

iCE65 as IrDA 4PPM Tranceiver

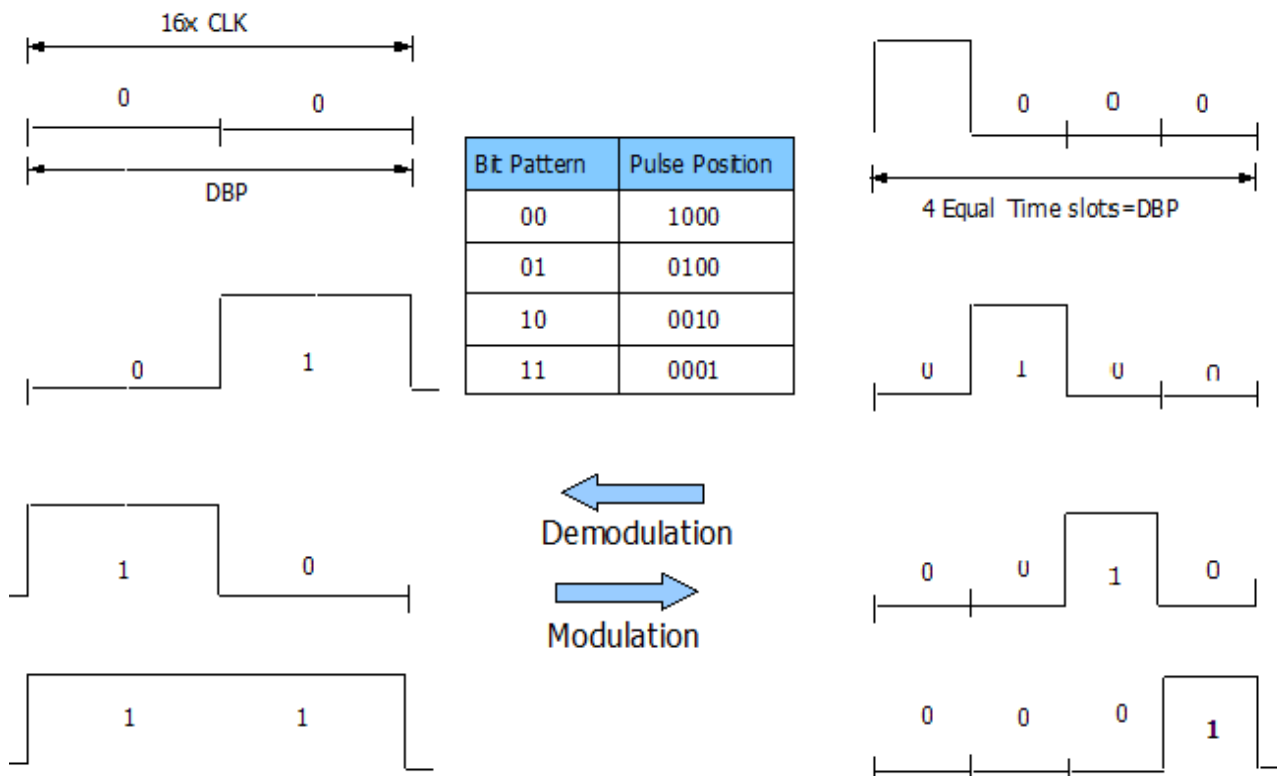
Overview

The IrDA creates an interoperable, low cost, low power, half-duplex serial data interconnection standard that supports a walk-up, point-to-point user model that is adaptable to a wide range of appliances and devices. IR technologies are better suited for short distance, low-to-medium data throughput, wireless communication channels. This design example illustrates the implementation of an IrDA Fast InfraRed(FIR) Transceiver using SiliconBlue iCE65 FPGAs.

Description

IrDA defines a set of specifications, or protocol stack, that provides for the establishment and maintenance of a link so that error free communication is possible. The IrDA Standards include three mandatory specifications: the Physical Layer, Link Access Protocol (IrLAP), and Link Management Protocol (IrLMP). Beginning with Version 1.1 of the IrDA physical layer specification, a 4 Mbps data rate is supported. The FIR(4 Mbps) scheme uses the four pulse position modulation (4PPM) scheme, where one complete symbol is represented by 4 equal time slices called "chips". In FIR every chip has a 125ns duration, and every symbol represents 2 bits of data. Because there are four unique chip positions within each symbol in 4PPM, four independent symbols exist in which only one chip is logically a "one" while all other chips are logically a "zero." We define these four unique symbols to be the only legal data symbols (DD) allowed in 4PPM. Each DD represents two bits of payload data, or a single "data bit pair (DBP)", so that a byte of payload data can be represented by four DDs in sequence.

Figure 1 illustrates FIR(4PPM) modulation/demodulation schemes.



Implementation

The figure 2 and 3 illustrates the high level block diagram of IrDA Transmitter and Receiver modules. In IrDA FIR mode, a special packet format is used for data transfer. The packet format is divided into 4 blocks as shown in Table 1.

1. PA: Preamble-consists of exactly 16 repeated transmission of following symbols ("1000 0000 1010 1000").

2. STA (Start Sequence): It occurs only once and has the the following symbols("0000 0011 0000 0011 0110 0000 0110 0000"). After the preamble, the receiver looks for STA sequence for synchronization.

3. DD (Actual Data encoded in 4ppm scheme along with CRC32 bit): After the start sequence, the transmitter transmits the data in encoded form along with CRC32 to facilitate error detection at receiver end.

$$CRC(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

In receiver, the data gets decoded only after the STA flag is raised.

4. STO (Stop Sequence): It also occurs once and has the the sequence("0000 0011 0000 0011 0000 0110 0000 0110"), which indicates the end of a data frame.

Both the transmitter and receiver uses 2-bit address select signals for programming control registers, reading/writing data and reading status register. Table 2. lists the address mapping of Transmitter/Receiver. These register address along with active low output enable(oe_n) and write enable(we_n) configures the read/write mode for the register addressed.

Preamble	STA	DD along with CRC32	STO
----------	-----	---------------------	-----

Table 1: IrDA Packet format

Register Address	Mode	Description
00	X	No Operation
01	R/W	Configuration Register
10	W	Write(Tx) data FIFO
11	R	Status Register

Table 2: IrDA Transmitter Register Address Mapping

Register Address	Mode	Description
00	X	No Operation
01	R/W	Configuration Register
11	W	Read(Rx) data FIFO
10	R	Status Register

Table 3: IrDA Receiver Register Address Mapping

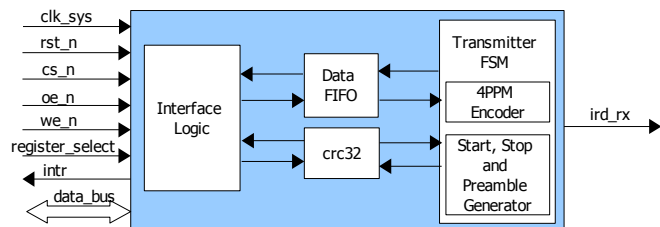


Fig 2: Block diagram of IrDA Transmitter system

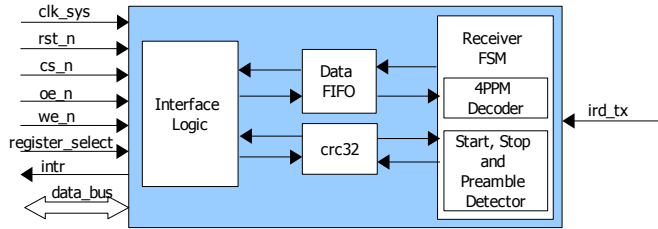


Fig 3: Block diagram of IrDA Receiver system

Table 4.1, Table 4.2, Table 4.3 and Table 4.4 lists the description of configuration and status register bit fields. Transmitter starts data transaction as soon as TXSTART bit asserted. Receiver continuously scans for Preamble and START patterns on ird_tx line. Transmit and Receive data are sliced in to 32-bit length. Configuration register bits {0 to 3} specify length of data transaction in multiples of 32-bit data block. Example: "1000" specifies data length of 8 such 32-bit words. Tx Write FIFO and Rx Read FIFO are of 32x32-bit each. This allows data transaction of maximum of 1024 bits of data per packet.

Transmitter/Receiver Configuration Register:

- TXSTART : Active High, Start data transaction
- INTRCLR : Clear Interrupt
- INTREN : Enable Interrupt
- FIFORST – Reset FIFO
- BLKLEN{4:0} : Block Length

Transmitter/Receiver Status Register:

- TX/RXBUSY : Transmitter/Receiver Busy
- RD EMPTY/HFULL/FULL and WR EMPTY/HFULL/FUL denote Read and Write FIFO empty, half full and full respectively.
- PAERR, STAERR, STOERR and CRCERR denote Preamble Error, Start Error, Stop Error and CRC error respectively.

Fig 4 illustrates the timing diagram of Read Data/Configuration/Status register of IrDA Receiver. Fig 5 illustrates the timing diagram of Write Data/Configuration register of IrDA Transmitter.

CFG21-31	CFG12	CFG11	CFG10	CFG9	CFG8-5	CFG4	CFG3	CFG2	CFG1	CFG0
X	TXSTART	INTRCLR	INTREN	FIFORST	X	BLKLEN4	BLKLEN3	BLKLEN2	BLKLEN1	BLKLEN0

Table 4.1: Transmitter Configuration Register

STS4-31	STS3	STS2	STS1	STS0
X	WRFULL	WRHFULL	WREEMPTY	TXBUSY

Table 4.2: Transmitter Status Register

CFG21-31	CFG11	CFG10	CFG9	CFG8-5	CFG4	CFG3	CFG2	CFG1	CFG0
X	INTRCLR	INTREN	FIFORST	X	BLKLEN4	BLKLEN3	BLKLEN2	BLKLEN1	BLKLEN0

Table 4.3: Receiver Configuration Register

STS8-31	STS7	STS6	STS5	STS4	STS3	STS2	STS1	STS0
X	PAERR	STAERR	STOERR	CRCERR	RDFULL	RDHFULL	RDEEMPTY	RXBUSY

Table 4.2: Receiver Status Register

Pin	Direction	I/O Bank	Voltage	Description
clk_sys	In	Any of 0, 1, 2, 3	2.5/3.3V	System clock(global clock)
rst_n	In	Any of 0, 1, 2, 3	2.5/3.3V	Active low system reset
cs_n	In	Any of 0, 1, 2, 3	1.8/2.5/3.3V	Active low Chip Select
oe_n	In	Any of 0, 1, 2, 3	1.8/2.5/3.3V	Active Low Read Enable
we_n	In	Any of 0, 1, 2, 3	1.8/2.5/3.3V	Active Low Write Enable
register_select[1:0]	In	Any of 0, 1, 2, 3	1.8/2.5/3.3V	Register address
Intr	Out	Any of 0, 1, 2, 3	1.8/2.5/3.3V	IrDA Transmit Interrupt
data_bus[31:0]	Inout	Any of 0, 1, 2, 3	1.8/2.5/3.3V	Input/Out pin for data bus
ird_rx/ird_tx	Out/In	Any of 0, 1, 2, 3	3.3V	IR device data input(Tx)/output(Rx)

Table 4: Pin Description For Transmitter/Receiver

Design	Device	Logic Cells	IO Cells
Transmitter	iCE65L04-UCB284	551	41
Receiver	iCE65L04-UCB284	630	41

Table 3: Resource Utilization

Conclusion

This design example demonstrates the implementation of a IrDA 4PPM Transceiver using iCE FPGAs. iCE FPGA's very low power capabilities makes iCE FPGAs an obvious choice for implementing a IrDA FIR Transceiver for battery operated compact/handheld devices like PDAs, cellular phones etc..

Disclaimer

SiliconBlue is providing this document, design example, or information "as is." SiliconBlue does not ensure that this implementation or information is void of any claims of infringement. As a reader / implementer, you are responsible for obtaining any rights you may require for your implementation. SiliconBlue also does not warranty the fitness of this design example to be readily implemented in any specific context. It is your sole responsibility to validate this design example and its correctness while implementing it for your requirement.

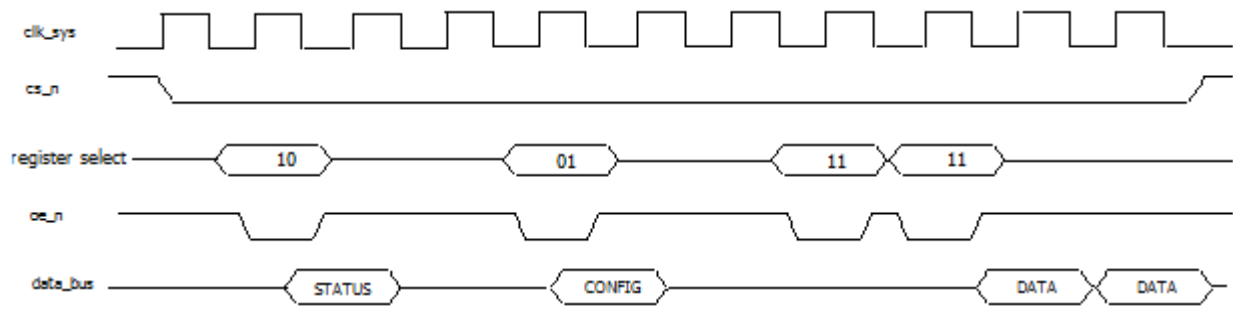


Fig 4. Timing Diagram: Read status, configuration and data

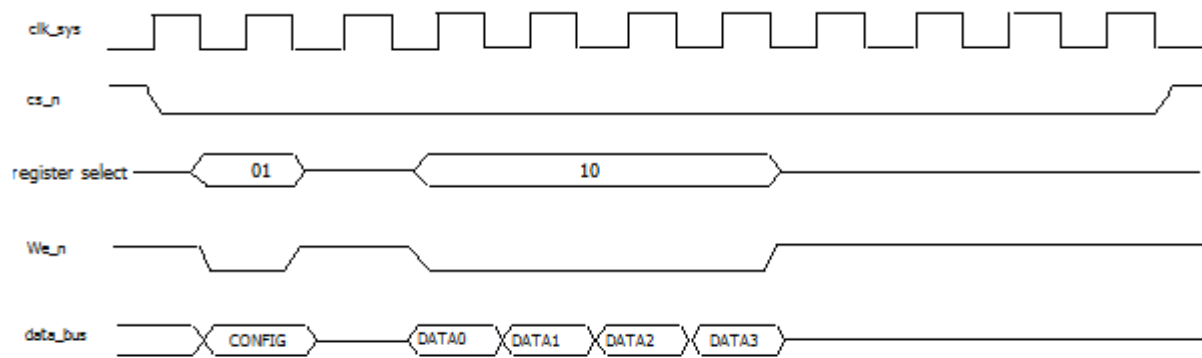


Fig 5. Timing Diagram : Write configuration and data