9.6.11 Data Transfer Commands

All data transfer commands utilize the same parameter bytes (except for FORMAT A TRACK command) and return the same result data bytes. The only difference between them is the five bits (0–4) of the first byte.

9.6.11.1 READ DATA

The READ DATA command contains nine command bytes that place the FDC into the Read Data mode. Each READ operation is initialized by a READ DATA command. The FDC locates the sector to be read by matching ID Address Marks and ID fields from the command with the information on the diskette. The FDC then transfers the data to the FIFO. When the data from the given sector has been read, the READ DATA command is completed and the sector address is automatically incremented by 1. The data from the next sector is read and transferred to the FIFO in the same manner. Such a continuous read function is called a "Multi-Sector Read Operation".

If a TC or an implied TC (FIFO overrun/underrun) is received, the FDC stops sending data, but continues to read data from the current sector and checks the CRC bytes until the end of the sector is reached and the read operation is completed.

The sector size is determined by the N parameter value as calculated in the equation below:
Sector Size = $2^{(7+N\ value)}$ bytes.

The DTL parameter determines the number of bytes to be transferred. Therefore, if N = 00h, setting the sector size to 128 and the DTL parameter value is less than this, the remaining bytes will be read and checked for CRC errors by the FDC. If this occurs in a write operation, the remaining bytes will be filled with 0. If the sector size is not 128 (N > 00h), DTL should be set to FFh.

In addition to performing Multi-Sector Read operations, the FDC can perform Multi-Track Read operations. When the MT parameter is set, the FDC can read both sides of a disk automatically.

The combination of N and MT parameter values determines the amount of data that can be transferred during either type of READ operation. Table 9-16 shows the maximum data transfer capacity and the final sector the FDC reads based on these parameters.

Table 9-16. Effects of MT and N Bits

MT	N	Maximum Transfer Capacity	Final Sector Read from Disk
0	1	256 X 26 = 6656	26 on side 0 or side 1
1	1	256 X 52 = 13312	26 on side 1
0	2	512 X 15 = 7680	15 on side 0 or side 1
1	2	512 X 30 = 15360	15 on side 1
0	3	1024 X 8 = 8192	8 on side 0 or side 1
1	3	1024 X 16 = 16384	16 on side 1

9.6.11.2 READ DELETED DATA

The READ DELETED DATA command is the same as the READ DATA command, except that a Deleted Data Address Mark (as opposed to a Data Address Mark) is read at the beginning of the Data Field. This command is typically used to mark a bad sector on a diskette.

9.6.11.3 READ A TRACK

After receiving a pulse from the INDEX# pin, the READ A TRACK command reads the entire data field from each sector of the track as a continuous block. If any ID or Data Field CRC error is found, the FDC continues to read data from the track and indicates the error at the end. Because the Multi-Track [and Skip] operation[s] is[are] not allowed under this command, the MT and SK bits should be low (0) during the command execution.

This command terminates normally when the number of sectors specified by EOT has not been read. If, however, no ID Address Mark has been found by the second occurrence of the INDEX pulse, the FDC will set the IC code in the ST0 to 01, indicating an abnormal termination, and then finish the command.

9.6.11.4 WRITE DATA

The WRITE DATA command contains nine command bytes that place the FDC into the Write Data mode. Each WRITE operation is initialized by a WRITE DATA command. The FDC locates the sector to be written by reading ID fields and matching the sector address from the command with the information on the diskette. Then the FDC reads the data from the host via the FIFO and writes the data into the sector's data field. Finally, the FDC computes the CRC value, storing it in the CRC field, and increments the sector number (stored in the R parameter) by 1. The next data field is written into the next sector in the same manner. Such a continuous writes function is called a "Multi-Sector Write Operation".

If a TC or an implied TC (FIFO overrun/underrun) is received, the FDC stops writing data and fills the remaining data field with 0s. If a check of the CRC value indicates an error in the sector ID Field, the FDC will set the IC code in the ST0 to 01 and the DE bit in the ST1 to 1, indicating an abnormal termination, and then terminate the WRITE DATA command. The maximum data transfer capacity and the DTL, N, and MT parameters are the same as in the READ DATA command.

9.6.11.5 WRITE DELETED DATA

The WRITE DELETED DATA command is the same as the WRITE DATA command, except that a Deleted Data Address Mark (instead of a Data Address Mark) is written at the beginning of the Data Field. This command is typically used to mark a bad sector on a diskette.

9.6.11.6 FORMAT A TRACK

The FORMAT A TRACK command is used to format an entire track. Initialized by an INDEX pulse, it writes data to the Gaps, Address Marks, ID fields and Data fields according to the density mode selected (FM or MFM). The Gap and Data field values are controlled by the host-specified values programmed into N, SC, GPL, and D during the Command phase. The Data field is filled with the data byte specified by D. The four data bytes per sector (C, H, R, and N) needed to fill the ID field are supplied by the host. The C, R, H, and N values must be renewed for each new sector of a track. Only the R parameter value must be changed when a sector is formatted, allowing the disk to be formatted with non-sequential sector addresses. These steps are repeated until a new INDEX pulse is received, at which point the FORMAT A TRACK command is terminated.

9.6.11.7 SCAN

The SCAN command allows the data read from the disk to be compared with the data sent from the system. There are three SCAN commands:

SCAN EQUAL Disk Data = System Data

SCAN HIGH OR EQUAL Disk Data ≥ System Data

SCAN LOW OR EQUAL Disk Data ≤ System Data

The SCAN command execution continues until the scan condition has been met, or the EOT has been reached, or if TC is asserted. Read errors on the disk have the same error condition as the READ DATA command. If the SK bit is set, sectors with deleted data address marks are ignored. If all sectors read are skipped, the command terminates with bit 3 of the ST2 being set. The Result phase of the command is shown below:

Table 9-17. SCAN Command Result

Command	**Status Register**		**Condition**
	D2	**D3**	
SCAN EQUAL	0	1	Disk = System
	1	0	Disk ≠ System
SCAN HIGH OR EQUAL	0	1	Disk = System
	0	0	Disk > System
	1	0	Disk < System
SCAN LOW OR EQUAL	0	1	Disk = System
	0	0	Disk < System
	1	0	Disk > System

9.6.11.8 VERIFY

The VERIFY command is used to read logical sectors containing a Normal Data Address Mark from the selected drive without transferring the data to the host. This command acts like a READ DATA command except that no data is transferred to the host. This command is designed for post-format or post write verification. Data is read from the disk, as the controller checks for valid Address Marks in the Address and Data Fields. The CRC is computed and checked against the previously stored value. Because no data is transferred to the host, the TC (Terminal Count of DMA) cannot be used to terminate this command. An implicit TC will be issued to the FDC by setting the EC bit. This implicit TC will occur when the SC value has been decrement to 0. This command can also be terminated by clearing the EC bit and when the EOT value equals to the final sector to be checked.

Table 9-18. VERIFY Command Result

MT	EC	SC/EOT	Termination Result
0	0	SC = DTL EOT ≤ # Sectors per side	No Errors
0	0	SC = DTL EOT > # Sectors per side	Abnormal Termination
0	1	SC ≤ # Sectors Remaining and EOT ≤ # Sectors per side	No Errors
0	1	SC > # Sectors Remaining or EOT > # Sectors per side	Abnormal Termination
1	0	SC = DTL EOT > # Sectors per side	No Errors
1	0	SC = DTL EOT > # Sectors per side	Abnormal Termination
1	1	SC ≤ # Sectors Remaining and EOT ≤ # Sectors per side	No Errors
1	1	SC > # Sectors Remaining or EOT > # Sectors per side	Abnormal Termination

9.6.12 Control Commands

The control commands do not transfer any data. Instead they are used to monitor and manage the data transfer. Three of the Control commands generate an interrupt when finished — READ ID, RE-CALIBRATE and SEEK. It is strongly recommended that a SENSE INTERRUPT STATUS command be issued after these commands to capture their valuable interrupt information. The RE-CALIBRATE, SEEK, and SPECIFY commands do not return any result bytes.

9.6.12.1 READ ID

The READ ID command is used to find the actual recording head position. It stores the first readable ID field value into the FDC registers. If the FDC cannot find an ID Address Mark by the time a second INDEX pulse is received, an abnormal termination will be generated by setting the IC code in the ST0 to 01.

9.6.12.2 CONFIGURE

The CONFIGURE command determines some special operation modes of the controller. It needs not be issued if the default values of the controller meet the system requirements.
EIS: Enable Implied Seeks. A Seek operation is performed before a READ, WRITE, SCAN, or VERIFY commands.
0 = Disable (default).
1 = Enable.
DFIFO: Disable FIFO.
0 = Enable.
1 = Disable (default).

POLLD: Disable polling of the drives.
0 = Enable (default). When enabled, a single interrupt is generated after a reset.
1 = Disable.
FIFOTHR: The FIFO threshold in the execution phase of data transfer commands. They are programmable from 00 to 0F hex (1 bytes to 16 bytes). Defaults to one byte.
PRETRK: The Pre-compensation Start Track Number. They are programmable from track 0 to FF hex (track 0 to track 255). Defaults to track 0.

9.6.12.3 RE-CALIBRATE

The RE-CALIBRATE command retracts the FDC read/write head to the track 0 position, resetting the value of the PCN counter and checking the TRK0# status. If TRK0# is low, the DIR# pin remains low and step pulses are issued. If TRK0# is high, SE [and EC bits] of ST0 are set high, and the command is terminated. When TRK0# remains low for 79 step pulses, the RE-CALIBRATE command is terminated by setting SE and EC bits of ST0 high. Consequently, for disks that can accommodate more than 80 tracks, more than one RE-CALIBRATE command is required to retract the head to the physical track 0.

The FDC is in a non-busy state during the Execution phase of this command, making it possible to issue another RE-CALIBRATE command in parallel with the current command.

On power-up, software must issue a RE-CALIBRATE command to properly initialize the FDC and its attached drives.

9.6.12.4 SEEK

The SEEK command controls the FDC read/write head movement from one track to another. The FDC compares the current head position, stored in PCN, with NCN values after each step pulse to determine what direction to move the head, if required. The direction of movement is determined below:

PCN < NCN — Step In: Set DIR# signal to 1 and issues step pulses
PCN > NCN — Step Out: Set DIR# signal to 0 and issues step pulses
PCN = NCN — Terminate the command by setting the ST0 SE bit to 1

The impulse rate of step pulse is controlled by Stepping Rate Time (SRT) bit in the SPECIFY command. The FDC is in a non-busy state during the Execution phase of this command, making it possible to issue another SEEK command in parallel with the current command.

9.6.12.5 RELATIVE SEEK

The RELATIVE SEEK command steps the selected drive in or out in a given number of steps. The DIR bit is used to determine to step in or out. RCN (Relative Cylinder Number) is used to determine how many tracks to step the head in or out from the current track. After the step operation is completed, the controller generates an interrupt, but the command has no Result phase. No other command except the SENSE INTERRUPT STATUS command should be issued while a RELATIVE SEEK command is in progress.

9.6.12.6 DUMPREG

The DUMPREG command is designed for system run-time diagnostics, and application software development, and debug. This command has one byte of Command phase and 10 bytes of Result phase, which return the values of parameters set in other commands.

9.6.12.7 LOCK

The LOCK command allows the programmer to fully control the FIFO parameters after a hardware reset. If the LOCK bit is set to 1, the parameters DFIFO, FIFOTHR, and PRETRK in the CONFIGURE command are not affected by a software reset. If the bit is set to 0, those parameters are set to default values after a software reset.

9.6.12.8 VERSION

The VERSION command is used to determine the controller being used. In Result phase, a value of 90 hex is returned in order to be compatible with the 82077.

9.6.12.9 SENSE INTERRUPT STATUS

The SENSE INTERRUPT STATUS command resets the interrupt signal (IRQ) generated by the FDC, and identifies the cause of the interrupt via the IC code and SE bit of the ST0, as shown in Table 9-19.

It may be necessary to generate an interrupt when any of the following conditions occur:

- Before any Data Transfer or READ ID command
- After SEEK or RE-CALIBRATE commands (no result phase exists)
- When a data transfer is required during an Execution phase in the non-DMA mode

Table 9-19. Interrupt Identification

SE	IC Code	Cause of Interrupt
0	11	Polling.
1	00	Normal termination of SEEK or RE-CALIBRATE command.
1	01	Abnormal termination of SEEK or RE-CALIBRATE command.

9.6.12.10 SENSE DRIVE STATUS

The SENSE DRIVE STATUS command acquires drive status information. It has no Execution phase.

9.6.12.11 SPECIFY

The SPECIFY command sets the initial values for the HUT (Head Unload Time), HLT (Head Load Time), SRT (Step Rate Time), and ND (Non-DMA mode) parameters. The possible values for HUT, SRT, and HLT are shown in Table 9-20, Table 9-21 and Table 9-22 respectively. The FDC is operated in DMA or non-DMA mode based on the value specified by the ND parameters.

Table 9-20. HUT Values (ms)

Parameter	1 Mbps	500 Kbps	300 Kbps	250 Kbps
0	128	256	426	512
1	8	16	26.7	32
…	…	…	…	…
E	112	224	373	448
F	120	240	400	480

Table 9-21. SRT Values (ms)

Parameter	1 Mbps	500 Kbps	300 Kbps	250 Kbps
0	8	16	26.7	32
1	7.5	15	25	30
…	…	…	…	…
E	1	2	3.33	4
F	0.5	1	1.67	2

Table 9-22. HLT Values

Parameter	1 Mbps	500 Kbps	300 Kbps	250 Kbps
00	128	256	426	512
01	1	2	3.33	4
02	2	4	6.7	8
…	…	…	…	…
7E	126	252	420	504
7F	127	254	423	508

9.6.12.12 PERPENDICULAR MODE

The PERPENDICULAR MODE command is used to support the unique READ/WRITE/FORMAT commands of Perpendicular Recording disk drives (4 Mbytes unformatted capacity). This command configures each of the four logical drives as a perpendicular or conventional disk drive via the DC3-DC0 bits or with the GAP and WG control bits. Perpendicular Recording drives operate in “Extra High Density” mode at 1 Mbps, and are downward compatible with 1.44 Mbyte and 720 Kbyte drives at 500 Kbps (High Density) and 250 Kbps (Double Density) respectively. This command should be issued during the initialization of the floppy controller. Then, when a drive is accessed for a FORMAT A TRACK or WRITE DATA command, the controller adjusts the format or Write Data parameters based on the data rate. If WG and GAP are used (not set to 00), the operation of the FDC is based on the values of GAP and WG. If WG and GAP are set to 00, setting DCn to 1 will set drive n to Perpendicular mode. DC3-DC0 are unaffected by a software reset, but WG and GAP are both cleared to 0 after a software reset.

Table 9-23. Effects of GAP and WG on FORMAT A TRACK and WRITE DATA Commands

GAP	WG	Mode	Length of GAP2 FORMAT FIELD	Portion of GAP2 Re-Written by WRITE DATA Command
0	0	Conventional	22 bytes	0 byte
0	1	Perpendicular (500 Kbps)	22 bytes	19 bytes
1	0	Reserved (Conventional)	22 bytes	0 byte
1	1	Perpendicular (1 Mbps)	41 bytes	38 bytes

Table 9-24. Effects of Drive Mode and Data Rate on FORMAT A TRACK and WRITE DATA Commands

Data Rate	Drive Mode	Length of GAP2 FORMAT FIELD	Portion of GAP2 Re-Written by WRITE DATA Command
250/300/500 Kbps	Conventional	22 bytes	0 byte
	Perpendicular	22 bytes	19 bytes
1 Mbps	Conventional	22 bytes	0 byte
	Perpendicular	41 bytes	38 bytes

9.6.12.13 INVALID

The INVALID command indicates when an undefined command has been sent to FDC. The FDC will set the bit 6 and the bit 7 in the Main Status Register to 1 and terminate the command without issuing an interrupt.

9.6.13 DMA Transfers

DMA transfers are enabled by the SPECIFY command and are initiated by the FDC by activating the LDRQ# cycle during a DATA TRANSFER command. The FIFO is enabled directly by asserting the LPC DMA cycles.

9.6.14 Low Power Mode

When writing a 1 to the bit 6 of the DSR, the controller is set to low power mode immediately. All the clock sources including Data Separator, Microcontroller, and Write precompensation unit will be gated. The FDC can be resumed from the low-power state in two ways. One is a software reset via the DOR or DSR; and the other is a read or write to either the Data Register or Main Status Register. The second method is more preferred since all internal register values are retained.

9.7 Serial Port (UART) Register Description

The IT8705F incorporates two enhanced serial ports that perform serial to parallel conversion on received data, and parallel to serial conversion on transmitted data. Each of the serial channels individually contains a programmable baud rate generator, which is capable of dividing the input clock by a number ranging from 1 to 65535. The data rate of each serial port can also be programmed from 115.2K baud down to 50 baud. The character options are programmable for 1 start bit; 1, 1.5 or 2 stop bits; even, odd, stick or no parity; and privileged interrupts.

Table 9-25. Serial Channel Registers

Register	DLAB*	Address	READ	WRITE
Data	0	Base + 0h	RBR (Receiver Buffer Register)	TBR (Transmitter Buffer Register)
Control	0	Base + 1h	IER (Interrupt Enable Register)	IER
	x	Base + 2h	IIR (Interrupt Identification Register)	FCR (FIFO Control Register)
	x	Base + 3h	LCR (Line Control Register)	LCR
	x	Base + 4h	MCR (Modem Control Register)	MCR
	1	Base + 0h	DLL (Divisor Latch LSB)	DLL
	1	Base + 1h	DLM (Divisor Latch MSB)	DLM
Status	x	Base + 5h	LSR (Line Status Register)	LSR
	x	Base + 6h	MSR (Modem Status Register)	MSR
	x	Base + 7h	SCR (Scratch Pad Register)	SCR

* DLAB is bit 7 of the Line Control Register.

9.7.1 Data Registers

The TBR and RBR individually holds from five to eight data bits. If the transmitted data is less than eight bits, it aligns to the LSB. Either received or transmitted data is buffered by a shift register, and is latched first by a holding register. The bit 0 of any word is first received and transmitted.

(1) Receiver Buffer Register (RBR) (Read only, Address offset=0, DLAB=0)

This register receives and holds the incoming data. It contains a non-accessible shift register which converts the incoming serial data stream into a parallel 8-bit word.

(2) Transmitter Buffer Register (TBR) (Write only, Address offset=0, DLAB=0)

This register holds and transmits the data via a non-accessible shift register, and converts the outgoing parallel data into a serial stream before transmission.

9.7.2 Control Registers: IER, IIR, FCR, DLL, DLM, LCR and MCR

(1) Interrupt Enable Register (IER) (Read/write, Address offset=1, DLAB=0)

The IER is used to enable (or disable) four active high interrupts which activate the interrupt outputs with its lower four bits: IER(0), IER(1), IER(2), and IER(3).

Table 9-26. Interrupt Enable Register Description

Bit	Default	Description
7-4	-	**Reserved**
3	0	**Enable MODEM Status Interrupt** Set this bit high to enable the Modem Status Interrupt when one of the Modem Status Registers changes its bit status.
2	0	**Enable Receiver Line Status Interrupt** Set this bit high to enable the Receiver Line Status Interrupt, which is caused when, Overrun, Parity, Framing or Break occurs.
1	0	**Enable Transmitter Holding Register Empty Interrupt** Set this bit high to enable the Receiver Line Status Interrupt which is caused when Overrun, Parity, Framing or Break errors occur.
0	0	**Enable Received Data Available Interrupt** Set this bit high to enable the Received Data Available Interrupt.

(2) Interrupt Identification Register (IIR) (Read only, Address offset=2)

This register facilitates the host CPU to determine interrupt priority and its source. The priority of four existing interrupt levels is listed below:

1. Received Line Status (highest priority)
2. Received Data Ready
3. Transmitter Holding Register Empty
4. Modem Status (lowest priority)

When a privileged interrupt is pending and the type of interrupt is stored in the IIR which is accessed by the host, the serial channel holds back all interrupts and indicates the pending interrupts with the highest priority to the host. Any new interrupts will not be acknowledged until the host access is completed. The contents of the IIR are described in the table on the next page:

Table 9-27. Interrupt Identification Register

FIFO Mode	Interrupt Identification Register			Interrupt Set and Reset Functions			
Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	**Priority Level**	**Interrupt Type**	**Interrupt Source**	**Interrupt Reset Control**
0	X	X	1	-	None	None	-
0	1	1	0	First	Receiver Line Status	OE, PE, FE, or BI	Read LSR
0	1	0	0	Second	Received Data Available	Received Data Available	Read RBR or FIFO drops below the trigger level
1	1	0	0	Second	Character Time-out Indication	No characters have been removed from or input to the RCVR FIFO during the last 4 character times and there is at least 1 character in it during this time	Read RBR
0	0	1	0	Third	Transmitter Holding Register Empty	Transmitter Holding Register Empty	Read IIR if THRE is the Interrupt Source Write THR
0	0	0	0	Fourth	Modem Status	CTS#, DSR#, RI#, DCD#	Read MSR

Note: X = Not defined.

IIR(0): Used to indicate a pending interrupt in either a hard-wired prioritized or polled environment with a logic 0 state. In such a case, IIR contents may be used as a pointer to the appropriate interrupt service routine.
IIR(1), IIR(2): Used to identify the highest priority interrupt pending.
IIR(3): In non-FIFO mode, this bit is a logic 0. In the FIFO mode, this bit is set along with bit 2 when a time-out interrupt is pending.
IIR(4), IIR(5): Always logic 0.
IIR(6), IIR(7): Are set when FCR(0) = 1.

(3) FIFO Control Register (FCR) (Write Only, Address offset=2)

This register is used to enable/clear the FIFO, and set the RCVR FIFO trigger level.

Table 9-28. FIFO Control Register Description

Bit	Default	Description
7-6	-	**Receiver Trigger Level Select** These bits set the trigger levels for the RCVR FIFO interrupt.
5-4	0	**Reserved**
3	0	This bit doesn't affect the Serial Channel operation. RXRDY and TXRDY functions are not available on this chip.
2	0	**Transmitter FIFO Reset** This self-clearing bit clears all contents of the XMIT FIFO and resets its related counter to 0 via a logic "1."
1	0	**Receiver FIFO Reset** Setting this self-clearing bit to logic 1 clears all contents of the RCVR FIFO and resets its related counter to 0 (except the shift register).
0	0	**FIFO Enable** XMIT and RCVR FIFOs are enabled when this bit is set high. XMIT and RCVR FIFOs are disabled and cleared respectively when this bit is cleared to low. This bit must be logic 1 if the other bits of the FCR are written, or they will not be properly programmed. When this register is switched to non-FIFO mode, all its contents are cleared.

Table 9-29. Receiver FIFO Trigger Level Encoding

FCR (7)	FCR (6)	RCVR FIFO Trigger Level
0	0	1 byte
0	1	4 bytes
1	0	8 bytes
1	1	14 bytes

(4) Divisor Latches (DLL, DLM) (Read/write, Address offset=0,1 DLAB=1)

Two 8-bit Divisor Latches (DLL and DLM) store the divisor values in a 16-bit binary format. They are loaded during the initialization to generate a desired baud rate.

(5) Baud Rate Generator (BRG)

Each serial channel contains a programmable BRG which can take any clock input (from DC to 8 MHz) to generate standard ANSI/CCITT bit rates for the channel clocking with an external clock oscillator. The DLL or DLM is a number of 16-bit format, providing the divisor range from 1 to 2^{16} to obtain the desired baud rate. The output frequency is 16X data rate.

Table 9-30. Baud Rates Using (24 MHz ¸ 13) Clock

Desired Baud Rate	Divisor Used
50	2304
75	1536
110	1047
134.5	857
150	768
300	384
600	192
1200	96
1800	64
2000	58
2400	48
3600	32
4800	24
7200	16
9600	12
19200	6
38400	3
57600	2
115200	1

(6) Scratch Pad Register (Read/write, Address offset=7)

This 8-bit register does not control the UART operation in any way. It is intended as a scratch pad register to be used by programmers to temporarily hold general purpose data.

(7) Line Control Register (LCR) (Read/write, Address offset=3)

LCR controls the format of the data character and supplies the information of the serial line. Its contents are described on the next page:

Table 9-31. Line Control Register Description

Bit	Default	Description
7	0	**Divisor Latch Access Bit (DLAB)** Must be set to high to access the Divisor Latches of the baud rate generator during READ or WRITE operations. It must be set low to access the Data Registers (RBR and TBR) or the Interrupt Enable Register.
6	0	**Set Break** Force the Serial Output (SOUT) to the spacing state (logic 0) by a logic 1, and this state will be preserved until a low level resetting LCR(6), enabling the serial port to alert the terminal in a communication system.
5	0	**Stick Parity** When this bit and LCR(3) are high at the same time, the parity bit is transmitted, and then detected by receiver, in opposite state by LCR(4) to force the parity bit into a known state and to check the parity bit in a known state.
4	0	**Even Parity Select** When parity is enabled (LCR(3) = 1), LCR(4) = 0 selects odd parity, and LCR(4) = 1 selects even parity.
3	0	**Parity Enable** A parity bit, located between the last data word bit and stop bit, will be generated or checked (transmit or receive data) when LCR(3) is high.
2	0	**Number of Stop Bits** This bit specifies the number of stop bits in each serial character, as summarized in Table 9-32.
1-0	00	**Word Length Select [1:0]** 11: 8 bits 10: 7 bits 01: 6 bits 00: 5 bits

Table 9-32. Stop Bits Number Encoding

LCR (2)	Word Length	No. of Stop Bits
0	-	1
1	5 bits	1.5
1	6 bits	2
1	7 bits	2
1	8 bits	2

Note: The receiver will ignore all stop bits beyond the first, regardless of the number used in transmission.

(8) MODEM Control Register (MCR) (Read/write, Address offset=4)

Control the interface by the modem or data set (or device emulating a modem).

Table 9-33. Modem Control Register Description

Bit	Default	Description
7-5	-	**Reserved**
4	0	**Internal Loop Back** Provide a loopback feature for diagnostic test of the serial channel when it is set high. Serial Output (SOUT) is set to the Marking State Shift Register output loops back into the Receiver Shift Register. All Modem Control inputs (CTS#, DSR#, RI# and DCD#) are disconnected. The four Modem Control outputs (DTR#, RTS#, OUT1 and OUT2) are internally connected to the four Modem Control inputs and forced to inactive high. The transmitted data are then immediately received, allowing the processor to verify the transmit and receive data path of the serial channel.
3	0	**Out2** This bit enables the serial port interrupt output by logic 1.
2	0	**Out1** This bit does not have an output pin and can only be read or written by the CPU.
1	0	**Request to Send (RTS)** Control the Request to Send (RTS#) which is in an inverse logic state with that of MCR(1).
0	0	**Data Terminal Ready (DTR)** Control the Data Terminal ready (DTR#) which is in an inverse logic state with the MCR(0).

9.7.3 Status Registers: LSR and MSR

(1) Line Status Register (LSR) (Read/write, Address offset=5)

This register provides status indications and is usually the first register read by the CPU to determine the cause of an interrupt or to poll the status of each serial channel. The contents of the LSR are described below:

Table 9-34. Line Status Register Description

Bit	Default	Description
7	0	**Error in Receiver FIFO** In 16450 mode, this bit is always 0. In the FIFO mode, it sets high when there is at least one parity error, framing or break interrupt in the FIFO. This bit is cleared when the CPU reads the LSR, if there are no subsequent errors in the FIFO.
6	1	**Transmitter Empty** This read only bit indicates that the Transmitter Holding Register and Transmitter Shift Register are both empty. Otherwise, this bit is "0," and has the same function in the FIFO mode.
5	1	**Transmitter Holding Register Empty** Transmitter Holding Register Empty (THRE). This read only bit indicates that the TBR is empty and is ready to accept a new character for transmission. It is set high when a character is transferred from the THR into the Transmitter Shift Register, causing a priority 3 IIR interrupt which is cleared by a read of IIR. In the FIFO mode, it is set when the XMIT FIFO is empty, and cleared when at least one byte is written to the XMIT FIFO.
4	0	**Line Break** Break Interrupt (BI) status bit indicates that the last character received was a break character, (invalid but entire character), including parity and stop bits. This occurs when the received data input is held in the spacing (logic 0) for longer than a full word transmission time (start bit + data bits + parity + stop bit). When any of these error conditions is detected (LSR(1) to LSR(4)), a Receiver Line Status interrupt (priority 1) will be generated in the IIR with the IER(2) enabled previously.
3	0	**Framing Error** Framing Error (FE) bit, a logic 1, indicates that the stop bit in the received character was not valid. It resets low when the CPU reads the contents of the LSR.
2	0	**Parity Error** The parity error (PE) indicates by a logic 1 that the received data character does not have the correct even or odd parity, as selected by LCR(4). It will be reset to "0" whenever the LSR is read by the CPU.
1	0	**Overrun Error** Overrun Error (OE) bit indicates by a logic 1 that the RBR has been overwritten by the next character before it had been read by the CPU. In the FIFO mode, the OE occurs when the FIFO is full and the next character has been completely received by the Shift Register. It will be reset when the LSR is read by the CPU.
0	0	**Data Ready** A logic "1" indicates a character has been received by the RBR. And a logic "0" indicates all the data in the RBR or the RCVR FIFO have been read.

(2) MODEM Status Register (MSR) (Read/write, Address offset=6)

This 8-bit register indicates the current state of the control lines with modems or peripheral devices in addition to this current state information. Four of these eight bits MSR(4) - MSR(7) can provide the state change information when a modem control input changes state. It is reset low when the host reads the MSR.

Table 9-35. Modem Status Register Description

Bit	Default	Description
7	0	**Data Carrier Detect** Receive Line Signal Detect - Indicates the complement status of Receive Line Signal Detect (DCD#) input. If MCR(4) = 1, MSR(7) is equivalent to OUT2 of the MCR.
6	0	**Ring Indicator** Ring Indicator (RI#) - Indicates the complement status to the RI# input. If MCR(4)=1, MSR(6) is equivalent to OUT1 in the MCR.
5	0	**Data Set Ready** Data Set Ready (DSR#) - Indicates that the modem is ready to provide received data to the serial channel receiver circuitry. If the serial channel is in the loop mode (MCR(4) = 1), MSR(5) is equivalent to DTR# in the MCR.
4	0	**Clear to Send** Clear to Send (CTS#) – Indicates the complement of CTS# input. When the serial channel is in the loop mode (MCR(4)=1), MSR(5) is equivalent to RTS# in the MCR.
3	0	**Delta Data Carrier Detect** Indicate that the DCD# input state has been changed since the last time read by the host.
2	0	**Trailing Edge Ring Indicator** Indicate that the RI input state to the serial channel has been changed from a low to high since the last time read by the host. The change to logic 1 does not activate the TERI.
1	0	**Delta Data Set Ready** Delta Data Set Ready (DDSR) - A logic "1" indicates that the DSR# input state to the serial channel has been changed since the last time read by the host.
0	0	**Delta Clear to Send** This bit indicates the CTS# input to the chip has changed state since the last time the MSR was read.

9.7.4 Reset

Reset of the IT8705F should be held to an idle mode reset high for 500 ns until initialization, and this causes the following:

Initialization of the transmitter and receiver internal clock counters.

Table 9-36. Reset Control of Registers and Pinout Signals

Register/Signal	Reset Control	Reset Status
Interrupt Enable Register	Reset	All bits Low
Interrupt Identification Register	Reset	Bit 0 is high and bits 1-7 are low
FIFO Control Register	Reset	All bits Low
Line Control Register	Reset	All bits Low
Modem Control Register	Reset	All bits Low
Line Status Register	Reset	Bits 5 and 6 are high, others are low
Modem Status Register	Reset	Bits 0-3 low, bits 4-7 input signals
SOUT0, SOUT1	Reset	High
RTS0#, RTS1#, DTR0#, DTR1#	Reset	High
IRQ of Serial Port	Reset	High Impedance

9.7.5 Programming

Each serial channel of the IT8705F is programmed by control registers, whose contents define the character length, number of stop bits, parity, baud rate and modem interface. Even though the control register can be written in any given order, the IER should be the last register written because it controls the interrupt enables. After the port is programmed, these registers can still be updated whenever the port is not transferring data.

9.7.6 Software Reset

This approach allows the serial port returning to a completely known state without a system reset. This is achieved by writing the required data to the LCR, DLL, DLM and MCR. The LSR and RBR must be read before enabling interrupts to clear out any residual data or status bits that may be invalid for subsequent operations.

9.7.7 Clock Input Operation

The input frequency of the Serial Channel is 24 MHz ÷ 13, not exactly 1.8432 MHz.

9.7.8 FIFO Interrupt Mode Operation

(1) RCVR Interrupt

When setting FCR(0)=1 and IER(0)=1, the RCVR FIFO and receiver interrupts are enabled. The RCVR interrupt occurs under the following conditions:

a. The receive data available interrupt, the IIR, and receive data available indication will be issued only if the FIFO has reached its programmed trigger level. They will be cleared as soon as the FIFO drops below its trigger level

b. The receiver line status interrupt has higher priority over the received data available interrupt

c. The time-out timer will be reset after receiving a new character or after the host reads the RCVR FIFO whenever a time-out interrupt occurs. The timer will be reset when the host reads one character from the RCVR FIFO

RCVR FIFO time-out Interrupt: By enabling the RCVR FIFO and receiver interrupts, the RCVR FIFO time-out interrupt will occur under the following conditions:

a. The RCVR FIFO time-out interrupt will occur only if there is at least one character in the FIFO whenever the interval between the most recent received serial character and the most recent host read from the FIFO is longer than four consecutive character times

b. The RLCK clock signal input is used to calculate character times

c. The time-out timer will be reset after receiving a new character or after the host reads the RCVR FIFO whenever a time-out interrupt occurs. The timer will be reset when the host reads one character from the RCVR FIFO

(2) XMIT Interrupt

By setting FCR(0) and IER(1) to high, the XMIT FIFO and transmitter interrupts are enabled, and the XMIT interrupt occurs under the conditions described below:

a. The transmitter interrupt occurs when the XMIT FIFO is empty, and it will be reset if the Transmitter Holding Register (THR) is written or the IIR is read

b. The transmitter FIFO empty indications will be delayed one character time minus the last stop bit time whenever the following condition occurs: THRE = 1 and there have not been at least two bytes in the transmitter FIFO at the same time since the last THRE = 1. The transmitter interrupt after changing FCR(0) will be immediate, if it is enabled. Once the first transmitter interrupt is enabled, the THRE indication is delayed one character time minus the last stop bit time

The character time-out and RCVR FIFO trigger level interrupts are in the same priority order as the received data available interrupt. The XMIT FIFO empty is in the same priority as the transmitter holding register empty interrupt.

FIFO Polled Mode Operation [FCR (0)=1, and IER(0), IER(1), IER(2), IER(3) or all are “0”]

Either or both XMIT and RCVR can be in this operation mode. The operation mode can be programmed by users and is responsible for checking the RCVR and XMIT status via the LSR described below:

LSR(7): RCVR FIFO error indication.

LSR(6): XMIT FIFO and Shift register empty.

LSR(5): XMIT FIFO empty indication.

LSR(4) - LSR(1): Specify that errors have occurred. Character error status is handled in the same way as in the interrupt mode. The IIR is not affected since IER(2)=0.

LSR(0): This bit is high whenever the RCVR FIFO contains at least one byte.

No trigger level is reached or time-out condition indicated in the FIFO Polled mode.

9.8 Parallel Port

The IT8705F incorporates one multi-mode high performance parallel port, which supports the IBM AT, PS/2 compatible bi-directional Standard Parallel Port (SPP), the Enhanced Parallel Port (EPP) and the Extended Capabilities Port (ECP). Please refer to the IT8705F Configuration registers and Configuration Description for information on enabling/disabling, changing the base address of the parallel port, and operation mode selection.

Table 9-37. Parallel Port Connector in Different Modes

Host Connector	Pin No.	SPP	EPP	ECP
1	76	STB#	WRITE#	NStrobe
2-9	71-68, 66-63	PD0 - 7	PD0 - 7	PD0 – 7
10	62	ACK#	INTR	nAck
11	61	BUSY	WAIT#	Busy PeriphAck(2)
12	60	PE	(NU) (1)	PError nAckReverse(2)
13	59	SLCT	(NU) (1)	Select
14	77	AFD#	DSTB#	nAutoFd HostAck(2)
15	75	ERR#	(NU) (1)	nFault nPeriphRequest(2)
16	73	INIT#	(NU) (1)	nInit nReverseRequest(2)
17	74	SLIN#	ASTB#	nSelectIn

Notes: 1. NU: Not used.
2. Fast mode.
3. For more information, please refer to the IEEE 1284 standard.

9.8.1 SPP and EPP Modes

Table 9-38. Address Map and Bit Map for SPP and EPP Modes

Register	Address	R/W	D0	D1	D2	D3	D4	D5	D6	D7	Mode
Data Port	Base 1+0H	R/W	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	SPP/EPP
Status Port	Base 1+1H	RO	TMOUT	1	1	ERR#	SLCT	PE	ACK#	BUSY#	SPP/EPP
Control Port	Base 1+2H	R/W	STB	AFD	INIT	SLIN	IRQE	PDDIR	1	1	SPP/EPP
EPP Address Port	Base 1+3H	R/W	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	EPP
EPP Data Port0	Base 1+4H	R/W	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	EPP
EPP Data Port1	Base 1+5H	R/W	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	EPP
EPP Data Port2	Base 1+6H	R/W	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	EPP
EPP Data Port3	Base 1+7H	R/W	PD0	PD1	PD2	PD3	PD4	PD5	PD6	PD7	EPP

Note 1. The Base address 1 depends on the Logical Device configuration registers of Parallel Port (0X60, 0X61).

(1)Data Port (Base Address 1 + 00h)

This is a bi-directional 8-bit data port. The direction of data flow is determined by the bit 5 of the logic state of the control port register. It forwards directions when the bit is low and reverses directions when the bit is high.

(2)Status Port (Base Address 1 + 01h)

This is a **read only** register. Writing to this register has no effects. The contents of this register are latched during an IOR cycle.
Bit 7 - BUSY#: Inverse of printer BUSY signal. A logic "0" means that the printer is busy and cannot accept another character. A logic "1" means that it is ready to accept the next character.
Bit 6 - ACK#: Printer acknowledge, a logic "0" means that the printer has received a character and is ready to accept another. A logic "1" means that it is still processing the last character.
Bit 5 - PE: Paper end, a logic "1" indicates the paper end.
Bit 4 - SLCT: Printer selected, a logic "1" means that the printer is on line.
Bit 3 - ERR#: Printer error signal, a logic "0" means an error has been detected.
Bits 1, 2: Reserved, these bits are always "1" when read.
Bit 0 - TMOUT: This bit is valid only in EPP mode and indicates that a 10-msec time-out has occurred in EPP operation. A logic "0" means no time-out occurred and a logic “1” means that a time-out error has been detected. This bit is cleared by a LRESET# or by writing a logic “1” to it. When the IT8705F is selected to non-EPP mode (SPP or ECP), this bit is always logic "1" when read.

(3)Control Port (Base Address 1 + 02h)

This port provides all output signals to control the printer. The register can be read and written.
Bits 6, 7: Reserved, these two bits are always "1" when read.
Bit 5 - PDDIR: Data port direction control. This bit determines the direction of the data port. Set this bit "0" to output the data port to PD bus and "1" to input from PD bus.
Bit - 4 IRQE: Interrupt request enable. Setting this bit "1" enables the interrupt requests from the parallel port to the host. An interrupt request is generated by a "0" to "1" transition of the ACK# signal.
Bit - 3 SLIN: Inverse of SLIN# pin, setting this bit to "1" selects the printer.
Bit - 2 INIT: Initiate printer, setting this bit to "0" initializes the printer.
Bit - 1 AFD: Inverse of the AFD# pin, setting this bit to "1" causes the printer to automatically advance one line after each line is printed.
Bit 0 - STB: Inverse of the STB# pin. This pin controls the data strobe signal to the printer.

(4)EPP Address Port (Base Address 1 + 03h)

The EPP Address Port is only available in the EPP mode. When the host writes to this port, the contents of D0 -D7 are buffered and output to PD0 - PD7. The leading edge of IOW (Internal signal, active when LPC I/O write cycle is on this address) causes an EPP ADDRESS WRITE cycle. When the host reads from this port, the contents of PD0 - PD7 are read. The leading edge of IOR (Internal signal, active when LPC I/O read cycle is on this address) causes an EPP ADDRESS read cycle.

(5)EPP Data Ports 0-3 (Base Address 1 + 04h - 07h)

The EPP Data Ports are only available in the EPP mode. When the host writes to these ports, the contents of D0 - D7 are buffered and output to PD0 - PD7. The leading edge of IOW (Internal signal, active when LPC I/O WRITE cycle is on this address) causes an EPP DATA WRITE cycle. When the host reads from these ports, the contents of PD0 - PD7 are read. The leading edge of IOR (Internal signal, active when LPC I/O read cycle is on this address) causes an EPP DATA read cycle.

9.8.2 EPP Mode Operation

When the parallel port of the IT8705F is set in the EPP mode, the SPP mode is also available. If no EPP Address/Data Port address is decoded (Base address + 03h- 07h), the PD bus is in the SPP mode, and the output signals such as STB#, AFD#, INIT#, and SLIN# are set by the SPP control port. The direction of the data port is controlled by the bit 5 of the control port register. There is a 10-msec time required to prevent the system from lockup. The time has elapsed from the beginning of the IOCHRDY (Internal signal: When active, the IT8705F will issue Long Wait in SYNC field) high (EPP read/write cycle) to WAIT# being de-asserted. If a time-out occurs, the current EPP read/write cycle is aborted and a logic "1" will be read in the bit 0 of the status port register. The host must write 0 to bits 0, 1, 3 of the control port register before any EPP read/write cycle (EPP spec.) The pins STB#, AFD# and SLIN# are controlled by hardware for the hardware handshaking during EPP read/write cycle.

(1) EPP ADDRESS WRITE

1. The host writes a byte to the EPP Address Port (Base address + 03h). The chip drives D0 - D7 onto PD0 - PD7.
2. The chip asserts WRITE# (STB#) and ASTB# (SLIN#) after IOW becomes active.
3. The peripheral de-asserts WAIT#, indicating that the chip may begin the termination of this cycle. Then, the chip de-asserts ASTB#, latches the address from D0 - D7 to PD bus, allowing the host to complete the I/O write cycle.
4. The peripheral asserts WAIT#, indicating that it acknowledges the termination of the cycle. Then, the chip de-asserts WRITE to terminate the cycle.

(2) EPP ADDRESS READ

1. The host reads a byte from the EPP Address Port. The chip drives PD bus to tri-state for the peripheral to drive.
2. The chip asserts ASTB# after IOR becomes active.
3. The peripheral drives the PD bus valid and de-asserts WAIT, indicating that the chip may begin the termination of this cycle. Then, the chip de-asserts ASTB#, latches the address from PD bus to D0 -D7, allowing the host to complete the I/O read cycle.
4. The peripheral drives the PD bus to tri-state and then asserts WAIT#, indicating that it acknowledges the termination of the cycle.

(3) EPP DATA WRITE

1. The host writes a byte to the EPP Data Port (Base address +04H - 07H). The chip drives D0- D7 onto PD0 -PD7.
2. The chip asserts WRITE# (STB#) and DSTB# (AFD#) after IOW becomes active.
3. The peripheral de-asserts WAIT#, indicating that the chip may begin the termination of this cycle. Then, the chip de-asserts DSTB#, latches the data from D0 - D7 to the PD bus, allowing the host to complete the I/O write cycle.
4. The peripheral asserts WAIT#, indicating that it acknowledges the termination of the cycle. Then, the chip de-asserts writes to terminate the cycle.

(4) EPP DATA READ

1. The host reads a byte from the EPP DATA Port. The chip drives PD bus to tri-state for the peripheral to drive.
2. The chip asserts DSTB# after IOR becomes active.
3. The peripheral drives PD bus valid and de-asserts WAIT#, indicating that the chip may begin the termination of this cycle. Then, the chip de-asserts DSTB#, latches the data from PD bus to D0 - D7, allowing the host to complete the I/O read cycle.
4. The peripheral tri-states the PD bus and then asserts WAIT#, indicating that it acknowledges the termination of the cycle.

9.8.3 ECP Mode Operation

This mode is both software and hardware compatible with the existing parallel ports, allowing ECP to be used as a standard LPT port when the ECP mode is not required. It provides an automatic high-burst-bandwidth channel that supports DMA or the ECP mode in both forward and reverse directions. A 16-byte FIFO is implemented in both forward and reverse directions to smooth data flow and increase the maximum bandwidth allowed. The port supports automatic handshaking for the standard parallel port to improve compatibility and expedite the mode transfer. It also supports run-length encoded (RLE) decompression in hardware. Compression is accomplished by counting identical bytes and transmitting an RLE byte that indicates how many times a byte has been repeated. The IT8705F does not support hardware RLE compression. For a detailed description, please refer to "Extended Capabilities Port Protocol and ISA Interface Standard".

Table 9-39. Bit Map of the ECP Registers

Register	D7	D6	D5	D4	D3	D2	D1	D0
Data	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
ecpAFifo	Addr/RLE	Address or RLE field						
dsr	nBusy	nAck	PError	Select	NFault	1	1	1
dcr	1	1	PDDIR	IRQE	SelectIn	nInit	AutoFd	Strobe
cFifo	Parallel Port Data FIFO							
ecpDFifo	ECP Data FIFO							
tFifo	Test FIFO							
cnfgA	0	0	0	1	0	0	0	0
cnfgB	0	intrValue	0	0	0	0	0	0
ecr	mode			nErrIntrEn	dmaEn	ServiceIntr	full	empty

(1) ECP Register Definitions

Table 9-40. ECP Register Definitions

Name	Address	R/W	ECP Mode	Function
data	Base 1 +000H	R/W	000-001	Data Register
ecpAFifo	Base 1 +000H	R/W	011	ECP FIFO (Address)
dsr	Base 1 +001H	R/W	All	Status Register
dcr	Base 1 +002H	R/W	All	Control Register
cFifo	Base 2 +000H	R/W	010	Parallel Port Data FIFO
ecpDFifo	Base 2 +000H	R/W	011	ECP FIFO (DATA)
tFifo	Base 2 +000H	R/W	110	Test FIFO
cnfgA	Base 2 +000H	RO	111	Configuration Register A
cnfgB	Base 2 +001H	R/W	111	Configuration Register B
ecr	Base 2 +002H	R/W	All	Extended Control Register

Notes:
1. The Base address 1 depends on the Logical Device configuration registers of Parallel Port (0X60, 0X61).
2. The Base address 2 depends on the Logical Device configuration registers of Parallel Port (0X62, 0X63).

(2) ECP Mode Descriptions

Table 9-41. ECP Mode Descriptions

Mode	Description
000	Standard Parallel Port Mode
001	PS/2 Parallel Port Mode
010	Parallel Port FIFO Mode
011	ECP Parallel Port Mode
110	Test Mode
111	Configuration Mode

Note: Please refer to the ECP Register Description on pages 125-126 for a detailed description of the mode selection.

(3) ECP Pin Descriptions

Table 9-42. ECP Pin Descriptions

Pin	Symbol	Type	Description
76	**nStrobe (HostClk)**	O	Used for handshaking with Busy to write data and addresses into the peripheral device.
71-68, 66-63	**PD0-PD7**	I/O	Address or data or RLE data.
62	**nAck (PeriphClk)**	I	Used for handshaking with nAutoFd to transfer data from the peripheral device to the host.
61	**Busy (PeriphACK)**	I	The peripheral uses this signal for flow control in the forward direction (hand-shaking with nStrobe). In the reverse direction, this signal is used to determine whether a command or data information is present on PD0-PD7.
60	**PError (nAckReverse)**	I	Used to acknowledge nInit from the peripheral which drives this signal low, allowing the host to drive the PD bus.
59	**Select**	I	Printer On-Line indication.
77	**nAutoFd (HostAck)**	O	In the reverse direction, it is used for handshaking between the nAcK and the host. When it is asserted, a peripheral data byte is requested. In the forward direction, this signal is used to determine whether a command or data information is present on PD0 - PD7.
75	**nFault (nPeriphRequest)**	I	In the forward direction (only), the peripheral is allowed (but not required) to assert this signal (low) to request a reverse transfer while in ECP mode. The signal provides a mechanism for peer-to-peer communication. It is typically used to generate an interrupt to the host, which has the ultimate control over the transfer direction.
73	**nInit (nReverseRequest)**	O	The host may drive this signal low to place the PD bus in the reverse direction. In the ECP mode, the peripheral is permitted to drive the PD bus when nInit is low, and nSelectIn is high.
74	**NSelectIn (1284 Active)**	O	Always inactive (high) in the ECP mode.

(4) Data Port (Base 1+00h, Modes 000 and 001)

Its contents will be cleared by a RESET. In a write operation, the contents of the LPC data fields are latched by the Data Register. The contents are then sent without being inverted to PD0~ PD7. In a read operation, the contents of data ports are read and sent to the host.

(5) ecpAFifo Port (Address/RLE) (Base 1 +00h, Mode 011)

Any data byte written to this port are placed in the FIFO and tagged as an ECP Address/RLE. The hardware then automatically sends this data to the peripheral. Operation of this port is valid only in the forward direction (dcr(5)=0).

(6) Device Status Register (dsr) (Base 1 +01h, Mode All)

Bits 0, 1 and 2 of this register are not implemented. These bit states are remained at high in a read operation of the Printer Status Register.
dsr(7): This bit is the inverted level of the Busy input.
dsr(6): This bit is the state of the nAck input.
dsr(5): This bit is the state of the PError input.
dsr(4): This bit is the state of the Select input.
dsr(3): This bit is the state of the nFault input.
dsr(2)-dsr(0): These bits are always 1.

(7) Device Control Register (dcr) (Base 1+02h, Mode All)

Bits 6 and 7 of this register have no function. They are set high during the read operation, and cannot be written. Contents in bits 0-5 are initialized to "0" when the RESET pin is active.
dcr(7)-dcr(6): These two bits are always high.
dcr(5): Except in the modes 000 and 010, setting this bit low means that the PD bus is in output operation; setting it high, in input operation. This bit will be forced to low in mode 000.
dcr(4): Setting this bit high enables the interrupt request from the peripheral to the host due to a rising edge of the nAck input.
dcr(3): It is inverted and output to SelectIn.
dcr(2): It is output to nInit without inversion.
dcr(1): It is inverted and output to nAutoFd.
dcr(0): It is inverted and output to nStrobe.

(8) Parallel Port Data FIFO (cFifo) (Base 2+00h, Mode 010)

Bytes written or DMA transferred from the host to this FIFO are sent by a hardware handshaking to the peripheral according to the standard parallel port protocol. This operation is only defined for the forward direction.

(9) ECP Data FIFO (ecpDFifo) (Base 2+00h, Mode 011)

When the direction bit dcr(5) is 0, bytes written or DMA transferred from the host to this FIFO are sent by hardware handshaking to the peripheral according to the ECP parallel port protocol. When dcr(5) is 1, data bytes from the peripheral to this FIFO are read in an automatic hardware handshaking. The host can receive these bytes by performing read operations or DMA transfers from this FIFO.

(10) Test FIFO (tFifo) (Base 2+00h, Mode 110)

The host may operate read/write or DMA transfers to this FIFO in any direction. Data in this FIFO will be displayed on the PD bus without using hardware protocol handshaking. The tFifo will not accept new data after it is full. Making a read operation from an empty tFifo causes the last data byte to return.

(11) Configuration Register A (cnfgA) (Base 2+00h, Mode 111)

This **read only** register indicates to the system that interrupts are ISA-Pulses compatible. This is an 8-bit implementation by returning a 10h.

(12) Configuration Register B (cnfgB) (Base 2+01h, Mode 111)

This register is **read only**.
cnfgB(7): A logic "0" read indicates that the chip does not support hardware RLE compression.
cnfgB(6): Reserved
cnfgB(5)-cnfg(3): A value 000 read indicates that the interrupt must be selected with jumpers.
cnfgB(2)-cnfg(0): A value 000 read indicates that the DMA channel is jumpered 8-bit DMA.

(13) Extended Control Register (ecr) (Base 2+02h, Mode All)

ECP function control register.
ecr(7)-ecr(5): These bits are used for read/write and mode selection.

Table 9-43. Extended Control Register (ECR) Mode and Description

ECR	Mode and Description
000	**Standard Parallel Port Mode** The FIFO is reset and the direction bit dcr(5) is always 0 (forward direction) in this mode.
001	**PS/2 Parallel Port Mode** It is similar to the SPP mode, except that the dcr(5) is **read/write**. When dcr(5) is 1, the PD bus is tri-state. Reading the data port returns the value on the PD bus instead of the value of the data register.
010	**Parallel Port Data FIFO Mode** This mode is similar to the 000 mode, except that the host writes or DMA transfers the data bytes to the FIFO. The FIFO data is then transmitted to the peripheral using the standard parallel port protocol automatically. This mode is only valid in the forward direction (dcr(5)=0).
011	**ECP Parallel Port Mode** In the forward direction, bytes placed into the ecpDFifo and ecpAFifo are placed in a single FIFO and automatically transmitted to the peripheral under the ECP protocol. In the reverse direction, bytes are transmitted to the ecpDFifo from the ECP port.
100, 101	**Reserved** Not defined.
110	**Test mode** In this mode, the FIFO data may be read or written, but it cannot be sent to the peripheral.
111	**Configuration mode** In this mode, the cnfgA and cnfgB registers are accessible at 0x400 and 0x401.

ecr(4): nErrIntrEn, read/write, Valid in ECP(011) Mode
1: Disable the interrupt generated on the asserting edge of the nFault input.
0: Enable the interrupt pulse on the asserting edge of the nFault. An interrupt pulse will be generated if nFault is asserted, or if this bit is written from 1 to 0 in the low-level nFault.

ecr(3): dmaEn, read/write
1: Enable DMA. DMA starts when serviceIntr (ecr(2)) is 0.
0: Disable DMA unconditionally.

ecr(2): ServiceIntr, read/write
1: Disable DMA and all service interrupts.
0: Enable the service interrupts. This bit will be set to "1" by hardware when one of the three service interrupts has occurred.
Writing "1" to this bit will not generate an interrupt.

Case 1: dmaEn=1
During DMA, this bit is set to 1 (a service interrupt generated) if the terminal count is reached.

Case 2: dmaEn=0, dcr(5)=0
This bit is set to 1 (a service interrupt generated) whenever there are writeIntrThreshold or more bytes space free in the FIFO.

Case 3: dmaEn=0, dcr(5)=1

This bit is set to 1 (a service interrupt generated) whenever there are readIntrThreshold or more valid bytes to be read from the FIFO.

ecr(1): full, **read only**.
1: The FIFO is full and cannot accept another byte.
0: The FIFO has at least 1 free data byte space.
ecr(0): empty, **read only**.
1: The FIFO is empty.
0: The FIFO contains at least 1 data byte.

(14) Mode Switching Operation

In programmed I/O control (mode 000 or 001), P1284 negotiation and all other tasks that happen before data is transferred are software-controlled. Setting mode to 011 or 010 will cause the hardware to perform an automatic control-line handshaking, transferring information between the FIFO and the ECP port.

From the mode 000 or 001, it may be immediately switched to any other mode. To change directions, the mode must first be set to 001.

In the extended forward mode, the FIFO must be cleared and all the signals must be de-asserted before returning to mode 000 or 001. In ECP reverse mode, all data must be read from the FIFO before returning to mode 000 or 001. Usually, unneeded data is accumulated during ECP reverse handshaking, when the mode is changed during a data transfer. In such conditions, nAutoFd will be de-asserted regardless of the transfer state. To avoid bugs that may occur while handshaking signals are being processed, these guidelines must be followed.

(15) Software Operation (ECP)

Before the ECP operation can begin, it is first necessary for the host to switch the mode to 000 in order to negotiate with the parallel port. During this process, the host determines whether the peripheral supports the ECP protocol.

After this negotiation is completed, the mode is set to 011 (ECP). To enable the drivers, the direction must be set to 0. Both strobe and autoFd are set to 0, causing nStrobe and nAutoFd signals to be de-asserted.

All FIFO data transfers are PWord wide and PWord aligned. Permitted only in the forward direction, Address/RLE transfers are byte-wide. The ECP Address/RLE bytes may be automatically transmitted by writing to the ecpAFifo. Similarly, data PWords may be automatically sent via the ecpDFifo.

To change directions, the host switches mode to 001. It then negotiates either the forward or reverse channel, sets direction to 1 or 0, and finally switches mode to 001. If the direction is set to 1, the hardware performs handshaking for each ECP data byte read, then tries to fill the FIFO. At this time, PWords may be read from the ecpDFifo while it retains data. It is also possible to perform the ECP transfers by handshaking with individual bytes under programmed control in mode = 001, or 000, even though this is a comparatively time-consuming approach.

(16) Hardware Operation (DMA)

The Standard PC DMA protocol (through LDRQ#) is followed. As in the programmed I/O case, software sets the direction and state. Next, the desired count and memory addresses are programmed into DMA controller. The dmaEn is set to 1, and the serviceIntr is set to 0. To complete the process, the DMA channel with the DMA controller is unmasked. The contents in the FIFO are emptied or filled by DMA using the right mode and direction.

DMA is always transferred to or from the FIFO located at 0 x 400. By generating an interrupt and asserting a serviceIntr. DMA is disabled when the DMA controller reaches the terminal count. By not asserting LDRQ# for more than 32 consecutive DMA cycles, blocking of refresh requests is eliminated.

When it is necessary to disable a DMA while performing a data transfer, the host DMA controller is disabled, serviceIntr is then set to 1, and dmaEn is next set to 0. If the contents in FIFO are empty or full, the DMA will start again. This is first done by enabling the host DMA controller, and then setting dmaEn to 1. Finally, serviceIntr is set to 0. Upon the completion of a DMA transfer in the forward direction, the software program must wait until the contents in FIFO are empty and the busy line is low to make sure that all data can reach the peripheral device successfully.

(17) Interrupts

It is necessary to generate an interrupt when any of the following states are reached:

1. serviceIntr = 0, dmaEn = 0, direction = 0, and the number of PWords in the FIFO is greater than or equal to writeIntrThreshold
2. serviceIntr = 0, dmaEn = 0, direction = 1, and the number of PWords in the FIFO is greater than or equal to readIntrThreshold
3. serviceIntr = 0, dmaEn = 1, and DMA reaches the terminal count
4. nErrIntrEn = 0 and nFault goes from high to low or when nErrIntrEn is set from 1 to 0 and nFault is asserted
5. ackIntEn = 1. In current implementations of using existing parallel ports, the interrupt generated may be either edge or level trigger type

(18) Interrupt Driven Programmed I/O

It is also possible to use an interrupt-driven programmed I/O to execute either ECP or parallel port FIFOs. An interrupt will occur in the forward direction when serviceIntr is “0” and the number of free PWords in the FIFO is equal to or greater than writeIntrThreshold. If either of these conditions is not met, it may be filled with writeIntrThreshold PWords. An interrupt will occur in the reverse direction when serviceIntr is “0” and the number of available PWords in the FIFO is equal to readIntrThreshold. If it is full, the FIFO can be completely emptied in a single burst. If it is not full, only a number of PWords equal to readIntrThreshold may be read from the FIFO in a single burst. In the Test mode, software can determine the values of writeIntrThreshold, readIntrThreshold, and FIFO depth while accessing the FIFO.

Any PC LPC bus implementation that is adjusted to expedite DMA or I/O transfer must ensure that the bandwidth on the ISA is maintained on the interface. Although the LPC (even PCI) bus of PC cannot be directly controlled, the interface bandwidth of ECP port can be constrained to perform at the optimum speed.

(19) Standard Parallel Port

In the forward direction with DMA, the standard parallel port is run at or close to the permitted peak bandwidth of 500 KB/sec. The state machine does not examine nAck, but just begins the next DMA based on the Busy signal.

9.9 Consumer Remote Control (TV Remote) IR (CIR)

9.9.1 Overview

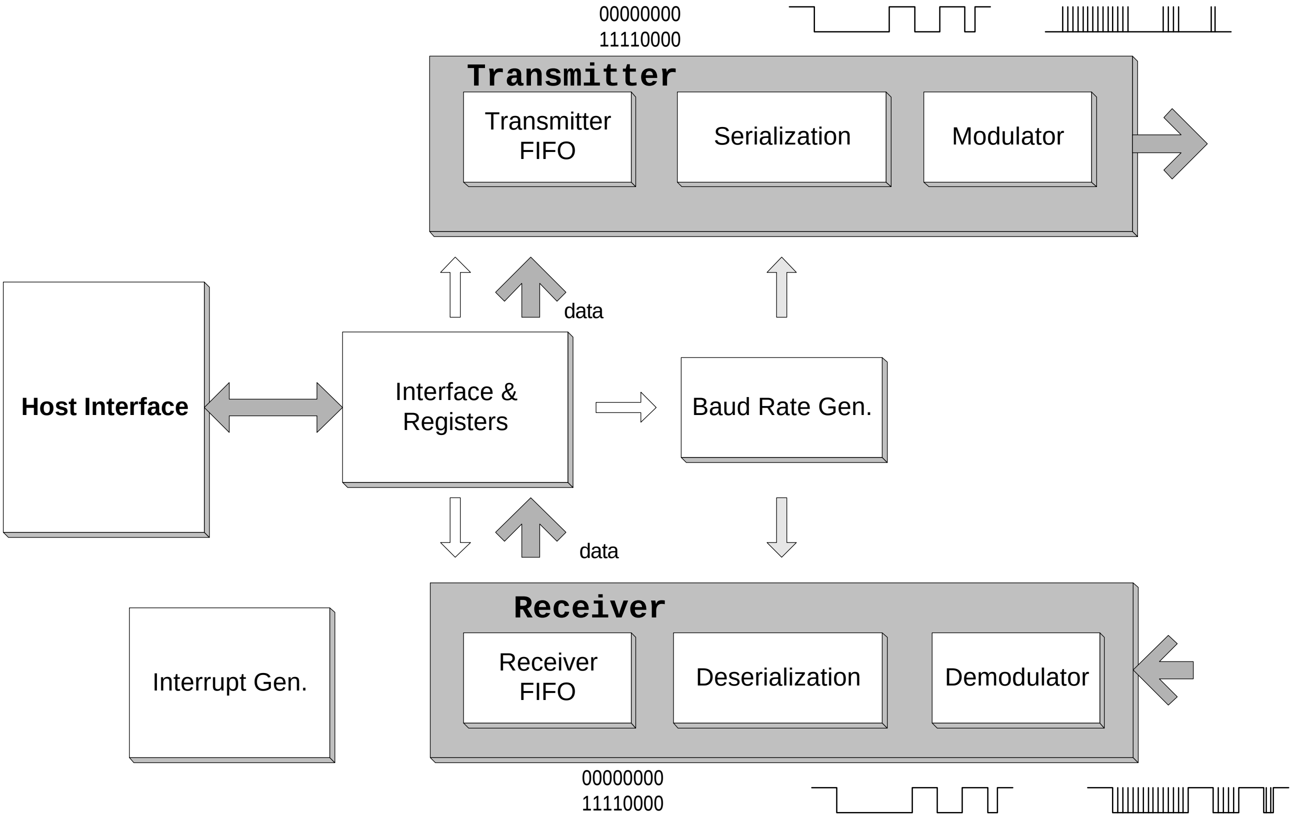
The CIR is used in Consumer Remote Control equipment, and is a programmable amplitude shift keyed (ASK) serial communication protocol. By adjusting frequencies, baud rate divisor values and sensitivity ranges, the CIR registers are able to support the popular protocols such as RC-5, NEC, and RECS-80. Software driver programming can support new protocols.

9.9.2 Features

- Supports 30 kHz - 57kHz (low frequency) or 400 kHz – 500 kHz (high frequency) carrier transmission
- Baud rates up to 115200 BPS (high frequency)
- Demodulation optional
- Supports transmission run-length encoding and deferral functions
- 32-byte FIFO for data transmission or data reception

9.9.3 Block Diagram

The CIR consists of the Transmitter and Receiver parts. The Transmitter part is responsible for transmitting data to the FIFO, processing the FIFO data by serialization, modulation and sending out the data through the LED device. The Receiver part is responsible for receiving data, processing data by demodulation, deserialization and storing data in the Receiver FIFO.



9.9.4 Transmit Operation

The data written to the Transmitter FIFO will be exactly serialized from LSB to MSB, modulated with carrier frequency and sent to the CIRTX output. The data is either in bit-string format or run-length decode.

Before the data transmission can begin, code byte write operations must be performed to the Transmitter FIFO DR. The bit TXRLE in the TCR1 should be set to "1" before the run-length decodes data can be written into the Transmitter FIFO. Setting TXENDF in the TCR1 will enable the data transmission deferral, and avoid the transmitter FIFO underrun. The bit width of the serialized bit string is determined by the value programmed in the baud rate divisor registers BDLR and BDHR. When the bits HCFS and CFQ[4:0] are set, either the high-speed or low-speed carrier range is selected, and the corresponding carrier frequency will also be determined. Bits TXMPM[1:0] and TXMPW[2:0] specify the pulse numbers in a bit width and the required duty cycles of the carrier pulse according to the communication protocol. Only a logic "0" can activate the Transmitter LED in the format of a series of modulating pulses.

9.9.5 Receive Operation

The Receiver function is enabled if the bit RXEN in the RCR is set to "1". Either demodulated or modulated RX# signal is loaded into the Receiver FIFO, and the bit RXEND in the RCR determines the demodulation logic should be used or not. Determine the baud rate by programming the baud rate divisor registers BDLR and BDHR, and the carrier frequencies by programming the bits HCFS and CFQ[4:0]. Set RDWOS to "0" to syn. The bit RXACT in the RCR is set to "1" when the serial data or the selected carrier is incoming, and the sampled data will then be kept in the Receiver FIFO. Write "1" to the bit RXACT to stop the Receiver operation; "0" to the bit RXEN to disable the Receiver.

9.9.6 Register Descriptions and Address

Table 9-44. List of CIR Registers

Address	R/W	Default	Register Name
Base + 0h	R/W	FFh	CIR Data Register (DR)
Base + 1h	R/W	00h	CIR Interrupt Enable Register (IER)
Base + 2h	R/W	01h	CIR Receiver Control Register (RCR)
Base + 3h	R/W	00h	CIR Transmitter Control Register 1 (TCR1)
Base + 4h	R/W	5Ch	CIR Transmitter Control Register 2 (TCR2)
Base + 5h	RO	00h	CIR Transmitter Status Register (TSR)
Base + 6h	RO	00h	CIR Receiver Status Register (RSR)
Base + 5h	R/W	00h	CIR Baud Rate Divisor Low Byte Register (BDLR)
Base + 6h	R/W	00h	CIR Baud Rate Divisor High Byte Register (BDHR)
Base + 7h	R/W	01h	CIR Interrupt Identification Register (IIR)

9.9.6.1 CIR Data Register (DR)

The DR, an 8-bit **read/write** register, is the data port for CIR. Data are transmitted and received through this register.

Address: Base address + 0h

Bit	R/W	Default	Description
7 – 0	R/W	FFh	**CIR Data Register (DR[7:0])** Writing data to this register causes data to be written to the Transmitter FIFO. Reading data from this register causes data to be received from the Receiver FIFO.

9.9.6.2 CIR Interrupt Enable Register (IER)

The IER, an 8-bit r**ead/write** register, is used to enable the CIR interrupt request.

Address: Base address + 1h

Bit	R/W	Default	Description
7 – 6	-	-	**Reserved for ITE Use**
5	R/W	0	**RESET (RESET)** This bit is a software reset function. Writing a “1” to this bit resets the registers of DR, IER, TCR1, BDLR, BDHR and IIR. This bit is then cleared to initial value automatically.
4	R/W	0	**Baud Rate Register Enable Function Enable (BR)** This bit is used to control the baud rate registers enable read/write function. Set this bit to “1” to enable the baud rate registers for CIR. Set this bit to “0” to disable the baud rate registers for CIR.
3	R/W	0	**Interrupt Enable Function Control (IEC)** This bit is used to control the interrupt enable function. Set this bit to “1” to enable the interrupt request for CIR. Set this bit to “0” to disable the interrupt request for CIR.
2	R/W	0	**Receiver FIFO Overrun Interrupt Enable (RFOIE)** This bit is used to control Receiver FIFO Overrun Interrupt request. Set this bit to “1” to enable Receiver FIFO Overrun Interrupt request. Set this bit to “0” to disable Receiver FIFO Overrun Interrupt request.
1	R/W	0	**Receiver Data Available Interrupt Enable (RDAIE)** This bit is used to enable Receiver Data Available Interrupt request. The Receiver will generate this interrupt when the data available in the FIFO exceeds the FIFO threshold level. Set this bit to “1” to enable Receiver Data Available Interrupt request. Set this bit to “0” to disable Receiver Data Available Interrupt request.
0	R/W	0	**Transmitter Low Data Level Interrupt Enable (TLDLIE)** This bit is used to enable Transmitter Low Data Level Interrupt request. The Transmitter will generate this interrupt when the data available in the FIFO is less than the FIFO threshold level. Set this bit to “1” to enable Transmitter Low Data Level Interrupt request. Set this bit to “0” to disable Transmitter Low Data Level Interrupt request.

9.9.6.3 CIR Receiver Control Register (RCR)

The RCR, an 8-bit **read/write** register, is used to control the CIR Receiver.

Address: Base address + 2h

Bit	R/W	Default	Description
7	R/W	0	**Receiver Data Without Sync. (RDWOS)** This bit is used to control the sync. logic for Receiving data. Set this bit to “1” to obtain the receiving data without sync. logic. Set this bit to “0” to obtain the receiving data in sync. logic.
6	R/W	0	**High-Speed Carrier Frequency Select (HCFS)** This bit is used to select Carrier Frequency between high-speed and low-speed. **030-58 kHz (Default)** 1400-500 kHz
5	R/W	0	**Receiver Enable (RXEN)** This bit is used to enable Receiver function. Set this bit to 1’ to enable the Receiver function. Set this bit to”0” to disable the Receiver function. When the Receiver is enabled, RXACT will be active if the selected carrier frequency is received.
4	R/W	0	**Receiver Demodulation Enable (RXEND)** This bit is used to control the Receiver Demodulation logic. Set this bit to “1” to enable Receiver Demodulation logic. Set this bit to “0” to disable Receiver Demodulation logic. If the Receiver device can not demodulate the correct carrier, set this bit to “1”.
3	R/W	0	**Receiver Active (RXACT)** This bit is used to control the Receiver operation. This bit is set to “0” when the Receiver is inactive. This bit will be set to “1” when the Receiver detects a pulse (RXEND=0) or pulse-train (RXEND=1) with correct carrier frequency. The Receiver then starts to sample the input data when Receiver Active is set. Write a “1” to this bit to clear the Receiver Active condition and make the Receiver enter the inactive mode.
2-0	R/W	001	**Receiver Demodulation Carrier Range (RXDCR[2:0])** These three bits are used to set the tolerance of the Receiver Demodulation carrier frequency. See Table 9-46 and Table 9-47.

9.9.6.4 CIR Transmitter Control Register 1 (TCR1)

The TCR1, an 8-bit **read/write** register, is used to control the Transmitter.

Address: Base address + 3h

Bit	R/W	Default	Description
7	R/W	0	**FIFO Clear (FIFOCLR)** Writing a "1" to this bit clears the FIFO. This bit is then cleared to "0" automatically.
6	R/W	0	**Internal Loopback Enable (ILE)** This bit is used to execute internal loopback for test and must be "0" in normal operation. Set this bit to "0" to disable the Internal Loopback mode. Set this bit to "1" to enable the Internal Loopback mode.
5 - 4	R/W	0	**FIFO Threshold Level (FIFOTL)** These two bits are used to set the FIFO threshold level. The FIFO length is 32 bytes for TX or RX function (ILE = 0) in normal operation and 16 bytes for both TX and RX in internal loopback mode (ILE = 1). **16-Byte Mode 32-Byte Mode** **00 1 1(Default)** 01 3 7 10 7 17 11 13 25
3	R/W	0	**Transmitter Run Length Enable (TXRLE)** This bit controls the Transmitter Run-Length encoding/decoding mode, which condenses a series of "1" or "0" into one byte with the bit value stored in bit 7 and number of bits minus 1 in bits 6 – 0. Set this bit to "1" to enable the Transmitter Run-Length encoding/decoding mode. Set this bit to "0" to disable the Transmitter Run-Length encoding/decoding mode.
2	R/W	0	**Transmitter Deferral (TXENDF)** This bit is used to avoid Transmitter underrun condition. When this bit is set to "1", the Transmitter FIFO data will be retained until the transmitter time-out condition occurs or the FIFO reaches full.
1 – 0	R/W	0	**Transmitter Modulation Pulse Mode (TXMPM[1:0])** These two bits are used to define the Transmitter modulation pulse mode. **TXMPM[1:0] Modulation Pulse Mode** C_pls mode (Default): Pulses are generated continuously for the entire logic "0" bit time 8_pls mode: 8 pulses are generated for each logic "0" bit 6_pls mode: 6 pulses are generated for each logic "0" bit 11: Reserved

9.9.6.5 CIR Transmitter Control Register 2 (TCR2)

The TCR2, an 8-bit **read/write** register, is used to determine the carrier frequency.

Address: Base address + 4h

<table>
<tr><th>Bit</th><th>R/W</th><th>Default</th><th>Description</th></tr>
<tr><td>7 – 3</td><td>R/W</td><td>01011</td><td><b>Carrier Frequency (CFQ[4:0])</b>
These five bits are used to determine the modulation carrier frequency. See Table 9-45.</td></tr>
<tr><td>2 – 0</td><td>R/W</td><td>100</td><td><b>Transmitter Modulation Pulse Width (TXMPW[2:0])</b>
These three bits are used to set the Transmitter Modulation pulse width. The duty cycle of the carrier will be determined according to the settings of Carrier Frequency and the selection of Transmitter Modulation Pulse Width.
<table>
<tr><th>TXMPW[2:0]</th><th>HCFS = 0</th><th>HCFS = 1</th></tr>
<tr><td>000</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>001</td><td>Reserved</td><td>Reserved</td></tr>
<tr><td>010</td><td>6 µs</td><td>0.7 µs</td></tr>
<tr><td>011</td><td>7 µs</td><td>0.8 µs</td></tr>
<tr><td><b>1008.7</b></td><td><b>ms 0.9</b></td><td><b>ms (Default)</b></td></tr>
<tr><td>101</td><td>10.6 µs</td><td>1.0 µs</td></tr>
<tr><td>110</td><td>13.3 µs</td><td>1.16 µs</td></tr>
<tr><td>111</td><td>Reserved</td><td>Reserved</td></tr>
</table></td></tr>
</table>

Table 9-45. Modulation Carrier Frequency

CFQ	Low Frequency (HCFS =0)	High Frequency (HCFS = 1)
00000	27 kHz	-
00010	29 kHz	-
00011	30 kHz	400 kHz
00100	31 kHz	-
00101	32 kHz	-
00110	33 kHz	-
00111	34 kHz	-
01000	35 kHz	450 kHz
01001	36 kHz	-
01010	37 kHz	-
01011	**38 kHz (Default)**	**480 kHz (Default)**
01100	39 kHz	-
01101	40 kHz	500 kHz
01110	41 kHz	-
01111	42 kHz	-
10000	43 kHz	-
10001	44 kHz	-
10010	45 kHz	-
10011	46 kHz	-
10100	47 kHz	-
10101	48 kHz	-
10110	49 kHz	-
10111	50 kHz	-
11000	51 kHz	-
11001	52 kHz	-
11010	53 kHz	-
11011	54 kHz	-
11100	55 kHz	-
11101	56 kHz	-
11110	57 kHz	-
11111	58 kHz	-

Table 9-46. Receiver Demodulation Low Frequency (HCFS = 0)

RXDCR	001		010		011		100		101		110		
CFQ	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	(Hz)
00001	26.25	29.75	24.5	31.5	22.75	33.25	21	35	19.25	36.75	17.5	38.5	28k
00010	27.19	30.81	25.38	32.63	23.56	34.44	21.75	36.25	19.94	38.06	18.13	39.88	29k
00011	28.13	31.88	26.25	33.75	24.38	35.63	22.5	37.5	20.63	39.38	18.75	41.25	30k
00100	29.06	32.94	27.13	34.88	25.19	36.81	23.25	38.75	21.31	40.69	19.38	42.63	31k
00101	30	34	28	36	26	38	24	40	22	42	20	44	32k
00110	30.94	35.06	28.88	37.13	26.81	39.19	24.75	41.25	22.69	43.31	20.63	45.38	33k
00111	31.88	36.13	29.75	38.25	27.63	40.38	25.5	42.5	23.38	44.63	21.25	46.75	34k
01000	32.81	37.19	30.63	39.38	28.44	41.56	26.25	43.75	24.06	45.94	21.88	48.13	35k
01001	33.75	38.25	31.5	40.5	29.25	42.75	27	45	24.75	47.25	22.5	49.5	36k
01010	34.69	39.31	32.38	41.63	30.06	43.94	27.75	46.25	25.44	48.56	23.13	50.88	37k
01011	**35.63**	**40.38**	**33.25**	**42.75**	**30.88**	**45.13**	28.5	**47.5**	**26.13**	**49.88**	**23.75**	**52.25**	**38k**
01100	36.56	41.44	34.13	43.88	31.69	46.31	29.25	48.75	26.81	51.19	24.38	53.63	39k
01101	37.5	42.5	35	45	32.5	47.5	30	50	27.5	52.5	25	55	40k
01110	38.44	43.56	35.88	46.13	33.31	48.69	30.75	51.25	28.19	53.81	25.63	56.38	41k
01111	39.38	44.63	36.75	47.25	34.13	49.88	31.5	52.5	28.88	55.13	26.25	57.75	42k
10000	40.31	45.69	37.63	48.38	34.94	51.06	32.25	53.75	29.56	56.44	26.88	59.13	43k
10001	41.25	46.75	38.5	49.5	35.75	52.25	33	55	30.25	57.75	27.5	60.5	44k
10010	42.19	47.81	39.38	50.63	36.56	53.44	33.75	56.25	30.94	59.06	28.13	61.88	45k
10011	43.13	48.88	40.25	51.75	37.38	54.63	34.5	57.5	31.63	60.38	28.75	63.25	46k
10100	44.06	49.94	41.13	52.88	38.19	55.81	35.25	58.75	32.31	61.69	29.38	64.63	47k
10101	45	51	42	54	39	57	36	60	33	63	30	66	48k
10110	45.94	52.06	42.88	55.13	39.81	58.19	36.75	61.25	33.69	64.31	30.63	67.38	49k
10111	46.88	53.13	43.75	56.25	40.63	59.38	37.5	62.5	34.38	65.63	31.25	68.75	50k
11000	47.81	54.19	44.63	57.38	41.44	60.56	38.25	63.75	35.06	66.94	31.88	70.13	51k
11001	49.18	54.55	46.88	57.69	44.78	61.22	42.86	65.22	41.1	69.77	39.47	75	52k
11010	49.69	56.31	46.38	59.63	43.06	62.94	39.75	66.25	36.44	69.56	33.13	72.88	53k
11011	50.63	57.38	47.25	60.75	43.88	64.13	40.5	67.5	37.13	70.88	33.75	74.25	54k
11100	51.56	58.44	48.13	61.88	44.69	65.31	41.25	68.75	37.81	72.19	34.38	75.63	55k
11101	52.5	59.5	49	63	45.5	66.5	42	70	38.5	73.5	35	77	56k
11110	53.44	60.56	49.88	64.13	46.31	67.69	42.75	71.25	39.19	74.81	35.63	78.38	57k

Table 9-47. Receiver Demodulation High Frequency (HCFS = 1)

RXDCR	001		010		011		100		101		110		
CFQ	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	(Hz)
00011	375	425	350	450	325	475	300	500	275	525	250	550	400k
01000	421.9	478.1	393.8	506.3	365.6	534.4	337.5	562.5	309.4	590.6	281.3	618.8	450k
01011	450	510	420	540	390	570	360	600	330	630	300	660	**480k**
01011	468.8	531.3	437.5	562.5	406.3	593.8	375	625	343.8	656.3	312.5	687.5	500k

9.9.6.6 CIR Baud Rate Divisor Low Byte Register (BDLR)

The BDLR, an 8-bit **read/write** register, is used to program the CIR Baud Rate clock.

Address: Base address + 5h (when BR = 1)

Bit	R/W	Default	Description
7-0	R/W	00h	**Baud Rate Divisor Low Byte (BDLR[7:0])** These bits are the low byte of the register used to divide the Baud Rate clock.

9.9.6.7 CIR Baud Rate Divisor Register High Byte (BDHR)

The BDHR, an 8-bit **read/write** register, is used to program the CIR Baud Rate clock.

Address: Base address + 6h (when BR = 1)

Bit	R/W	Default	Description
7-0	R/W	00h	**Baud Rate Divisor High Byte (BDHR[7:0])** These bits are the high byte of the register used to divide the Baud Rate clock.

Baud rate divisor = 115200 / baud rate
Ex1: 2400 bps → 115200 /2400 = 48 → 48(d) = 0030(h)
BDHR = 00(h), BDLR = 30(h)
Ex2: bit width = 0.565 ms → 1770 bps → 115200 / 1770 = 65 (d) = 41(h)
BDHR = 00(h), BDLR = 41(h)

9.9.6.8 CIR Transmitter Status Register (TSR)

The TSR, an 8-bit **read only** register, provides the Transmitter FIFO status.

Address: Base address + 5h

Bit	R/W	Default	Description
7-6	RO	-	**Reserved**
5-0	RO	000000	**Transmitter FIFO Byte Count (TXFBC[5:0])** Return the number of bytes left in the Transmitter FIFO.

9.9.6.9 CIR Receiver FIFO Status Register (RSR)

The RSR, an 8-bit **read only** register, provides the Receiver FIFO status.

Address: Base address + 6h

Bit	R/W	Default	Description
7	RO	0	**Receiver FIFO Time-out (RXFTO)** This bit will be set to "1" when a Receiver FIFO time-out condition occurs. The conditions that must exist for a Receiver FIFO time-out condition to occur include the followings: a. At least one byte has been in the Receiver FIFO is not empty for 64 ms or more, and b. The receiver has been inactive (RXACT=0) for over 64 ms or more, and c. More than 64 ms have elapsed since the last byte was read from the Receiver FIFO by the CPU
6	-	-	**Reserved**
5 – 0	RO	000000	**Receiver FIFO Byte Count (RXFBC)** Return the number of bytes left in the Receiver FIFO.

9.9.6.10 CIR Interrupt Identification Register (IIR)

The IIR, an 8-bit **read only** register, is used to identify the pending interrupt.

Address: Base address + 7h

Bit	R/W	Default	Description
7 - 3	-	-	**Reserved**
2 – 1	RO	00	**Interrupt Identification (IID[1:0])** These two bits are used to identify the source of the pending interrupt. **IID[1:0] Interrupt Source** 00 No interrupt 01 Transmitter Low Data Level Interrupt 10 Receiver Data Stored Interrupt 11 Receiver FIFO Overrun Interrupt
0	RO	1	**Interrupt Pending (IIP)** This bit will be set to "1" while an interrupt is pending.

9.10 Game Port Interface

The Game port integrates four timers for two joysticks. The IT8705F allows the Game Port base address to be located within the host I/O address space 100h to 0FFFh. Currently, most game software assume that the Game (or Joystick) I/O port is located at 201h.

A write to the Game port base address will trigger four timers. A read from the same address returns four bits that correspond to the output from the four timers, and other four status bits corresponding to the joystick buttons will also be returned. A button value of 0 indicates that the button is pressed. When the Game port base address is written, the X/Y timer bits go high. Once the Game port base address is written, each timer output remains high for a duration of time determined by the current joystick position.

9.10.1 Game Port (Base+0h)

Bit	Symbol	Description
7	JSBB2	**Joystick B, Button 2 (pin 56 of Joystick connector)**
6	JSBB1	**Joystick B, Button 1 (pin 55 of Joystick connector)**
5	JSBCY	**Joystick B, Coordinate Y (pin 54 of Joystick connector)**
4	JSBCX	**Joystick B, Coordinate X (pin 53 of Joystick connector)**
3	JSAB2	**Joystick A, Button 2 (pin 52 of Joystick connector)**
2	JSAB1	**Joystick A, Button 1 (pin 51 of Joystick connector)**
1	JSACY	**Joystick A, Coordinate Y (pin 50 of Joystick connector)**
0	JSACX	**Joystick A, Coordinate X (pin 49 of Joystick connector)**

9.11 FLASH ROM Interface

The IT8705F offers a solution for BIOS ROM on ISA-less bus MB. The IT8705F incorporates a unique BIOS ROM interface. The interface decodes the memory cycle of the LPC protocol and translates to an ISA-like memory cycle. As illlustrated in Figure 9-8, the width of the address bus is 18 bits, enabling the interface to support 4MB, 2MB and 1MB ROM chips. The width of the data bus is 8-bit wide. Three control signals are supported by the interface: FCS#, FWE# and FRD#. These three control signals are connected to the chip enable, write enable and output enable signal pins respectively on the ROM chip.

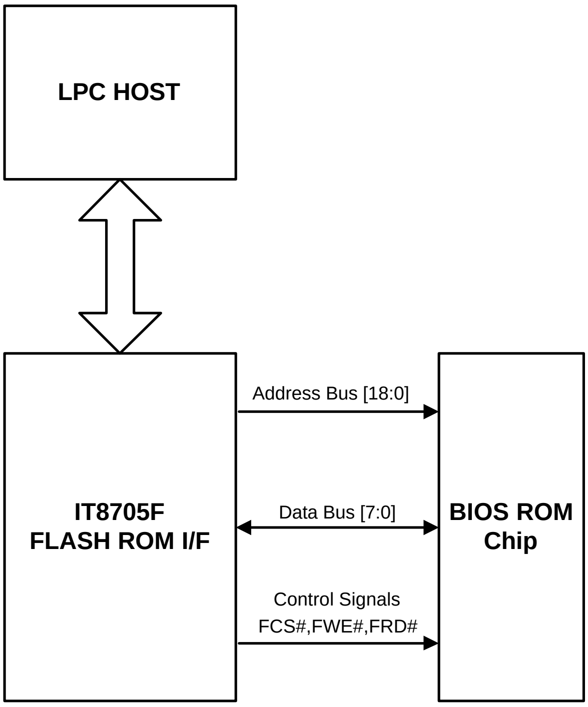


Figure 9-8. FLASH ROM Interface Diagram

Please refer to Table 7-1 decode segments where the BIOS ROM can be located. The selection of decode segment is decided by the power-on strapping. The FCS# signal is asserted when the address of LPC memory cycle falls in the specified segment. The FWE# and FRD# will be asserted as the command type in the memory cycle.

9.12 MIDI Interface

The IT8705F supports the MIDI capability by incorporating hardware to emulate the MPU-401 in the UART mode. It is software compatible with MPU-401 interface, but only supports the **UART mode** (non-intelligent mode). The UART is used to convert parallel data to the serial data required by MIDI. The serial data format is RS-232 like: 1 start bit, 8 data bits, and 1 stop bit. The serial data rate is fixed at 31.25 Kbaud.

9.12.1 MPU-401 Register Interface

The MPU-401 logical device occupies two consecutive I/O spaces. The device also uses an interrupt. Both the base address and the interrupt level are programmable. MIDI Base+0 is the MIDI Data port, and MIDI Base+ 1 is the Command/Status port.

MIDI Data Port: The MIDI Data Port is used to transmit and receive MIDI data. When in UART mode, all transmit data is transferred through a 16-byte FIFO and receive data through another 16-byte FIFO.

MIDI Data Port, MIDI Base+0

Bit	R/W	Description
7-0	R/W	**D7-D0** MIDI data 7-0.

Command/Status Port: The Command register is used to send instructions to the MPU-401. The Status register is used to receive buffer status information from the MPU-401. These two registers occupy the same I/O address.

Command Port, MIDI Base+1

Bit	R/W	Description
7-0	WO	**C7-C0** MIDI instruction command code 7-0.

Status Port, MIDI Base+1

Bit	R/W	Description
7	RO	**RXS** Receive buffer status flag: 0: Data in Receive Buffer. 1: Receive Buffer empty.
6	RO	**TXS** Transmit buffer status flag: 0: Transmit Buffer not full. 1: Transmit Buffer full.
5-0	-	**Reserved, always report 3Fh**

9.12.2 Operation

In the IT8705F, only two MPU-401 device instructions are available: RESET (code: FFh) and UART mode command (code: 3Fh). After power-up reset, the interface is in the **Intelligent mode** (non-UART mode). In this mode, the operation is defined below:

1. All reads of the Data port, MIDI Base+0, return the acknowledged code(FEh). Because only two commands are available, the receive buffer is always loaded with an acknowledge code in the Intelligent mode.
2. All writes to the Data port, MIDI Base+0, are ignored.
3. All writes to the Command port, MIDI Base+1, are monitored and acknowledged as follows:

3Fh: Set the interface into the UART mode and loads an acknowledged code (FEh) into the receive buffer which generates an interrupt
FFh: Set the interface into the initialization condition
Others: Not implemented

UART Mode:

1. All reads of the Data port, MIDI Base+0, return the next byte in the receive buffer FIFO. The serial data received from the MIDI_IN pin is stored in the receive buffer FIFO. The bit 7 RXS of the Status register is updated to reflect the new receive buffer FIFO status. The receive data available interrupt will be issued only if the FIFO has reached its programmed trigger level. The interrupts will be cleared as soon as the FIFO drops below its trigger level. The trigger level is programmable by changing bits 2-1 of the MIDI port Special Configuration Register, LDN8_F0h.
2.
3. All writes to the Data port, MIDI Base+0, are placed in the transmit buffer FIFO. Whenever the transmit buffer FIFO is not empty, the data bytes are read from the buffer in turn and sent out from the MIDI_OUT pin. The bit 6 TXS of the Status Register is updated to reflect the new transmit buffer FIFO status.
4.
5. All writes to the Command port, MIDI Base+1, are monitored and acknowledged below:

FFh: Set the interface into the initialization condition. The interface returns to the intelligent mode
Others: No operation

10. DC Electrical Characteristics

Absolute Maximum Ratings*

Applied Voltage	-0.5V to 7.0V
Input Voltage (Vi)	-0.5V to VCC+0.5V
Output Voltage (Vo)	-0.5V to VCC + 0.3V
Operation Temperature (Topt)	0°C to +70°C
Storage Temperature	-55°C to +125°C
Power Dissipation	300mW

***Comments**

Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to this device. These are stress ratings only. Functional operation of this device at these or any other conditions above those indicated in the operational sections of this specification is not implied or intended. Exposure to the absolute maximum rating conditions for extended periods may affect device reliability.

DC Electrical Characteristics (VCC = 5V ± 5%, Ta = 0°C to + 70°C)

Symbol	Parameter	Min.	Typ.	Max.	Unit	Conditions
DIO24 Type Buffer						
V_{OL}	Low Output Voltage			0.4	V	I_{OL} = 24 mA
V_{OH}	High Output Voltage	2.4			V	I_{OH} = -12 mA
V_{IL}	Low Input Voltage			0.8	V	
V_{IH}	High Input Voltage	2.2			V	
I_{IL}	Low Input Leakage		10		µA	V_{IN} = 0
I_{IH}	High Input Leakage			-10	µA	V_{IN} = VCC
I_{OZ}	3-state Leakage			20	µA	
DIO16 Type Buffer						
V_{OL}	Low Output Voltage			0.4	V	I_{OL} = 16 mA
V_{OH}	High Output Voltage	2.4			V	I_{OH} = -16 mA
V_{IL}	Low Input Voltage			0.8	V	
V_{IH}	High Input Voltage	2.2			V	
I_{IL}	Low Input Leakage		10		µA	V_{IN} = 0
I_{IH}	High Input Leakage			-10	µA	V_{IN} = VCC
I_{OZ}	3-state Leakage			20	µA	
DIO8 Type Buffer						
V_{OL}	Low Output Voltage			0.4	V	I_{OL} = 8 mA
V_{OH}	High Output Voltage	2.4			V	I_{OH} = -8 mA
V_{IL}	Low Input Voltage			0.8	V	
V_{IH}	High Input Voltage	2.2			V	
I_{IL}	Low Input Leakage		10		µA	V_{IN} = 0

DC Electrical Characteristics (VCC = 5V ± 5%, Ta = 0°C to + 70°C)[cont’d]

Symbol	Parameter	Min.	Typ.	Max.	Unit	Conditions
I_{IH}	High Input Leakage			-10	µA	V_{IN} = VCC
I_{OZ}	3-state Leakage			20	µA	
DO40 Type Buffer						
V_{OL}	Low Output Voltage			0.5	V	I_{OL} = 48 mA
V_{OH}	High Output Voltage	2.4			V	I_{OH} = -8 mA
DI Type Buffer						
V_{IL}	Low Input Voltage			0.8	V	
V_{IH}	High Input Voltage	2.2			V	
I_{IL}	Low Input Leakage		10		µA	V_{IN} = 0
I_{IH}	High Input Leakage			-10	µA	V_{IN} = VCC

11. AC Characteristics (Vcc = 5.0V ± 5%, Ta = 0°C to + 70°C)

11.1 Clock Input Timings

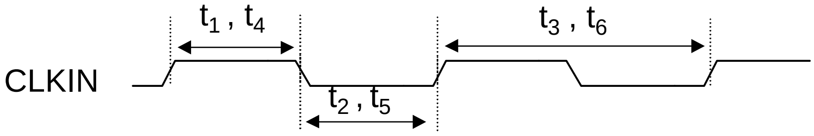


Figure 11-1. Clock Input Timings

Table 11-1. Clock Input Timings AC Table

Symbol	Parameter	Min.	Typ.	Max.	Unit
t_1	Clock high pulse width when CLKIN=48 MHz[1]		10.5		nsec
t_2	Clock low pulse width when CLKIN=48 MHz [1]		10.5		nsec
t_3	Clock period when CLKIN=48MHz [1]		21		nsec
t_4	Clock high pulse width when CLKIN=24 MHz [1]		21		nsec
t_5	Clock low pulse width when CLKIN=24 MHz [1]		21		nsec
t_6	Clock period when CLKIN=24 MHz [1]		42		nsec

Not tested. Guaranteed by design.

11.2 LCLK (PCICLK) and LRESET#

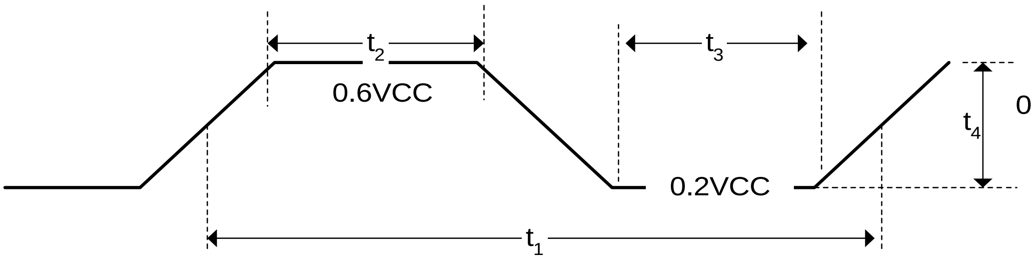


Figure 11-2. LCLK (PCICLK) and LRESET# Timings

Table 11-2. LCLK (PCICLK) and LRESET# AC Table

Symbol	Parameter	Min.	Typ.	Max.	Unit
t_1	LCLK cycle time	28			nsec
t_2	LCLK high time	11			nsec
t_3	LCLK low time	11			nsec
t_4	LRESET# low pulse width	1.5			µsec

11.3 LPC and SERIRQ Timings

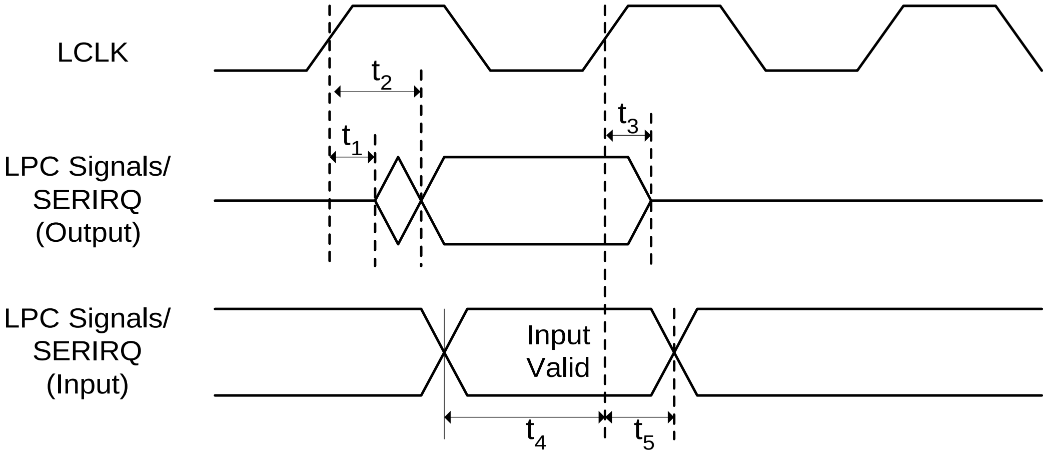


Figure 11-3. LPC and SERIRQ Timings

Table 11-3. LPC and SERIRQ Timings AC Table

Symbol	Parameter	Min.	Typ.	Max.	Unit
t_1	Float to active delay	3			nsec
t_2	Output valid delay			12	nsec
t_3	Active to float delay			6	nsec
t_4	Input setup time	9			nsec
t_5	Input hold time	3			nsec

11.4 Serial Port, ASKIR, SIR and Consumer Remote Control Timings

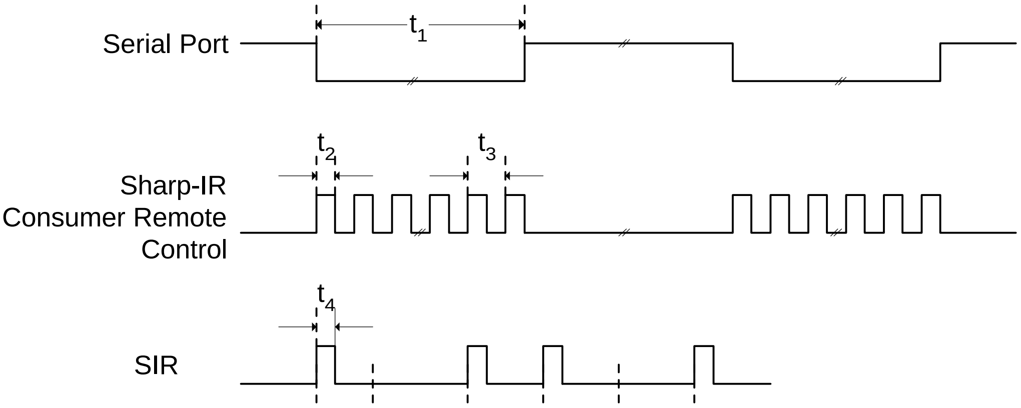


Figure 11-4. Serial Port, ASKIR, SIR and Consumer Remote Control Timings

Table 11-4. Serial Port, ASKIR, SIR and Consumer Remote Control Timings AC Table

Symbol	Parameter	Conditions	Min.	Max.	Unit
t_1	Single bit time in serial port and ASKIR	Transmitter	$t_{BTN} – 25$ [1]	$t_{BTN} + 25$	nsec
		Receiver	t_{BTN} – 2%	t_{BTN} + 2%	nsec
t_2	Modulation signal pulse width in ASKIR	Transmitter	950	1050	nsec
		Receiver	500		nsec
t_3	Modulation signal period in ASKIR	Transmitter	1975	2025	nsec
		Receiver	2000X(23/24)	2000X(25/24)	nsec
t_4	SIR signal pulse width	Transmitter, Variable	(3/16) x t_{BTN} – 25	(3/16) x t_{BTN} + 25	nsec
		Transmitter, Fixed	1.48	1.78	µsec
		Receiver	1		µsec

t_{BTN} is the nominal bit time in Serial Port, ASKIR, and SIR. It is determined by the setting on the Baud Rate Generator Divisor registers.

11.5 Modem Control Timings

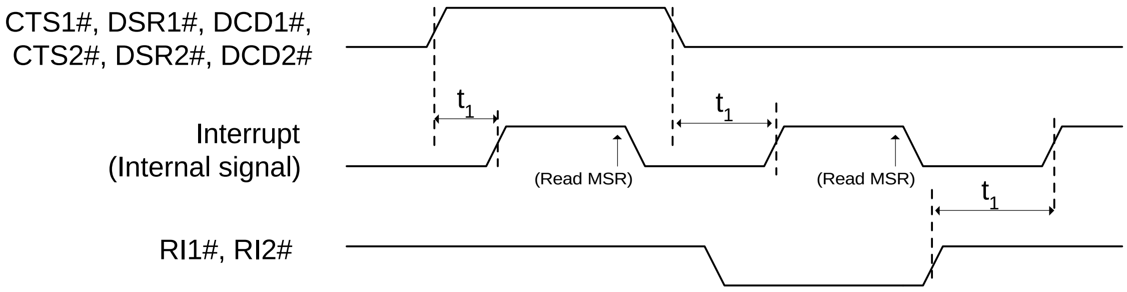


Figure 11-5. Modem Control Timings

Table 11-5. Modem Control Timings AC Table

Symbol	Parameter	Min.	Typ.	Max.	Unit
t_1	Float to active delay			40	nsec

11.6 Floppy Disk Drive Timings

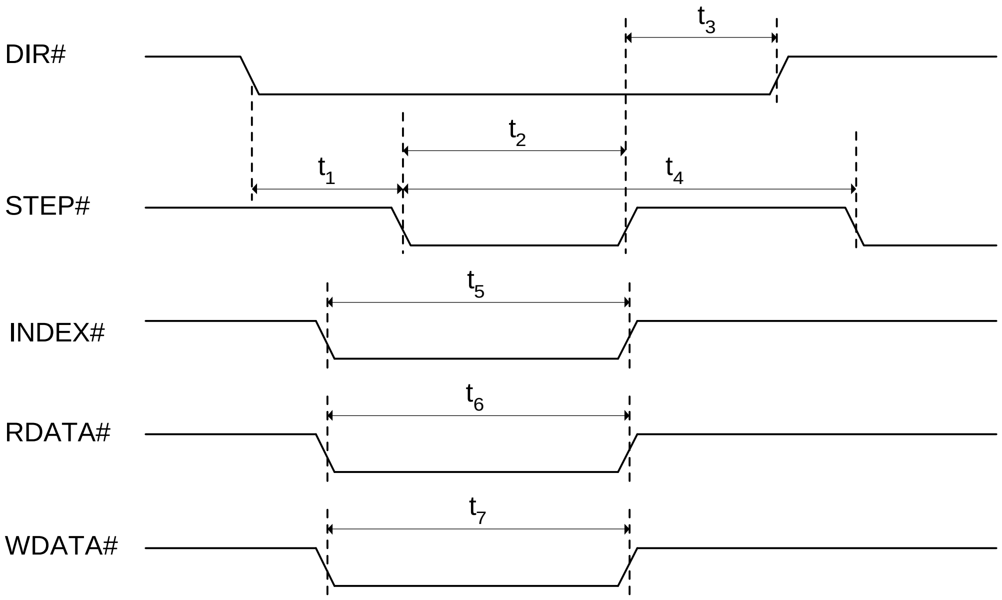


Figure 11-6. Floppy Disk Drive Timings

Table 11-6. Floppy Disk Drive Timings AC Table

Symbol	Parameter	Min.	Typ.	Max.	Unit
t_1	DIR# active to STEP# low		4X t_{mclk}[1]		nsec
t_2	STEP# active time (low)		24X t_{mclk}		nsec
t_3	DIR# hold time after STEP#		t_{SRT}[2]		msec
t_4	STEP# cycle time		t_{SRT}		msec
t_5	INDEX# low pulse width	2X t_{mclk}			nsec
t_6	RDATA# low pulse width	40			nsec
t_7	WDATA# low pulse width		1X t_{mclk}		nsec

1. t_{mclk} is the cycle of main clock for the microcontroller of FDC. t_{mclk} =8M/ 4M/ 2.4M/ 2M for 1M/ 500K/ 300K/ 250K bps transfer rates respectively.
2. t_{SRT} is the cycle of the Step Rate Time. Please refer to the functional description of the SPECITY command of FDC.

11.7 EPP Address or Data Write Cycle

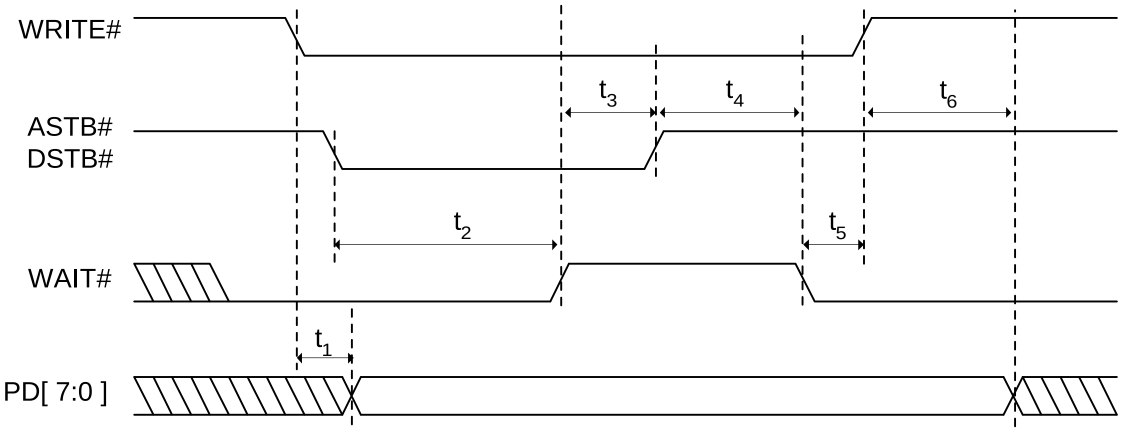


Figure 11-7. EPP Address or Data Write Cycle

Table 11-7. EPP Address or Data Write Cycle AC Table

Symbol	Parameter	Min.	Typ.	Max.	Unit
t_1	WRITE# asserted to PD[7:0] valid			50	nsec
t_2	ASTB# or DSTB# asserted to WAIT# de-asserted	0		10	nsec
t_3	WAIT# de-asserted to ASTB# or DSTB# de-asserted	65		135	nsec
t_4	ASTB# or DSTB# de-asserted to WAIT# asserted	0			nsec
t_5	WAIT# asserted to WRITE# de-asserted	65			nsec
t_6	PD[7:0] invalid after WRITE# de-asserted	0			nsec

11.8 EPP Address or Data Read Cycle

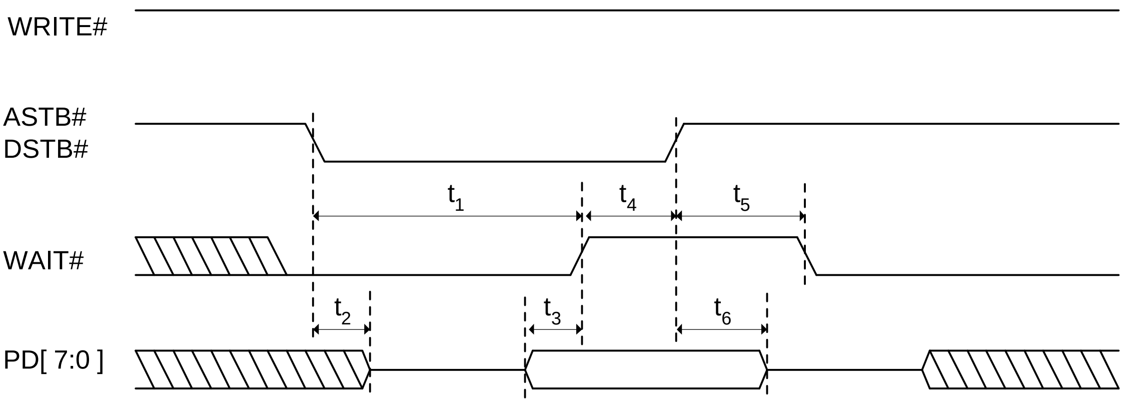


Figure 11-8. EPP Address or Data Read Cycle

Table 11-8. EPP Address or Data Read Cycle AC Table

Symbol	Parameter	Min.	Typ.	Max.	Unit
t_1	ASTB# or DSTB# asserted to WAIT# de-asserted			10	nsec
t_2	ASTB# or DSTB# asserted to PD[7:0] Hi-Z	0			nsec
t_3	PD[7:0] valid to WAIT# de-asserted	0			nsec
t_4	WAIT# de-asserted to ASTB# or DSTB# de-asserted	65		135	nsec
t_5	ASTB# or DSTB# de-asserted to WAIT# asserted	0			nsec
t_6	PD[7:0] invalid after ASTB# or DSTB# de-asserted	20			nsec

11.9 ECP Parallel Port Forward Timings

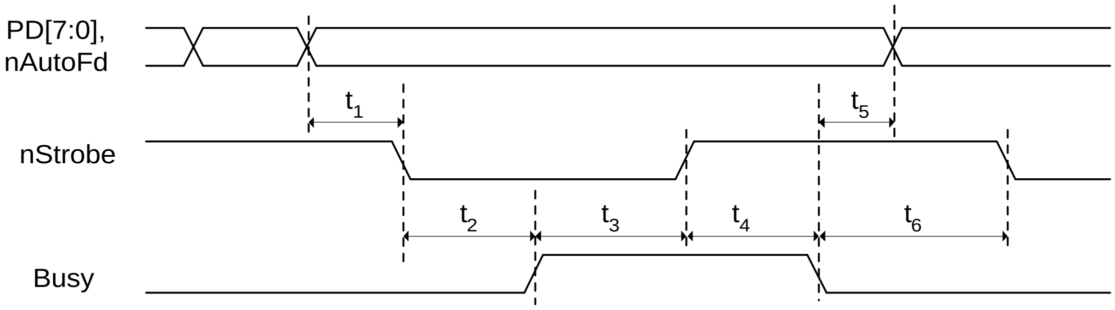


Figure 11-9. ECP Parallel Port Forward Timings

Table 11-9. ECP Parallel Port Forward Timings AC Table

Symbol	Parameter	Min.	Typ.	Max.	Unit
t_1	PD[7:0] and nAutoFd valid to nStrobe asserted			50	nsec
t_2	nStrobe asserted to Busy asserted	0			nsec
t_3	Busy asserted to nStrobe de-asserted	70		170	nsec
t_4	nStrobe de-asserted to Busy de-asserted	0			nsec
t_5	Busy de-asserted to PD[7:0] and nAutoFd changed	80		180	nsec
t_6	Busy de-asserted to nStrobe asserted	70		170	nsec

11.10 ECP Parallel Port Backward Timings

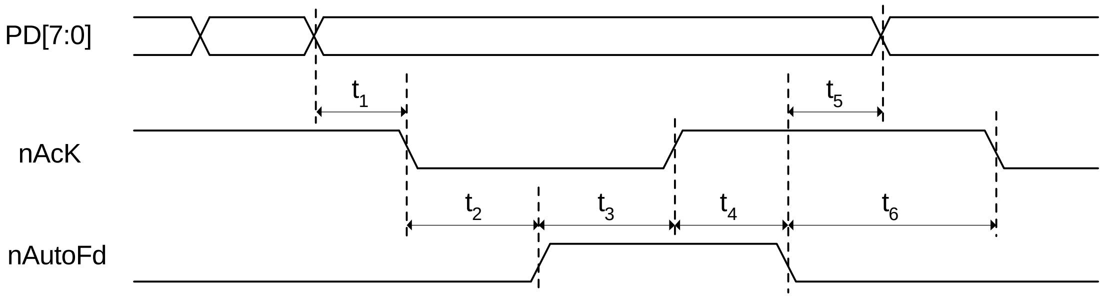


Figure 11-10. ECP Parallel Port Backward Timings

Table 11-10. ECP Parallel Port Backward Timings AC Table

Symbol	Parameter	Min.	Typ.	Max.	Unit
t_1	PD[7:0] valid to nAck asserted	0			nsec
t_2	nAck asserted to nAutoFd asserted	70		170	nsec
t_3	nAutoFd asserted to nAck de-asserted	0			nsec
t_4	nAck de-asserted to nAutoFd de-asserted	70		170	nsec
t_5	nAutoFd de-asserted to PD[7:0] changed	0			nsec
t_6	nAutoFd de-asserted to nAck asserted	0			nsec

11.11 Flash ROM I/F Write Timings

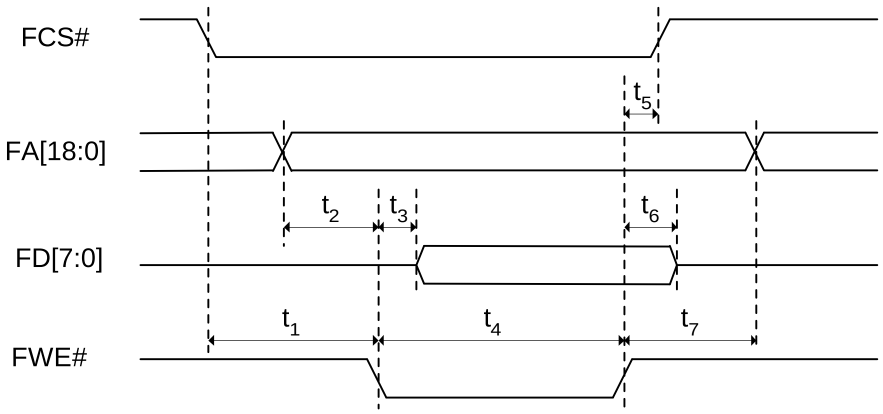


Figure 11-11. Flash ROM I/F Write Timings

Table 11-11. Flash ROM I/F Write Timing AC Table

Symbol	Parameter	Min.	Typ.[1]	Max.	Unit
t_1	FCS# valid to FWE# asserted	11Xt_{pci}			nsec
t_2	FA[18:0] valid to FWE# asserted	4Xt_{pci}			nsec
t_3	FWE# asserted to FD[7:0] valid		1Xt_{pci}		nsec
t_4	FWE# active pulse width		20Xt_{pci}		nsec
t_5	FWE# de-asserted to FCS# changed		(1/2)Xt_{pci}		nsec
t_6	FWE# de-asserted to FD[7:0] invalid		1Xt_{pci}		nsec
t_7	FWE# de-asserted to FA[18:0] changed	3Xt_{pci}			nsec

1. t_{pci} is cycle of PCICLK.

11.12 Flash ROM I/F Read Timings

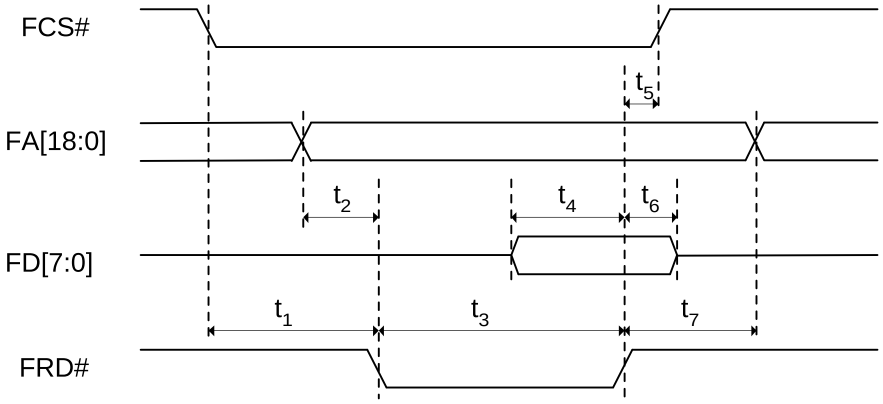


Figure 11-12. Flash ROM I/F Read Timings

Table 11-12. Flash ROM I/F Read Timings AC Table

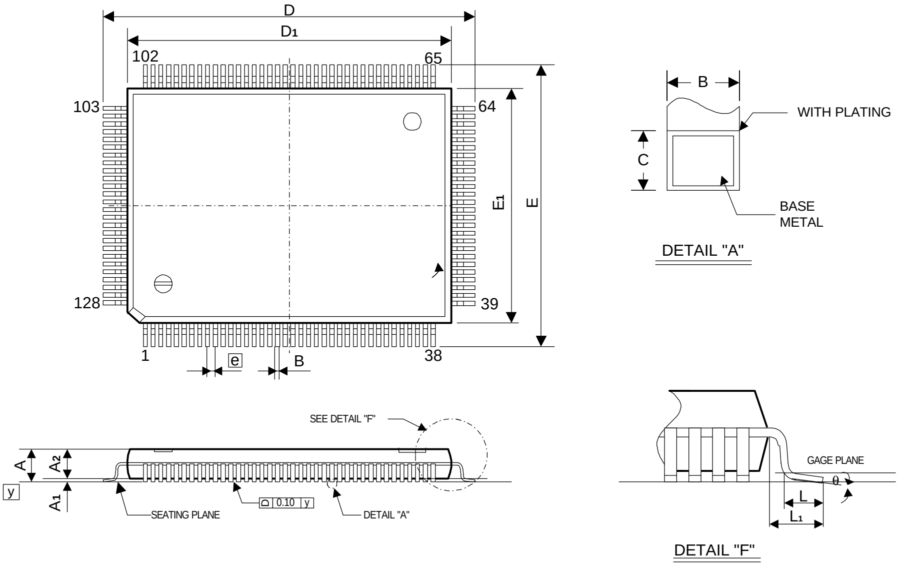
Symbol	Parameter	Min.	Typ.[1]	Max.	Unit
t_1	FCS# valid to FRD# asserted	$9Xt_{pci}$			nsec
t_2	FA[18:0] valid to FRD# asserted	$1Xt_{pci}$			nsec
t_3	FRD# active pulse width		$20Xt_{pci}$		nsec
t_4	FD[7:0] setup time	$4Xt_{pci}$			nsec
t_5	FRD# de-asserted to FCS# changed		$1Xt_{pci}$		nsec
t_6	FD[7:0] hold time	0			nsec
t_7	FRD# de-asserted to FA[18:0] changed		$5Xt_{pci}$		nsec

[1] t_{pci} is cycle of PCICLK.

12. Package Information

PQFP 128L Outline Dimensions

unit: inches/mm



Symbol	Dimension in Inch			Dimension in mm		
	Min.	Nom.	Max.	Min.	Nom.	Max.
A	-	-	0.134	-	-	3.40
A_1	0.010	-	-	0.25	-	-
A_2	0.107	0.112	0.117	2.73	2.85	2.97
B	0.007	0.009	0.011	0.17	0.22	0.27
C	0.004	-	0.008	0.09	-	0.20
D	0.906	0.913	0.921	23.00	23.20	23.40
D_1	0.783	0.787	0.791	19.90	20.00	20.10
E	0.669	0.677	0.685	17.00	17.20	17.40
E_1	0.547	0.551	0.555	13.90	14.00	14.10
e	0.020 BSC			0.5 BSC		
L	0.029	0.035	0.041	0.73	0.88	1.03
L_1	0.063 BSC			1.60 BSC		
y	-	-	0.004	-	-	0.10
θ	0¢ X	-	7¢ X	0¢ X	-	7¢ X

Notes:

1. DIMENSIONS D_1 AND E_1 DO NOT INCLUDE MOLD PROTRUSION. BUT MOLD MISMATCH IS INCLUDED.
2. DIMENSIONS B DOES NOT INCLUDE DAMBAR PROTRUSION.
3. CONTROLLING DIMENSION: MILLIMETER.

13. Ordering Information

Part No.	Package
IT8705F	128 PQFP