<tr>
<td>001</td>
<td>PS/2 mode: In this mode, it works like the standard PS/2 Parallel Port.</td>
</tr>
<tr>
<td>010</td>
<td>Parallel Port Data FIFO mode: In this mode, it works like the standard Parallel Port, but the FIFO is enabled. It can be enabled only when the direction is 0.</td>
</tr>
<tr>
<td>011</td>
<td>ECP mode: In this mode, it works in the ECP mode.</td>
</tr>
<tr>
<td>100</td>
<td>EPP mode: In this mode, it works like the EPP mode, if the option is set in ppm[1:0]. CR04.</td>
</tr>
<tr>
<td>101</td>
<td>Reserved.</td>
</tr>
<tr>
<td>110</td>
<td>Test mode: It is used to test FIFO in this mode.</td>
</tr>
<tr>
<td>111</td>
<td>Configuration mode: In this mode, ECP_CAR and ECP_CBR can be accessed via the offset address been set in adrp[9:2]. CR23.</td>
</tr>
</tbody>
</table>
Configuration register set

There are configuration registers for chip initial setup. These registers can be accessed via indexed mapping by using index base I/O address 3F0/3F1h or 370/371h after the configuration cycle being enabled. The configuration index address is selected via the external pull-up or pull-down resistor on the pin SYSOPT. In order to enter the configuration cycle system has to write the index I/O port 3F0h or 370h with the data 55h twice consecutively. When the configuration cycle is enabled, system can access these configuration registers via index I/O port 3F0h/3F1h or 370h/371h. When the configuration is completed, system should write the index I/O port with the data AAh to exit the configuration cycle. The address map of these registers is shown below:

<table>
<thead>
<tr>
<th>Default</th>
<th>Reg</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>28/2A</td>
<td>CR00</td>
<td>valid</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>fdcpwr</td>
<td>0</td>
<td>ideen[1:0]</td>
<td></td>
</tr>
<tr>
<td>9C</td>
<td>CR01</td>
<td>crlock</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>ppmode</td>
<td>ltpwr</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>88</td>
<td>CR02</td>
<td>cm2pwr</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>cm1pwr</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>70</td>
<td>CR03</td>
<td>p94s1</td>
<td>ident</td>
<td>1</td>
<td>p18s</td>
<td>0</td>
<td>p94s0</td>
<td>efdc</td>
<td>p58s</td>
</tr>
<tr>
<td>00</td>
<td>CR04</td>
<td>altir</td>
<td>epprev</td>
<td>0</td>
<td>0</td>
<td>00ppfdc[1:0]</td>
<td>pppm[1:0]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>CR05</td>
<td>abswp</td>
<td>dens[1:0]</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FF</td>
<td>CR06</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>fd1id[1:0]</td>
<td>fd0id[1:0]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CR07</td>
<td>fdcapd</td>
<td>cm1apd</td>
<td>cm2apd</td>
<td>lptapd</td>
<td>0</td>
<td>0</td>
<td>fboot[1:0]</td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CR08</td>
<td>adra[7:4]</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CR09</td>
<td>sizea[1:0]</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>adral0:8]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CR10A</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CR10F</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CR11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CR12:1D</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>80/82</td>
<td>CR1E</td>
<td>adrg[9:4]</td>
<td>sizeg[1:0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CR1F</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>fd1dt[0:1]</td>
<td>fd0dt[0:1]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3C/FC</td>
<td>CR20</td>
<td>adrif9:4</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3C/7C</td>
<td>CR21</td>
<td>adrif9:4</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3D/FD</td>
<td>CR22</td>
<td>adrif9:4</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00/DE</td>
<td>CR23</td>
<td>adrpf9:2</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00/FE</td>
<td>CR24</td>
<td>adruf9:3</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00/BE</td>
<td>CR25</td>
<td>adru9:3</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00/20</td>
<td>CR26</td>
<td>dma[3:0]</td>
<td>dmap[3:0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00/65</td>
<td>CR27</td>
<td>irqf[3:0]</td>
<td>irqp[3:0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00/43</td>
<td>CR28</td>
<td>irqf[3:0]</td>
<td>irqv[3:0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00/01</td>
<td>CR29</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>irqi[3:0]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>—</td>
<td>CR[2A:3F]</td>
<td>Reserved</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00</td>
<td>CR40</td>
<td>regdef</td>
<td>isadef</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
Configuration register 00 (CR00)

It is used to control the parallel port power and mode. The definition of the bits are:

<table>
<thead>
<tr>
<th>Bit Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>valid</td>
<td>Configuration Valid: Control software need to set this bit in the proper time to indicate a valid configuration cycle has occurred. It will be cleared after power up. This bit is used for a status report and do not affect any circuit.</td>
</tr>
<tr>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>1</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>2</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>3</td>
<td>FDC Power Down: This bit is set to &quot;1&quot; to enable the FDC operation. Setting this bit low will force the FDC into low power mode.</td>
</tr>
<tr>
<td>ideen[1:0]</td>
<td>IDE Enable: These bits control alternate function of the IDE interface as below: 00: IDE, IRSI2, IRSO2, IRQ_H disabled (Default) 01: Reserved (IDE, IRSI2, IRSO2, IRQ_H disabled). 10: IDE enabled (set as default via DRQ_B being pulled up externally). 11: IRSI2, IRSO2, IRQ_H Enabled.</td>
</tr>
</tbody>
</table>
* Configuration register 01 (CR01)

It is used to control the parallel port power and mode. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(01h)</td>
<td>WR</td>
<td>crlock</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>ppmode</td>
<td>lptpwr</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>crlock</td>
<td>Lock CRx: This bit is set &quot;1&quot; after the power on to enable the accessibility of configuration registers of CR00-CR17. When it is set to &quot;0&quot;, those registers can not be accessed and this bit can only be set back to &quot;1&quot; via a hardware reset or power-up reset.</td>
</tr>
<tr>
<td>6:5</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>3</td>
<td>ppmode</td>
<td>Parallel Port Mode: This bit is &quot;1&quot; as the default in the reset and sets the parallel port for Standard Printer Mode. The parallel port is set to Extended Parallel Port Modes when this bit is set to &quot;0&quot;.</td>
</tr>
<tr>
<td>2</td>
<td>lptpwr</td>
<td>Parallel Port Power Down: This bit is &quot;1&quot; as the default in the reset and sets the parallel port in normal operation mode. A &quot;0&quot; sets the port being in low power mode.</td>
</tr>
<tr>
<td>1:0</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
</tbody>
</table>

* Configuration register 02 (CR02)

It is used to control the serial port power mode. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(02h)</td>
<td>WR</td>
<td>cm2pwr</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>cm1pwr</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>cm2pwr</td>
<td>UART2 Power Down: This bit is &quot;1&quot; as the default in the reset and sets the Secondary UART port in normal operation mode. A &quot;0&quot; sets the port being in low power mode.</td>
</tr>
<tr>
<td>6:4</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>3</td>
<td>cm1pwr</td>
<td>UART1 Power Down: This bit is &quot;1&quot; as the default in the reset and sets the Primary UART port in normal operation mode. A &quot;0&quot; sets the port being in low power mode.</td>
</tr>
<tr>
<td>2:0</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
</tbody>
</table>
- Configuration register 03 (CR03)

It is used to control the FDC media setup and game port selection. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(03h)</td>
<td>WR</td>
<td>p94s1</td>
<td>ident</td>
<td>mfm</td>
<td>p18s</td>
<td>0</td>
<td>p94s0</td>
<td>efdc</td>
<td>p58s</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7,2</td>
<td>p94s[1:0]</td>
<td>nPIOCS/IRQ_B Selection: These bits select the function of pin 94 as below: 0x: Tri-state. 10: nPIOCS. 11: IRQ_B.</td>
</tr>
<tr>
<td>6</td>
<td>ident</td>
<td>IDENT Selection: This bit is used in conjunction with mfm to select the interface mode of FDC as below: x0: reserved. 01: PS/2 mode(3.5&quot; FDD). 11: AT mode(5.25&quot; FDD).</td>
</tr>
<tr>
<td>5</td>
<td>mfm</td>
<td>Fixed to &quot;1&quot; as AT mode, read only.</td>
</tr>
<tr>
<td>4</td>
<td>p18s</td>
<td>DDEN1/nIDERST Selection: This bit selects the function of pin 18: 0: DDEN1. 1: nIDERST.</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>1</td>
<td>efdc</td>
<td>Enhanced Floppy Mode: This bit is &quot;0&quot; as the default in the reset and sets the FDC to operate as normal mode. An &quot;1&quot; sets the FDC being in enhance mode and will set FDC_TDR to report additional information.</td>
</tr>
<tr>
<td>0</td>
<td>p58s</td>
<td>PWRGD/nGAMECS Selection: This bit selects the function of pin 58 as below: 0: PWRGD. 1: nGAMECS.</td>
</tr>
</tbody>
</table>
Configuration Register 04 (CR04)

It is used to control the parallel port setup and IR port selection. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>Address</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(04h) WR</td>
<td>altir</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Bit Name** | **Description**
--- | ---
7 altir | IR ALT I/O: This bit selects the IR output to alter in/out pins as below:
0: use IRSI1/IRSO1.
1: use IRSI2/IRSO2.
When it sets "1", it should be combined with ideen[1:0].CR00 being set to "11".

6 epprev | EPP Revision Control: This bit selects the revision of EPP as below:
0: EPP 1.9.
1: EPP 1.7.

5 midi2 | UART2 MIDI Clock Control: This bit selects the divisor of UART2 input clock as below:
0: Clock derived by a divisor of 13, normal operation.
1: Clock derived by a divisor of 12, MIDI operation.

4 midi1 | UART1 MIDI Clock Control: This bit selects the divisor of UART1 input clock as below:
0: Clock derived by a divisor of 13, normal operation.
1: Clock derived by a divisor of 12, MIDI operation.

3:2 ppfd[1:0] | Parallel Port FDC Control: These bits control the swap logic of the FDD interface to Printer interface as below:
00: Normal, printer I/F.
01: PPFD1 - one FDD. 10: PPFD2 - two FDD.
11: Reserved.

1:0 ppee[1:0] | Parallel Port Extended Modes Selection: These bits select the extended mode function as below when ppmode.CR01 is set to extended mode.
00: Standard and Bi-directional Modes(SPP, IBM PS/2).
01: EPP mode and SPP.
10: ECP mode (SPP can be selected via ECP_ECR as mode 000)
11: ECP&EPP mode (SPP/EPP can be selected via ECP_ECR as mode 000/100)
• Configuration register 05 (CR05)

It is used to control the FDC swap and density selection. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(05h)</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>abswp</td>
<td>dens1</td>
<td>dens0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:6</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>5</td>
<td>abswp</td>
<td>FDD A/B Swap: This bit controls the swap function of drives and nMOE[1:0] of FDC. A high will enable the function.</td>
</tr>
<tr>
<td>4:3</td>
<td>dens[1:0]</td>
<td>DENSEL Output Control: These bits control the output of DENSEL as below: 00: Normal, followed with data rate control and drive type control. 01: Reserved. 10: Fixed high. 11: Fixed low.</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>1:0</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
</tbody>
</table>

• Configuration register 06 (CR06)

It is used to hold the floppy disk drive types. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(06h)</td>
<td>WR</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>fd1id1</td>
<td>fd1id0</td>
<td>fd0id1</td>
<td>fd0id0</td>
</tr>
<tr>
<td>Reset Default</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:4</td>
<td>1</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>3:2</td>
<td>fd1id[1:0]</td>
<td>FDD1 type: These bits hold the FDD1 floppy disk drive type.</td>
</tr>
<tr>
<td>1:0</td>
<td>fd0id[1:0]</td>
<td>FDD0 type: These bits hold the FDD0 floppy disk drive type.</td>
</tr>
</tbody>
</table>
- Configuration register 07 (CR07)

It is used to control the auto powerdown feature for each subsystem. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(07h) WR</td>
<td>fd capd</td>
<td>cm1 apd</td>
<td>cm2 apd</td>
<td>lpt apd</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>fd capd</td>
<td>FDC Auto Powerdown Mode: This bit enables the FDC auto powerdown feature when sets to high, and disables the feature when low. It is reset by power on reset and hardware reset both.</td>
</tr>
<tr>
<td>6</td>
<td>cm1 capd</td>
<td>COM1 Auto Powerdown Mode: This bit enables the COM1 auto powerdown feature when sets to high, and disables the feature when low. It is reset by power on reset and hardware reset both.</td>
</tr>
<tr>
<td>5</td>
<td>cm2 capd</td>
<td>COM2 Auto Powerdown Mode: This bit enables the COM2 auto powerdown feature when sets to high, and disables the feature when low. It is reset by power on reset and hardware reset both.</td>
</tr>
<tr>
<td>4</td>
<td>lpt capd</td>
<td>Parallel Port Auto Powerdown Mode: This bit enables the parallel port auto powerdown feature when sets to high, and disables the feature when low. It is reset by power on reset and hardware reset both.</td>
</tr>
<tr>
<td>3:2</td>
<td>reserved</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>1:0</td>
<td>fboot[1:0]</td>
<td>Boot Floppy: These bits defines the boot floppy as below.</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>Drive A.</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>Drive B.</td>
</tr>
</tbody>
</table>

- Configuration register 08 (CR08)

It is the address input of Programmable I/O decoder. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(08h) WR</td>
<td>adra7</td>
<td>adra6</td>
<td>adra5</td>
<td>adra4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:4</td>
<td>adra[7:4]</td>
<td>Programmable I/O Address Bit [7:4]: These bits are the lower 4 bits of addresses for the Programmable I/O decoder.</td>
</tr>
<tr>
<td>3:0</td>
<td>reserved</td>
<td>Reserved and read only.</td>
</tr>
</tbody>
</table>
• Configuration register 09 (CR09)

It is the address input and size control of Programmable I/O decoder. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(09h)</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Reset Default

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| 7:6 | sizea[1:0]   | Programmable I/O Configuration Control: These bits control the function and the decode size of Programmable I/O decoder as below:
0: Programmable I/O disabled.
01: 1 Byte decode, A[3:0] = 0000b.
| 5:3 | 0            | Reserved and read only.                                                      |
| 2:0 | adra[10:8]   | Programmable I/O Address Bit [10:8]: These bits are the upper 3 bits of addresses for the Programmable I/O decoder. |

• Configuration register 0A (CR0A)

It is the address input and size control of Programmable I/O decoder. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(0Ah)</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Reset Default

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:4</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>3:0</td>
<td>pthr[3:0]</td>
<td>ECP FIFO Threshold: These bits define the FIFO threshold value for the ECP mode parallel port.</td>
</tr>
</tbody>
</table>
• Configuration register 0B (CR0B)

It is used to hold the floppy disk data rate. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(0Bh)</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>fd1drt1</td>
<td>fd1drt0</td>
<td>fd0drt1</td>
<td>fd0drt0</td>
</tr>
<tr>
<td>Reset Default</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:4</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>3:2</td>
<td>fd1drt[1:0]</td>
<td>FDD1 Data Rate: These bits hold the FDD1 data rate.</td>
</tr>
<tr>
<td>1:0</td>
<td>fd0drt[1:0]</td>
<td>FDD0 Data Rate: These bits hold the FDD0 data rate.</td>
</tr>
</tbody>
</table>

• Configuration register 0C (CR0C)

It is used to define the Ir interface. In addition, this register is reset by the power on reset or a hardware reset. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address</th>
<th>Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(0Ch)</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>irmod2</td>
<td>irmod1</td>
<td>irmod0</td>
<td>irdpx</td>
<td>irtxp</td>
<td>irxpx</td>
</tr>
<tr>
<td>Reset Default</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:6</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>5:3</td>
<td>irmod [2:0]</td>
<td>Ir Mode Selection: These bits define Ir mode for UART2 as below:000: Standard (No Ir). 001: IrDA (HP-SIR). 010: ASK-IR@500K. 011: Reserved. 1xx: Reserved.</td>
</tr>
<tr>
<td>2</td>
<td>irdpx</td>
<td>Ir Duplex Selection: This bit selects the duplex mode of Ir interface (UART2). 0: Full Duplex. 1: Half Duplex.</td>
</tr>
<tr>
<td>1</td>
<td>irtxp</td>
<td>Ir Transmit Polarity: This bit selects the polarity of Ir transmitting (UART2). 0: IRSO output non-inverted. (default) 1: IRSO output inverted.</td>
</tr>
<tr>
<td>0</td>
<td>irxpx</td>
<td>Ir Receive Polarity: This bit selects the polarity of Ir Receiving (UART2). 0: IRSI output non-inverted. (default) 1: IRSI output inverted. (default by DRQ_A pull up)</td>
</tr>
</tbody>
</table>
**Configuration register 0D (CR0D)**

It is read only for device ID identification. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(0Dh) R</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Configuration register 0E (CR0E)**

It is read only for revision ID identification. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(0Eh) R</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Configuration register F-1D (CR[0F:1D])**

These registers are reserved and read as 00h.

**Configuration register 1E (CR1E)**

It is used to define the GAME port address selection. The address can be set to 48 locations with 16-byte boundaries from 100h-3F0h. By the way, nCS = 0 and A10 = 0 are required to qualify the nGAMECS output, and the bit p58s.CR03 can override this register if it selects another output function, PWRGD. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
<th>sizeg1</th>
<th>sizeg0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(1Eh) WR</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0/1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:2</td>
<td>adrg[9:4]</td>
<td>nGAMECS Address Bit [9:4]: These bits are the upper 6 bits of addresses for the nGAMECS decoder.</td>
</tr>
</tbody>
</table>
• Configuration register 1F (CR1F)

It is used to select the drive type. The definition of the bits are:

<table>
<thead>
<tr>
<th>CR1F: Configuration Register 1F (CSR(1Fh))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address Type</td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>csr(1Fh)</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:4</td>
<td>0 Reserved and read only.</td>
<td></td>
</tr>
<tr>
<td>3:2</td>
<td>d1dt[0:1]</td>
<td>FDD1 Drive Type: These bits hold the FDD1 drive type.</td>
</tr>
<tr>
<td>1:0</td>
<td>fd0dt[0:1]</td>
<td>FDD0 Drive Type: These bits hold the FDD0 drive type.</td>
</tr>
</tbody>
</table>

• Configuration Register 20 (CR20)

It is used to define the FDC address selection. The address can be set to 48 locations with 16-byte boundaries from 100h-3F0h. By the way, nCS = 0 and A10 = 0 are required to qualify for the registers accesses of FDC. A[3:0] are decoded as 0xxxb. The definition of the bits are:

<table>
<thead>
<tr>
<th>CR20: Configuration Register 20 (CSR(20h))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address Type</td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>csr(20h)</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:2</td>
<td>adrf[9:4]</td>
<td>FDC Address Bit [9:4]: These bits are the upper 6 bits of addresses for the FDC decoder. It can be set to ISA mode default by pull-up DRQ_B.</td>
</tr>
<tr>
<td>1:0</td>
<td>0 Reserved and read only.</td>
<td></td>
</tr>
</tbody>
</table>
• Configuration register 21 (CR21)

It is used to define the IDE interface base address for the control registers (0-7). The address can be set to 48 locations with 16-byte boundaries from 100h-3F0h. System can set adri[9:8] as "00" to disable the decoder. By the way, nCS = 0 and A10 = 0 are required to qualify for the registers accesses of IDE interface. A[3:0] are decoded as 0xxb. The definition of the bits are:

<table>
<thead>
<tr>
<th>CR21: Configuration Register 21 (CSR(21h))</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong></td>
</tr>
<tr>
<td>----------------</td>
</tr>
<tr>
<td>csr(21h)</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:2</td>
<td>adri[9:4]</td>
<td>IDE Base Address Bit [9:4]: These bits are the upper 6 bits of addresses for the IDE decoder. It can be set to ISA mode default by pull-up DRQ_B.</td>
</tr>
<tr>
<td>1:0</td>
<td></td>
<td>Reserved and read only.</td>
</tr>
</tbody>
</table>

• Configuration register 22 (CR22)

It is used to define the alternate IDE interface base address for the control registers (0-7). The address can be set to 48 locations with 16-byte boundaries from 106h-3F6h. System can set adre[9:8] as "00" to disable the decoder. By the way, nCS = 0 and A10 = 0 are required to qualify for the registers accesses of IDE interface. A[3:0] must be decoded as 0110b. The definition of the bits are:

<table>
<thead>
<tr>
<th>CR22: Configuration Register 22 (CSR(22h))</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong></td>
</tr>
<tr>
<td>----------------</td>
</tr>
<tr>
<td>csr(22h)</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:2</td>
<td>adri[9:4]</td>
<td>IDE Alternate Base Address Bit [9:4]: These bits are the upper 6 bits of addresses for the IDE decoder. It can be set to ISA mode default by pull-up DRQ_B.</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Reserved and read only.</td>
</tr>
</tbody>
</table>
• Configuration register 23 (CR23)
It is used to define the parallel port address. When EPP mode is not enabled, the address can be set to 192 locations with 4-byte boundaries from 100h-3FCh. When EPP mode is enabled, it can be set to 96 locations with 8-byte boundaries from 100h-3F8h. System can set adrp[9:8] as "00" to disable the decoder. By the way, nCS = 0 and A10 = 0 are required to qualify for the registers accesses of parallel port when in compatible, bi-directional, or EPP modes (A10 is active when in ECP mode). It can be set to ISA mode default by pull-up DRQ_B. The definition of the bits are:

<table>
<thead>
<tr>
<th>CR23: Configuration Register 23 (CSR(23h))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>csr(23h)</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

• Configuration register 24 (CR24)
It is used to define the UART1 base address. The address can be set to 96 locations with 8-byte boundaries from 100h-3F8h. System can set adru[9:8] as "00" to disable the decoder. By the way, nCS = 0 and A10 = 0 are required to qualify for the registers accesses of UART1. The definition of the bits are:

<table>
<thead>
<tr>
<th>CR24: Configuration Register 24 (CSR(24h))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>csr(24h)</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:1</td>
<td>adru[9:3]</td>
<td>UART1 Base Address Bit [9:3]: These bits are the upper 7 bits of addresses for the UART1 decoder. It can be set to ISA mode default by pull-up DRQ_B.</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
</tbody>
</table>

• Configuration register 25 (CR25)
It is used to define the UART2 base address. The address can be set to 96 locations with 8-byte boundaries from 100h-3F8h. System can set adrv[9:8] as "00" to disable the decoder. By the way, nCS = 0 and A10 = 0 are required to qualify for the registers accesses of UART2. The definition of the bits are:

<table>
<thead>
<tr>
<th>CR25: Configuration Register 25 (CSR(25h))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>csr(25h)</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:1</td>
<td>adrv[9:3]</td>
<td>UART2 Base Address Bit [9:3]: These bits are the upper 7 bits of addresses for the UART2 decoder. It can be set to ISA mode default by pull-up DRQ_B.</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>Reserved and read only.</td>
</tr>
</tbody>
</table>
• Configuration register 26 (CR26)

It is used to define the DMA channels for FDC and parallel port. Any unselected DMA acknowledge output is in tri-state. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(26h) WR</td>
<td>dmaf3</td>
<td>dmaf2</td>
<td>dmaf1</td>
<td>dmaf0</td>
<td>dmap3</td>
<td>dmap2</td>
<td>dmap1</td>
<td>dmap0</td>
</tr>
<tr>
<td>Reset Default</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:4 dmaf[3:0]</td>
<td>FDC DMA Channel: These bits defines the selection of DMA channel for FDC as the table shown below. It can be set to ISA mode default by pull-up DRQ_B.</td>
</tr>
<tr>
<td>3:0 dmap[3:0]</td>
<td>Parallel Port DMA Channel: These bits defines the selection of DMA channel for parallel port as the table shown below.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>dmax[3:0]</th>
<th>DMA Selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>None</td>
</tr>
<tr>
<td>0001</td>
<td>DMA_A</td>
</tr>
<tr>
<td>0010</td>
<td>DMA_B</td>
</tr>
<tr>
<td>0011</td>
<td>DMA_C</td>
</tr>
<tr>
<td>Others</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

• Configuration register 2[7:9] (CR2[7:9])

These registers are used to define the IRQ channels for FDC, parallel port, UART1, UART2, and IRQIN. Any unselected IRQ output is in tri-state. The definition of the bits are:

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(27h) WR</td>
<td>irqf3</td>
<td>irqf2</td>
<td>irqf1</td>
<td>irqf0</td>
<td>irqp3</td>
<td>irqp2</td>
<td>irqp1</td>
<td>irqp0</td>
</tr>
<tr>
<td>Reset Default</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7:4 irqf[3:0]</td>
<td>FDC IRQ Channel: These bits defines the selection of IRQ channel for FDC as the table shown below. It can be set to ISA mode default by pull-up DRQ_B.</td>
</tr>
<tr>
<td>3:0 irqp[3:0]</td>
<td>Parallel Port IRQ Channel: These bits defines the selection of IRQ channel for parallel port as the table shown below. It can be set to ISA mode default by pull-up DRQ_B.</td>
</tr>
</tbody>
</table>
### CR28: Configuration Register 28 (CSR(28h))

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(28h)</td>
<td>WR</td>
<td>irqu3</td>
<td>irqu2</td>
<td>irqu1</td>
<td>irqu0</td>
<td>irqv3</td>
<td>irqv2</td>
<td>irqv0</td>
</tr>
<tr>
<td>Reset Default</td>
<td>0</td>
<td>0/1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0/1</td>
<td>0/1</td>
</tr>
</tbody>
</table>

**Bit Name**
- **irqu[3:0]**: UART1 IRQ Channel: These bits define the selection of IRQ channel for UART1 as the table shown below. It can be set to ISA mode default by pull-up DRQ_B.
- **irqv[3:0]**: UART2 IRQ Channel: These bits define the selection of IRQ channel for UART2 as the table shown below. It can be set to ISA mode default by pull-up DRQ_B.

### CR29: Configuration Register 29 (CSR(29h))

<table>
<thead>
<tr>
<th>Address Type</th>
<th>bit7</th>
<th>bit6</th>
<th>bit5</th>
<th>bit4</th>
<th>bit3</th>
<th>bit2</th>
<th>bit1</th>
<th>bit0</th>
</tr>
</thead>
<tbody>
<tr>
<td>csr(29h)</td>
<td>WR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>irqi3</td>
<td>irqi2</td>
<td>irqi1</td>
</tr>
<tr>
<td>Reset Default</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0/1</td>
</tr>
</tbody>
</table>

**Bit Name**
- **irqi[3:0]**: IRQIN IRQ Channel: These bits define the selection of IRQ channel forIRQIN as the table shown below. It can be set to ISA mode default by pull-up DRQ_B.

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>None</td>
</tr>
<tr>
<td>0001</td>
<td>IRQ_A</td>
</tr>
<tr>
<td>0010</td>
<td>IRQ_B</td>
</tr>
<tr>
<td>0011</td>
<td>IRQ_C</td>
</tr>
<tr>
<td>0100</td>
<td>IRQ_D</td>
</tr>
<tr>
<td>0101</td>
<td>IRQ_E</td>
</tr>
<tr>
<td>0110</td>
<td>IRQ_F</td>
</tr>
<tr>
<td>0111</td>
<td>Reserved</td>
</tr>
<tr>
<td>1000</td>
<td>IRQ_H</td>
</tr>
<tr>
<td>Others</td>
<td>Reserved</td>
</tr>
</tbody>
</table>
・Configuration register 40 (CR40)

This register is used to indicate the default setup for some specific pins or registers. The terminal status of the pins will reflect to these bits. By setting the external pull-up or not will change the value of this resistors to the ISA default value. The definition of the bits are:

<table>
<thead>
<tr>
<th>CR40: Configuration Register 40 (CSR(40h))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
</tr>
<tr>
<td>csr(27h)</td>
</tr>
<tr>
<td>Reset Default</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| 7 | regdef | Register Definition: This bit defines the power on default of irrxp.CR0C via DRQ_A being pulled up externally or not. The definition is:
0: (Internal pull-down), power-on default of irrxp.CR0C is 0.
1: (External pull-up), power-on default of irrxp.CR0C is 1. |
| 6 | isadef | ISA Definitivn: This bit defines the power on default of ideen1.CR00 and CR[20:29] via DRQ_B being pulled up externally or not. The definition is:
0: (Internal pull-down), power-on defaults are set to Normal mode.
1: (External pull-up), power-on defaults are set to ISA mode. |
| 5:0 | 0 | Reserved and read only. |

Advanced Information

Definition of DDEN[1:0]

The pins are controlled by several registers and defined as following:

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>Register</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>p18s.CR03</td>
</tr>
<tr>
<td></td>
<td>fdndt[0:1].CR1F</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>(0,0)</td>
</tr>
<tr>
<td>DDEN1</td>
<td>DDEN1</td>
</tr>
<tr>
<td>DDEN0</td>
<td>DDEN0</td>
</tr>
<tr>
<td>Note</td>
<td>DDEN[1:0] is defined by fdndt[0:1].CR1F</td>
</tr>
</tbody>
</table>

When the pins are defined as nIDERST and DDEN0, the pin nIDERST is always driven as negated RESET signal of ISA-bus. The pin DDEN0 is then followed with the assignment defined by fdndt[0:1].CR1F as similar as the other case that the pins are defined as DDEN1 and DDEN0. By the way, the assignments of these pins are depended on the value of fdndt[0:1].CR1F being selected currently.
In addition, the functions of DRATE[1:0] and DENSEL are controlled by several registers and defined as following:

* for DENSEL

<table>
<thead>
<tr>
<th>dens[1:0].CR05</th>
<th>ident.CR03</th>
<th>drate[1:0].DSR/CCR</th>
<th>DENSEL</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>dens1 dens0</td>
<td></td>
<td>d rate1 d rate0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0 0</td>
<td>0</td>
<td>1 0</td>
<td>0</td>
<td>PS/2 mode (3.5” FDD).</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1 0</td>
<td>1</td>
<td>AT mode (5.25” FDD).</td>
</tr>
<tr>
<td>1 1</td>
<td>x</td>
<td>x x</td>
<td>1</td>
<td>Fixed high output.</td>
</tr>
<tr>
<td>1 1</td>
<td>x</td>
<td>x x</td>
<td>0</td>
<td>Fixed low output.</td>
</tr>
</tbody>
</table>

* for DRATE[1:0]

<table>
<thead>
<tr>
<th>d rate1.DSR/CCR</th>
<th>d rate0.DSR/CCR</th>
<th>DRATE1</th>
<th>DRATE0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Normally, when ident.CR03 is configured as AT mode (5.25” FDD), DENSEL output is high when the data rate is 500Kbps or above, and is low when the data rate is 300Kbps or 250Kbps. If the bit is configured as PS/2 mode (3.5” FDD), DENSEL output is then driven in opposite of the definition in the AT-mode depended on the data rate defined in d rate[1:0].DSR/CCR. When the system supports 2MB tape driver or 3-mode FDD, the definition of the data rate is different and is controlled by d rt[1:0].CR0B.
The definition of data rate is configured by \texttt{drt[1:0].CR0B} and \texttt{drate[1:0].DSR/CCR} as following:

<table>
<thead>
<tr>
<th>\texttt{drt[1:0].CR0B}</th>
<th>\texttt{drate[1:0].DSR/CCR}</th>
<th>\texttt{Data Rate}</th>
<th>\texttt{Note}</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{drt1}</td>
<td>\texttt{drt0}</td>
<td>\texttt{drate1}</td>
<td>\texttt{drate0}</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

When system is set to AT mode, 3-mode drive supporting, i.e. \texttt{drt[1:0].CR0B= (0,1)}, the FDD can be configured as 300 rpm, 500Kbps, 2/1MB mode (traditional 3.5” FDD) by setting DENSEL to high. In this case, \texttt{drate[1:0].DSR/CCR} need to be set to (0,0). In the other case, 360rpm, 500Kbps, 1.6MB mode, DENSEL must be low and \texttt{drate[1:0].DSR/CCR} need to be set to (0,1). Below shown is the format control of the 3-mode FDD:

<table>
<thead>
<tr>
<th>\texttt{FD Type}</th>
<th>Formatted</th>
<th>Density</th>
<th>Motor Speed</th>
<th>Data Rate</th>
<th>DENSEL</th>
<th>HD</th>
</tr>
</thead>
<tbody>
<tr>
<td>2MB</td>
<td>1.44MB</td>
<td>2HD</td>
<td>300rpm</td>
<td>500Kbps</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1.6MB</td>
<td>1.2MB</td>
<td>2HD</td>
<td>360rpm</td>
<td>500Kbps</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1MB</td>
<td>720KB</td>
<td>2DD</td>
<td>300rpm</td>
<td>250Kbps</td>
<td>—</td>
<td>0</td>
</tr>
</tbody>
</table>

Note: HD input is unused on the FDD interface.
FDC port swapping

Programming `ppfdcits of CR04 with 01/10` will swap the FDD I/F to parallel port. There are two kinds of swap mode. One is B-disk swap only (PPFD1 mode), the other one is two disk swap (PPFD2 mode). When the swap is enabled, parallel port can not be used as normal until the mode has been reset. Write of SPP_CPR is inhibited, but SPP_DPR can be accessible with no meaning. In addition, the status lines of LPT are stuck on the specific value for read back and the read back of SPP_CPR will be bypassed to a guided register for responding "Cable not connected". In PPFD1 mode, nSTB and PD6 signal are set as input for dummy, and the FDC status inputs from both connectors (FDD and LPT) are combined internally. In PPFD2 mode, nSTB and PD6 are changed to output for the second disk, and the pins of FDC interface are tri-stated for disable. The related system pinouts or registers for parallel port during swap mode are defined as no operation (inactive) as following:

<table>
<thead>
<tr>
<th>Pinout</th>
<th>Assignment</th>
<th>Register</th>
<th>Assignment (On Read)</th>
</tr>
</thead>
<tbody>
<tr>
<td>nDACK_lpt</td>
<td>Keep the same status</td>
<td>SPP_DPRData Reg</td>
<td>last SPP_DPR(write)</td>
</tr>
<tr>
<td>DRQ_lpt</td>
<td><img src="#" alt="ECP: Hi-ZEC &amp; dmaen: 0ECP &amp; ! dmaen:Hi-Z" /></td>
<td>SPP_CPRControl Reg</td>
<td>&quot;Cable not connected&quot;, means: strobe, autofd, slc= 0 ninit= 1</td>
</tr>
<tr>
<td>IRQ_lpt</td>
<td>Hi-Z or 0, depending on setting</td>
<td>SPP_SPRStatus Reg</td>
<td>nbusy, pe, slct= 0 nack, nerr =1</td>
</tr>
</tbody>
</table>

The interface pinouts of FDD as reassigned as following:

<table>
<thead>
<tr>
<th>Connector Pin No.</th>
<th>Chip Pin No.</th>
<th>Spp Mode Pin Name</th>
<th>Pin Attribute</th>
<th>FDC Swap Pin Name</th>
<th>Pin Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>77</td>
<td>nSROTBE</td>
<td>I/O</td>
<td>nDS0</td>
<td>I/(O)</td>
</tr>
<tr>
<td>2</td>
<td>71</td>
<td>PD0</td>
<td>I/O</td>
<td>nIDX</td>
<td>I</td>
</tr>
<tr>
<td>3</td>
<td>70</td>
<td>PD1</td>
<td>I/O</td>
<td>nTRK0</td>
<td>I</td>
</tr>
<tr>
<td>4</td>
<td>69</td>
<td>PD2</td>
<td>I/O</td>
<td>nWP</td>
<td>I</td>
</tr>
<tr>
<td>5</td>
<td>68</td>
<td>PD3</td>
<td>I/O</td>
<td>nRDD</td>
<td>I</td>
</tr>
<tr>
<td>6</td>
<td>66</td>
<td>PD4</td>
<td>I/O</td>
<td>nDCHG</td>
<td>I</td>
</tr>
<tr>
<td>7</td>
<td>65</td>
<td>PD5</td>
<td>I/O</td>
<td></td>
<td>I</td>
</tr>
<tr>
<td>8</td>
<td>64</td>
<td>PD6</td>
<td>I/O</td>
<td>(nMOE0)</td>
<td>I/(O)</td>
</tr>
<tr>
<td>9</td>
<td>63</td>
<td>PD7</td>
<td>I/O</td>
<td></td>
<td>I</td>
</tr>
<tr>
<td>10</td>
<td>62</td>
<td>nACK</td>
<td>I</td>
<td>nDS1</td>
<td>O</td>
</tr>
<tr>
<td>11</td>
<td>61</td>
<td>BUSY</td>
<td>I</td>
<td>nMOE1</td>
<td>O</td>
</tr>
<tr>
<td>12</td>
<td>60</td>
<td>PE</td>
<td>I</td>
<td>nWDD</td>
<td>O</td>
</tr>
<tr>
<td>13</td>
<td>59</td>
<td>SLCT</td>
<td>I</td>
<td>nWG</td>
<td>O</td>
</tr>
<tr>
<td>14</td>
<td>76</td>
<td>nAFD</td>
<td>I/O</td>
<td>DENSEL</td>
<td>O</td>
</tr>
<tr>
<td>15</td>
<td>75</td>
<td>nERROR</td>
<td>I</td>
<td>nHSEL</td>
<td>O</td>
</tr>
<tr>
<td>16</td>
<td>74</td>
<td>nINIT</td>
<td>I/O</td>
<td>nDIR</td>
<td>O</td>
</tr>
<tr>
<td>17</td>
<td>73</td>
<td>nSLCTIN</td>
<td>I/O</td>
<td>nSTEP</td>
<td>O</td>
</tr>
</tbody>
</table>

( ): additional assignment in PPFD2 mode.
Power management

There are two kinds of power management. The first one is auto-powerdown mode, the other one is direct power down mode.

Using auto-powerdown mode can save power during the normal operation. In this mode each subsystem can monitor the system accesses and the status from target interface to decide to enter the powerdown mode and can be recovered immediately while system accesses the subsystem or some specific status from target interfaces. The status of each subsystem is software transparency for system power management. In addition, the register sets are unchanged, and most of the related interface are still active except some control pins.

For the implementation of system power management, direct powerdown mode can be used for the control transparency by system power management unit. In the direct power down mode, subsystems are controlled via setting or clearing the related registers for function disabling or enabling. Eventually, the register contents are keep unchanged. But for the consistency of the system setting for reconfigurable subsystem function, we recommend system should configure the related subsystem again after the subsystem being set to exit direct power down mode, except the configuration is the same as the previous setting and the chip power is still available during the mode changes. In addition, the related interface pins are mostly being turned off, except some status signals.

• FDC power management
  * Description:

<table>
<thead>
<tr>
<th>Mode</th>
<th>Entry</th>
<th>Pri</th>
<th>Functions</th>
<th>Exit</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>Reset fdcpwr. CR00 then monitor the status of FDC till IDLE or APD.</td>
<td>H</td>
<td>- Set status to PD.</td>
<td>Set fdcpwr.CR00</td>
<td>- Set status back to IDLE.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Disable clock, DDS and most of block.</td>
<td></td>
<td>- All functions recovery.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Registers set to RO and keep the same contents.</td>
<td></td>
<td>- Recover the auto power down mode if the mode is set before. But reset the fdclpd.FDC_DSR if the mode is set before.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Input: All inactive.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Output: All tri-state.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mode</td>
<td>Entry</td>
<td>Pri</td>
<td>Functions</td>
<td>Exit</td>
<td>Functions</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
<td>-----</td>
<td>-----------</td>
<td>------</td>
<td>-----------</td>
</tr>
</tbody>
</table>
| DSR  | Set fdclpd. FDC_DSR then monitor the status of FDC till IDLE or APD. | M   | - Set status to APD.  
- Disable the APD mode if the mode is set.  
- Disable clock and DDS.  
- Registers keeps the same contents.  
- Input: All active  
- Output: Tri-state: nWE, nMOE[0:1], nWRDATA, DS[0:1].  
  Active: nHSEL, nSTEP, nDIR, DDEN0.  
Access  
FDC_MSR or FDC_DR, or doing a software reset via FDC_DOR/ FDC_DSR, or enable one of the moten[1:0].FDC_DOR  
- Set status to BUSY.  
- All functions recovery.  
- Recover the auto power down mode if the mode is set before. But reset the fdclpd. FDC_DSR if the mode is set before. |
| Auto | Set fcapd.CR07 then monitor the IDLE state:  
- nMOE[1:0] inactive, moten[1:0].  
  FDC_DOR set to zero.  
- State machine is idle, FDC_MSR= 80h and INT=0 (no polling).  
- Internal Head Unload Timer must be expired. Then check APD Timer (10 ms) is timeout. The timer is reset for any accesses of FDC_MSR or FDC_DR during the count down. | L   | - Set status to APD.  
- Disable clock and DDS.  
- Registers keeps the same contents.  
- Input: All active  
- Output: Tri-state: nWE, nMOE[0:1], nWRDATA, DS[0:1].  
  Active: nHSEL, nSTEP, nDIR, DDEN0.  
Access  
FDC_MSR or FDC_DR, or doing a software reset via FDC_DOR/ FDC_DSR, or enable one of the moten[1:0].FDC_DOR  
- Set status back to IDLE.  
- All functions recovery. |
Notice:
While system changes the state from APD state to BUSY state, fdclpd.FDC_DSR will be reset automatically. Eventually, fdcapd.CR07 is keep unchanged during the changes. The status of state will be output to the power management for the power control.
In the APD state, system interface input pins are kept unchanged, output pins are kept unchanged with no operation (i.e. IRQ_FDC and DRQ_FDC are low). In the PD state, system interface input pins are the same as APD state, but IRQ_FDC and DRQ_FDC is floated.
• Serial port power management
  
  Description

<table>
<thead>
<tr>
<th>Mode</th>
<th>Entry</th>
<th>Pri</th>
<th>Functions</th>
<th>Exit</th>
<th>Functions</th>
</tr>
</thead>
</table>
| Direct | Reset cm1/2pwr. CR02 then monitor the status of COM1/2 till transmitter and receiver are all be in IDLE or APD state. | H | - Set status to PD.  
- Disable clock and most of block.  
- Registers set to RO and keep the same contents.  
- Input: All inactive.  
- Output: Inactive. | Set cm1/2pwr.CR02 | - Set status back to IDLE.  
- All functions recovery.  
- Recover the auto power down mode if the mode is set before. |
| Auto | Set cm1/2apd.CR07 then monitor the IDLE state of transmitter or receiver:  
  a. Transmitter:  
  - CMn_THR and shift register are empty.  
  b. Receiver:  
  - CMn_RBR (or receive FIFO) is empty, and the receiver is waiting for a start bit. | L | - Set status to APD.  
- Disable clock.  
- Registers keeps the same contents.  
- Input: All active but gated, except RI and SIn.  
- Output: Inactive. | a. A write to the transmitter buffer.  
b. Pin SIn changes state.  
c. RI input is toggled. | a. Transmit recovery:  
- Set transmit status to BUSY.  
- Recover transmitter  
b. Receive recovery:  
- Set receive status to BUSY.  
- Recover receiver.  
c. All recovery:  
- Set status back to BUSY.  
- All functions recovery. |
Notice:
System does not change the output pins to tri-state during the power down mode in order to avoid the hazard state on buffer chip.
RI interrupts are kept valid and transitions when nRIn inputs changes during auto powerdown mode.
**Parallel Port power management**

* Description

<table>
<thead>
<tr>
<th>Mode</th>
<th>Entry</th>
<th>Pri</th>
<th>Functions</th>
<th>Exit</th>
<th>Functions</th>
</tr>
</thead>
</table>
| Direct | Reset ltpwr.CR01. | H | - Disable clock and most of block.  
- Registers set to RO and keep the same contents.  
- Input: All inactive.  
- Output: All tri-state. | Set ltpwr.CR01 | - All functions recovery.  
- Recover the auto power down mode if the mode is set before. |
| Auto | Set ltpad.CR07 then check the configuration status of EPP and ECP:  
a. EPP:  
+ EPP is not enabled in the configuration registers.  
+ EPP is not selected through ecr while in ECP mode.  
b. ECP:  
+ ECP is not enabled in the configuration registers.  
+ SPP, PS/2 Parallel port or EPP mode is selected through ecr while in ECP mode. | L | - Disable EPP or ECP clock.  
- Registers keeps the same contents.  
- Input: Active.  
- Output: Active. | + Reset ltpad.CR07  
+ ECP mode is changed through ecr register.  
+ The parallel port mode is changed via the configuration register. | + All functions recovered if reset ltpad.CR07  
+ Recover related block if it is activated via the reconfiguration. |

**Notice:**  
This block does not support status outputs for system request. The power status can be identified via reading back ltpwr.CR01 or ltpad.CR07
MISC

• Application notice:
  For the completion of power on initialization, PWRGD should be connected to system power good signal correctly. If the pin is failed to connect with power good signal, the power on initialization will be configured incorrectly. In such case, we recommend that using BIOS reconfiguration with the default value for all of the configuration registers is the best way to avoid the potential confictions.

• The polarity control of Ir-DA interface
  For the ease of IR setup, we can use an external pull up resistor to put on the DRQ_A pin. It will reflect to the bit 7 of Holtek private setup register, regdef.CR40, for cold start setup and enable the internal control logic for the polarity of Ir-DA interface. When it is enabled, the default value of irrxp.CR0C is changed from 0 to 1 after cold reset, and the polarity of IR receiver is changed.
## Operational Descriptions

### DC characteristics

<table>
<thead>
<tr>
<th>Buffer Type</th>
<th>Symbol</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
<th>Parameter</th>
<th>Test Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Power</strong></td>
<td>V\text{CC}</td>
<td>4.75</td>
<td></td>
<td>5.25</td>
<td>V</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>I Normal Input</strong></td>
<td>V\text{IL}</td>
<td>-0.5</td>
<td></td>
<td>0.8</td>
<td>V</td>
<td>Low level input</td>
<td>TTL level</td>
</tr>
<tr>
<td></td>
<td>V\text{IH}</td>
<td>2.0</td>
<td></td>
<td>5.5</td>
<td>V</td>
<td>High level input</td>
<td>TTL level</td>
</tr>
<tr>
<td></td>
<td>I\text{IL}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>Low input leakage</td>
<td>Vin= GND</td>
</tr>
<tr>
<td></td>
<td>I\text{IH}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>High input leakage</td>
<td>Vin= V\text{CC}</td>
</tr>
<tr>
<td><strong>ID Input with pull-down</strong></td>
<td>V\text{IL}</td>
<td>-0.5</td>
<td></td>
<td>0.8</td>
<td>V</td>
<td>Low level input</td>
<td>TTL level</td>
</tr>
<tr>
<td></td>
<td>V\text{IH}</td>
<td>2.0</td>
<td></td>
<td>5.5</td>
<td>V</td>
<td>High level input</td>
<td>TTL level</td>
</tr>
<tr>
<td></td>
<td>I\text{IL}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>Low input leakage</td>
<td>Vin= GND</td>
</tr>
<tr>
<td></td>
<td>I\text{IH}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>High input leakage</td>
<td>Vin= V\text{CC}</td>
</tr>
<tr>
<td><strong>IS Input schmitt-trigger</strong></td>
<td>V\text{IL}</td>
<td>-0.5</td>
<td></td>
<td>0.8</td>
<td>V</td>
<td>Low level input</td>
<td>TTL level</td>
</tr>
<tr>
<td></td>
<td>V\text{IH}</td>
<td>2.0</td>
<td></td>
<td>5.5</td>
<td>V</td>
<td>High level input</td>
<td>TTL level</td>
</tr>
<tr>
<td></td>
<td>I\text{IL}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>Low input leakage</td>
<td>Vin= GND</td>
</tr>
<tr>
<td></td>
<td>I\text{IH}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>High input leakage</td>
<td>Vin= V\text{CC}</td>
</tr>
<tr>
<td><strong>O4 4mA output</strong></td>
<td>V\text{OL}</td>
<td>0.4</td>
<td></td>
<td></td>
<td>V</td>
<td>Low level output</td>
<td>I\text{OL}=4mA</td>
</tr>
<tr>
<td></td>
<td>V\text{OH}</td>
<td>2.4</td>
<td></td>
<td></td>
<td>V</td>
<td>High level output</td>
<td>I\text{OH}=4mA</td>
</tr>
<tr>
<td></td>
<td>I\text{OL}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>Low output leakage</td>
<td>Vin= GND</td>
</tr>
<tr>
<td></td>
<td>I\text{OH}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>High output leakage</td>
<td>Vin= V\text{CC}</td>
</tr>
<tr>
<td><strong>OT4 4mA output with Tri-state</strong></td>
<td>V\text{OL}</td>
<td>0.4</td>
<td></td>
<td></td>
<td>V</td>
<td>Low level output</td>
<td>I\text{OL}=4mA</td>
</tr>
<tr>
<td></td>
<td>V\text{OH}</td>
<td>2.4</td>
<td></td>
<td></td>
<td>V</td>
<td>High level output</td>
<td>I\text{OH}=4mA</td>
</tr>
<tr>
<td></td>
<td>I\text{OL}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>Low output leakage</td>
<td>Vin= GND</td>
</tr>
<tr>
<td></td>
<td>I\text{OH}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>High output leakage</td>
<td>Vin= V\text{CC}</td>
</tr>
<tr>
<td><strong>OD8U 8mA output open-drain with pull-up</strong></td>
<td>V\text{OL}</td>
<td>0.4</td>
<td></td>
<td></td>
<td>V</td>
<td>Low level output</td>
<td>I\text{OL}=8mA</td>
</tr>
<tr>
<td></td>
<td>I\text{OL}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>Low output leakage</td>
<td>Vin= GND</td>
</tr>
<tr>
<td></td>
<td>I\text{OH}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>High output leakage</td>
<td>Vin= V\text{CC}</td>
</tr>
<tr>
<td><strong>O24 24mA output</strong></td>
<td>V\text{OL}</td>
<td>0.4</td>
<td></td>
<td></td>
<td>V</td>
<td>Low level output</td>
<td>I\text{OL}=24mA</td>
</tr>
<tr>
<td></td>
<td>V\text{OH}</td>
<td>2.4</td>
<td></td>
<td></td>
<td>V</td>
<td>High level output</td>
<td>I\text{OH}=24mA</td>
</tr>
<tr>
<td></td>
<td>I\text{OL}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>Low output leakage</td>
<td>Vin= GND</td>
</tr>
<tr>
<td></td>
<td>I\text{OH}</td>
<td>-10</td>
<td></td>
<td>+10</td>
<td>(\mu\text{A})</td>
<td>High output leakage</td>
<td>Vin= V\text{CC}</td>
</tr>
</tbody>
</table>

3rd Jan '97
<table>
<thead>
<tr>
<th>Buffer Type</th>
<th>Symbol</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
<th>Parameter</th>
<th>Test Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>O24P 24mA output</td>
<td>VOL</td>
<td>0.4</td>
<td>V</td>
<td>Low level output</td>
<td>IOL=24mA</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VOH</td>
<td>2.4</td>
<td>V</td>
<td>High level output</td>
<td>IOH=24mA</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IOL</td>
<td>–10</td>
<td>+10</td>
<td>µA</td>
<td>Low output leakage</td>
<td>Vin= GND</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IOH</td>
<td>–10</td>
<td>+10</td>
<td>µA</td>
<td>High output leakage</td>
<td>Vin= VCC</td>
<td></td>
</tr>
<tr>
<td>OD24 24mA output open-drain</td>
<td>VOL</td>
<td>0.4</td>
<td>V</td>
<td>Low level output</td>
<td>IOL=24mA</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IOL</td>
<td>–10</td>
<td>+10</td>
<td>µA</td>
<td>Low output leakage</td>
<td>Vin= GND</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IOH</td>
<td>–10</td>
<td>+10</td>
<td>µA</td>
<td>High output leakage</td>
<td>Vin= VCC</td>
<td></td>
</tr>
<tr>
<td>OD24U 24mA output open-drain</td>
<td>VOL</td>
<td>0.4</td>
<td>V</td>
<td>Low level output</td>
<td>IOL=24mA</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IOL</td>
<td>–10</td>
<td>+10</td>
<td>µA</td>
<td>Low output leakage</td>
<td>Vin= GND</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IOH</td>
<td>–10</td>
<td>+10</td>
<td>µA</td>
<td>High output leakage</td>
<td>Vin= VCC</td>
<td></td>
</tr>
<tr>
<td>OD48 48mA output open-drain</td>
<td>VOL</td>
<td>0.5</td>
<td>V</td>
<td>Low level output</td>
<td>IOL=48mA</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IOL</td>
<td>–10</td>
<td>+10</td>
<td>µA</td>
<td>Low output leakage</td>
<td>Vin= GND</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IOH</td>
<td>–10</td>
<td>+10</td>
<td>µA</td>
<td>High output leakage</td>
<td>Vin= VCC</td>
<td></td>
</tr>
</tbody>
</table>
# AC Characteristics

**FDC: Data rate=1000/500/300/250 KB/SEC**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.*</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAR</td>
<td>HA[10:0], AEN, nDACK Setup time to nIOR ↓</td>
<td>—</td>
<td>25</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TRA</td>
<td>HA[10:0], AEN, nDACK hold time from nIOR ↑</td>
<td>100pf Loading</td>
<td>0</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TRR</td>
<td>nIOR width</td>
<td>100pf Loading</td>
<td>200</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TFD</td>
<td>Data access time from nIOR ↓</td>
<td>100pf Loading</td>
<td>—</td>
<td>—</td>
<td>80</td>
<td>ns</td>
</tr>
<tr>
<td>TDH</td>
<td>Data hold time from nIOR ↓</td>
<td>—</td>
<td>10</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TDF</td>
<td>HD to float from nIOR ↑</td>
<td>—</td>
<td>10</td>
<td>—</td>
<td>50</td>
<td>ns</td>
</tr>
<tr>
<td>TRI</td>
<td>IRQ Delay from nIOR ↑</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TAW</td>
<td>HA[10:0], AEN, nDACK setup time to nIOW ↓</td>
<td>—</td>
<td>25</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TWA</td>
<td>HA[10:0], AEN, nDACK hold time from nIOW ↑</td>
<td>—</td>
<td>0</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TWW</td>
<td>nIOW width</td>
<td>—</td>
<td>200</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TDW</td>
<td>Data setup time to nIOW ↑</td>
<td>—</td>
<td>60</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TWD</td>
<td>Data hold time from nIOW ↑</td>
<td>—</td>
<td>0</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TWI</td>
<td>IRQ delay from nIOW ↑</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TMcy</td>
<td>DMA cycle time</td>
<td>—</td>
<td>27</td>
<td>—</td>
<td>—</td>
<td>µs</td>
</tr>
<tr>
<td>TAM</td>
<td>DMA reset delay time from nDACK ↓</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>50</td>
<td>ns</td>
</tr>
<tr>
<td>TMA</td>
<td>DRQ to nDACK delay</td>
<td>—</td>
<td>0</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TAA</td>
<td>nDACK width</td>
<td>—</td>
<td>260</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TMR</td>
<td>nIOR delay from DRQ</td>
<td>—</td>
<td>0</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TMW</td>
<td>nIOW delay from DRQ</td>
<td>—</td>
<td>0</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>TMRW</td>
<td>nIOW or nIOR response time from DRQ</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>12</td>
<td>µs</td>
</tr>
<tr>
<td>TFC</td>
<td>TC width</td>
<td>—</td>
<td>135</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
</tbody>
</table>

---

* HT8669IR 63 3rd Jan '97
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.*</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRST</td>
<td>RESET width</td>
<td>—</td>
<td>1.8/</td>
<td>3.0/</td>
<td>3.5/</td>
<td>µs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>TIDX</td>
<td>nIDX width</td>
<td>—</td>
<td>0.5/</td>
<td>0.9/</td>
<td>1.0/</td>
<td>µs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>TDST</td>
<td>nDIR setup time to nSTEP</td>
<td>—</td>
<td>1.0/</td>
<td>1.6/</td>
<td>2.0/</td>
<td>µs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>TSTD</td>
<td>nDIR hold time from nSTEP</td>
<td>—</td>
<td>24/</td>
<td>40/</td>
<td>48/</td>
<td>µs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>TSTEP</td>
<td>nSTEP pulse width</td>
<td>—</td>
<td>6.8/</td>
<td>11.5/</td>
<td>13.8/</td>
<td>µs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>TSC</td>
<td>nSTEP cycle time</td>
<td>—</td>
<td>**</td>
<td>**</td>
<td>**</td>
<td>µs</td>
</tr>
<tr>
<td>TWDD</td>
<td>nWDD pulse width</td>
<td>—</td>
<td>37/</td>
<td>100/</td>
<td>188/</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>62/</td>
<td>125/</td>
<td>210/</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>250</td>
<td>275</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWPC</td>
<td>Write Precompensation</td>
<td>—</td>
<td>37/</td>
<td>62/</td>
<td>87/</td>
<td>µs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>100/</td>
<td>125/</td>
<td>150/</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>188/</td>
<td>210/</td>
<td>235/</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>225</td>
<td>250</td>
<td>275</td>
<td></td>
</tr>
</tbody>
</table>

Notes:
* Typical n-values for TA=25°C and nominal value for supply voltage.
** Programmable from 2ms to 32ms in 2ms increments.

- Processor read operation

Diagram of processor read operation:

- HAEN, HA[10:0]
- nDACK
- nIOR
- HD[7:0]
- IRQ
• Processor write operation

HAEN, HA[10:0]  nDACK
nDACK
nIOW
HD[7:0]
IRQ

• DMA operation

DRQ
nDACK
nIOW or nIOR

• Terminal count

TC
RESET
nIDX
**FDD write/read operation**

- nWDD
- nRDD
- nIDX

**Seek operation**

- nDIR
- nSTEP
- RESET

**UART**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIRS</td>
<td>Delay from initial IRQ reset to transmit start</td>
<td>—</td>
<td>1/16</td>
<td>—</td>
<td>8/16</td>
<td>Baud Rate</td>
</tr>
<tr>
<td>TSTI</td>
<td>Delay from stop to interrupt</td>
<td>—</td>
<td>9/16</td>
<td>—</td>
<td>—</td>
<td>Baud Rate</td>
</tr>
<tr>
<td>THR</td>
<td>Delay from nIOW to reset interrupt</td>
<td>100pf Loading</td>
<td>—</td>
<td>—</td>
<td>175</td>
<td>ns</td>
</tr>
<tr>
<td>TSI</td>
<td>Delay from initial nIOW to interrupt</td>
<td>—</td>
<td>9/16</td>
<td>—</td>
<td>16/16</td>
<td>Baud Rate</td>
</tr>
<tr>
<td>TIR</td>
<td>Delay from nIOR to reset</td>
<td>100pf Loading</td>
<td>—</td>
<td>—</td>
<td>1</td>
<td>ns</td>
</tr>
<tr>
<td>TSINT</td>
<td>Delay from stop to set interrupt</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>1/2</td>
<td>Baud Rate</td>
</tr>
<tr>
<td>TRINT</td>
<td>Delay from nIOR reset interrupt</td>
<td>100pf Loading</td>
<td>—</td>
<td>—</td>
<td>250</td>
<td>ns</td>
</tr>
<tr>
<td>TMWO</td>
<td>Delay from nIOW to output</td>
<td>100pf Loading</td>
<td>—</td>
<td>—</td>
<td>200</td>
<td>ns</td>
</tr>
<tr>
<td>TSIM</td>
<td>Set interrupt delay from MODEM input</td>
<td>100pf Loading</td>
<td>—</td>
<td>—</td>
<td>250</td>
<td>ns</td>
</tr>
<tr>
<td>TRIM</td>
<td>Reset interrupt Delay from nIOR</td>
<td>100pf Loading</td>
<td>—</td>
<td>—</td>
<td>250</td>
<td>ns</td>
</tr>
<tr>
<td>N</td>
<td>Baud rate divisor</td>
<td>100pf Loading</td>
<td>—</td>
<td>—</td>
<td>$2^{16} - 1$</td>
<td>unit</td>
</tr>
</tbody>
</table>
• Transmitter timing

SO1, SO2

SIRQ1
SIRQ2

nIOW
(Write THR)
nIOR
(Read IIR)

• Receiver timing

SI1, SI2

SIRQ1
SIRQ2

• MODEM control timing

nIOW
(Write MCR)
nRTS, nDTR

nCTS, nDSR
nDCD

SIRQ1
SIRQ2

nIOR
(Read MSR)
nRI

67
**EPP address or data write cycle timing**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Max.</th>
<th>Units</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_{HWI} )</td>
<td>10</td>
<td>80</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>( t_{HWS} )</td>
<td>140</td>
<td>240</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>( t_{HWP} )</td>
<td>210</td>
<td></td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>( t_{PTO} )</td>
<td>10</td>
<td>12</td>
<td>( \mu )s</td>
<td></td>
</tr>
<tr>
<td>( t_{PSC} )</td>
<td>70</td>
<td>80</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>( t_{PCB} )</td>
<td>0</td>
<td>10</td>
<td>( \mu )s</td>
<td></td>
</tr>
<tr>
<td>( t_{PBY} )</td>
<td>220</td>
<td>290</td>
<td>ns</td>
<td>1</td>
</tr>
<tr>
<td>( t_{PBL} )</td>
<td>280</td>
<td>350</td>
<td>ns</td>
<td>1</td>
</tr>
<tr>
<td>( t_{PLY} )</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( t_{PYB} )</td>
<td>280</td>
<td>350</td>
<td>ns</td>
<td>1</td>
</tr>
<tr>
<td>( t_{PYI} )</td>
<td>350</td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

BUSY must be filtered to compensate for ringing on the parallel bus cable. BUSY is considered to have settle after it does not transition for a minimum of 210 \( \mu \)s.
Parallel port timing

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Max.</th>
<th>Units</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>t(_{HWC})</td>
<td>400</td>
<td></td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>t(_{PKI})</td>
<td></td>
<td>1.25</td>
<td>(\mu s)</td>
<td></td>
</tr>
<tr>
<td>t(_{PIW})</td>
<td>100</td>
<td>200</td>
<td>ns</td>
<td>1</td>
</tr>
<tr>
<td>t(_{PEI})</td>
<td></td>
<td>140</td>
<td>ns</td>
<td></td>
</tr>
</tbody>
</table>

The IRQ pulse width is in the range: 300~400ns when operating in ECP interrupt I/O or DMA I/O.
### Parameter Timing

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Max.</th>
<th>Units</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>tHRI</td>
<td>0</td>
<td>80</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>tHRB</td>
<td>0</td>
<td></td>
<td>ns</td>
<td>2</td>
</tr>
<tr>
<td>tHRZ</td>
<td>0</td>
<td>70</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>tPTO</td>
<td>10</td>
<td>12</td>
<td>µs</td>
<td></td>
</tr>
<tr>
<td>tPZC</td>
<td>140</td>
<td>210</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>tPCD</td>
<td>0</td>
<td></td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>tPDB</td>
<td>0</td>
<td></td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>tPBY</td>
<td>280</td>
<td>350</td>
<td>ns</td>
<td>1</td>
</tr>
<tr>
<td>tPLZ</td>
<td>350</td>
<td>420</td>
<td>ns</td>
<td>1</td>
</tr>
<tr>
<td>tPZY</td>
<td>0</td>
<td></td>
<td>ns</td>
<td></td>
</tr>
</tbody>
</table>

BUSY is considered to have settled after it does not transition for a minimum of 210 µs.

---

**Diagram:**

- HA[10:0]
- nIOR
- HD
- IOCHRDY
- nSTB
- PD[7:0]
- nAFD
- nSLCTIN
- BUSY
**ECP parallel port FIFO mode timing**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Max.</th>
<th>Units</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_{PDS} )</td>
<td>500</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( t_{PSW} )</td>
<td>500</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( t_{PBD} )</td>
<td>500</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( t_{PBS} )</td>
<td>900</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Maximum value only applies if there is data in the FIFO waiting to be written out.

Busy in not considered asserted or deasserted until it is stable for minimum of 280~350 µs.

**ECP parallel port forward timing**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Max.</th>
<th>Units</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_{PDS} )</td>
<td>0</td>
<td>180</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>( t_{PAS} )</td>
<td>0</td>
<td>70</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>( t_{PSB} )</td>
<td>0</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( t_{PBB} )</td>
<td>280</td>
<td>360</td>
<td>ns</td>
<td>2</td>
</tr>
<tr>
<td>( t_{PBY} )</td>
<td>0</td>
<td>ns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( t_{PYX} )</td>
<td>210</td>
<td>350</td>
<td>ns</td>
<td>1, 2</td>
</tr>
<tr>
<td>( t_{PYS} )</td>
<td>560</td>
<td>850</td>
<td>ns</td>
<td>1, 2</td>
</tr>
</tbody>
</table>

Busy in not considered asserted or deasserted until it is stable for minimum of 280~350 µs.
**ECP parallel port reverse timing**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Max.</th>
<th>Units</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_{\text{PDA}} )</td>
<td>PD[7:0], BUSY valid to nACK asserted</td>
<td>0</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>( t_{\text{PAD}} )</td>
<td>nACK asserted to nAFD deasserted</td>
<td>280</td>
<td>450</td>
<td>ns</td>
</tr>
<tr>
<td>( t_{\text{PAK}} )</td>
<td>nAFD asserted to nACK deasserted</td>
<td>0</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>( t_{\text{PKA}} )</td>
<td>nACK deasserted to nAFD asserted</td>
<td>350</td>
<td>500</td>
<td>ns</td>
</tr>
<tr>
<td>( t_{\text{PAX}} )</td>
<td>nAFD asserted to PD[7:0] changed</td>
<td>0</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>( t_{\text{PAA}} )</td>
<td>nAFD asserted to nACK asserted</td>
<td>0</td>
<td></td>
<td>ns</td>
</tr>
</tbody>
</table>

Maximum value only applies if there is room in FIFO and a terminal count has not been received. ACKZ is not considered asserted or deasserted until it is stable for minimum of 280–350 µs.

**MISC**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Max.</th>
<th>Units</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_{\text{HAA}} )</td>
<td>HA[10:0], HAEN and nCS valid to nIDEEN, nIDECS1, nIDECS2, nGAMECS asserted</td>
<td>29</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>( t_{\text{HAD}} )</td>
<td>HA[10:0], HAEN and nCS invalid to nIDEEN, nIDECS1, nIDECS2, nGAMECS deasserted</td>
<td>25</td>
<td></td>
<td>ns</td>
</tr>
</tbody>
</table>

**Diagrams**

- Diagram of ECP parallel port reverse timing parameters.
- Diagram of MISC parameters.