Abstract: An implementation of Manchester coding is being described in this paper. Manchester coding technique is a digital coding technique in which all the bits of the binary data are arranged in a particular sequence. The Intersil HD-15530 is a high performance CMOS device intended to service the requirements of MIL-STD-1553 and similar Manchester II encoded, time division multiplexed serial data protocols. This LSI chip is divided into two sections, an Encoder and a Decoder. These sections operate completely independent of each other, except for the Master Reset functions. This circuit meets many of the requirements of MIL-STD- 1553. The Encoder produces the sync pulse and the parity bit as well as the decoding of the data bits. The Decoder recognizes the sync pulse and identifies it as well as decoding the data bits and checking parity. This integrated circuit is fully guaranteed to support the 1MHz data rate of MIL-STD-1553 over both temperature and voltage. It interfaces with CMOS, TTL or N channel support circuitry. The HD-15530 can also be used in many party line digital data communications applications, such as an environmental control system driven from a single twisted pair cable of fiber optic cable throughout the building. The functions of the encoder section of the MED include a microprocessor interface, parallel to serial conversion, frame generation, and NRZ to Manchester encoding. This circuitry can run very fast since it does not require a high-frequency clock. The frame format used is similar to that of a UART. The Manchester decoder limits the maximum frequency of operation of the MED, since it uses a high-frequency clock. The receiver circuitry is more complex, since clock recovery and center sampling is done. Additional receiver functions are frame detection, decoding of Manchester to NRZ, serial to parallel conversion, and a microprocessor interface.

Keywords: Manchester coding, Encoder, Decoder, NRZ, Moore’s law, UART, clock frequency

I. INTRODUCTION

Manchester coding technique is a digital coding technique in which all the bits of the binary data are arranged in a particular sequence. Here a bit ‘1’ is represented by transmitting a high voltage for half duration of the input signal and for the next halftime period an inverted signal will be send. When transmitting ‘0’ in Manchester format, for the first half cycle a low voltage will send, and for the next half cycle a high voltage is send. The advantage of Manchester coding is that, when sending a data having continuous high signals or continuous low signal (e.g.: 11110000), it is difficult to calculate the number of Is and Os in the data. Because there is no transition from low to high or high to low for a particular time period (Here it is 4 x T, T is the time duration for a single pulse). The detection is possible only by calculating the time duration of the signal. But when we code this signal in Manchester format there will always be a transition from high to low or low to high for each bit. Thus for a receiver it is easier to detect the data in Manchester format and also the probability for occurrence of an error is very low in Manchester format and it is a universally accepted digital encoding technique.

II. MANCHESTER ENCODER AND DECODER

Manchester coding technique is a digital coding technique in which all the bits of the binary data are arranged in a particular sequence. Here a bit ‘1’ is represented by transmitting a high voltage for half duration of the input signal and for the next halftime period an inverted signal of the original signal (i.e., ‘0’) will send. When transmitting ‘0’ in Manchester format, for the first half cycle a low voltage will send, and for the next half cycle a high voltage will send.

The advantage of Manchester coding is that, when sending a data having continuous high signals or continuous low signal (e.g.: 11110000), it is difficult to calculate the number of Is and Os in the data. Because there is no transition from low to high or high to low for a particular time period (Here it is 4 x T, T is the time duration for a single pulse). The detection is possible only by calculating the time duration of the signal. But when we code this signal in Manchester format there will always be a transition from high to low or low to high for each bit. Thus for a receiver it is easier to detect the data in Manchester format and also the probability for occurrence of an error is very low in Manchester format and it is a universally accepted digital coding technique.

In this paper a Manchester encoding and decoding operations are achieved by VHDL coding technique. All the codes are written with reference to HD15530.
CMOS Manchester encoder and decoder IC. HD15530 IC contains a Manchester encoder and decoder blocks.

III. HD15530 CMOS MANCHESTER ENCODER AND DECODER

The Intersil HD-15530 is a high-performance CMOS device intended to service the requirements of MIL-STD-1553 and similar Manchester II encoded, time-division multiplexed serial data protocols. This LSI chip is divided into two sections, an Encoder and a Decoder. These sections operate completely independent of each other, except for the Master Reset functions. This circuit meets many of the requirements of MIL-STD-1553.

The Encoder produces the sync pulse and the parity bit as well as the encoding of the data bits. The Decoder recognizes the sync pulse and identifies it as well as decoding the data bits and checking parity. This integrated circuit is fully guaranteed to support the 1 MBits data rate of MIL-STD1553 over both temperature and voltage. It interfaces with CMOS, TTL or N channel support circuitry, and uses a standard 5V supply.

The HD-15530 can also be used in many party line digital data communications applications, such as an environmental control system driven from a single twisted pair cable of fiber optic cable throughout the building.

IV. ENCODER OPERATION

The Encoder requires a single clock with a frequency of twice the desired data rate applied at the SEND CLOCK input. An auxiliary divide by six counters is provided on chip which can be utilized to produce the SEND CLOCK by dividing the DECODER CLOCK.

The Encoder’s cycle begins when ENCODER ENABLE is high during a falling edge of ENCODER SHIFT CLOCK. This cycle lasts for one word length or twenty ENCODER SHIFT CLOCK periods. At the next low-to-high transition of the ENCODER SHIFT CLOCK, a high SYNC SELECT input actuates command sync or a low will produce data sync for the word. When the Encoder is ready to accept data, the SEND DATA output will go high and remain high for sixteen ENCODER SHIFT CLOCK periods. During these sixteen periods the data should be clocked into the SERIAL DATA input with every high-to-low transition of the ENCODER SHIFT CLOCK so it can be sampled on the low-to-high transition. After the sync and Manchester II coded data are transmitted through the BIPOLAR ONE and BIPOLAR ZERO outputs, the Encoder adds on an additional bit which is the parity for that word.

If ENCODER ENABLE is held high continuously, consecutive words will be encoded without an interframe gap. ENCODER ENABLE must go low by time as shown to prevent a consecutive word from being encoded.

At any time a low on OUTPUT INHIBIT input will force both bipolar outputs to a high state but will not affect the Encoder in any other way. To abort the Encoder transmission a positive pulse must be applied at MASTER RESET.
Anytime after or during this pulse, a low-to-high transition on SEND CLOCK clears the internal counters and initializes the Encoder for a new word.

The decoded data available at SERIAL DATA OBT is in NRZ. After all sixteen decoded bits have been transmitted the data is checked for odd parity. A high on VALID WORD output indicates a successful reception of a word without any Manchester or parity errors. At this time the Decoder is looking for a new sync character to start another output sequence.

VALID WORD will go low approximately 20 DECODER SHIFT CLOCK periods after it goes high if not reset low sooner by a valid sync and two valid Manchester bits as shown.

At any time in the above sequence a high input on DECODER RESET during a low-to-high transition of DECODER SHIFT CLOCK will abort transmission and initialize the Decoder to start looking for a new sync character.

**V. DECODER OPERATION**

The Decoder requires a single clock with a frequency of 12 times the desired data rate applied at the DECODER CLOCK input. The Manchester II coded data can be presented to the Decoder in one of two ways.

The BIPOLAR ONE and BIPOLAR ZERO inputs will accept data from a comparator sensed transformer coupled bus as specified in Military Spec 1553. The UNIPOLAR DATA input can only accept non-inverted Manchester II coded data. (E.g. from BIPOLAR ONE OUT of an Encoder through an inverter to Unipolar Data Input).

The Decoder is free running and continuously monitors its data input lines for a valid sync character and two valid Manchester data bits to start an output cycle. When a valid sync is recognized, the type of sync is indicated on COMMAND/DATA SYNC output. If the sync character was command sync, this output will go high and remain high for sixteen DECODER SHIFT CLOCK periods, otherwise it will remain low. The TAKE DATA output will go high and remain high while the Decoder is transmitting the decoded data through SERIAL DATA OUT.

VI. OVERVIEW OF PROGRAMMING AND SIMULATION

This system was realized in VHDL language using MODEL SIM Software.
VHDL Implementation of Manchester Encoder and Decoder

A digital system can be represented at different levels of abstraction. This keeps the description and design of complex systems manageable.

The highest level of abstraction is the behavioral level that describes a system in terms of what it does (or how it behaves) rather than in terms of its components and interconnection between them.

A behavioral description specifies the relationship between the input and output signals. The structural level, on the other hand, describes a system as a collection of gates and components that are interconnected to perform a desired function.

A structural description could be compared to a schematic of interconnected logic gates. It is a representation that is usually closer to the physical realization of a system. VHDL allows one to describe a digital system at the structural or the behavioral level. The behavioral level can be further divided into two kinds of styles: Data flow and Algorithmic. The dataflow representation describes how data moves through the system. This is typically done in terms of data flow between registers (Register Transfer level). The data flow model makes use of concurrent statements that are executed in parallel as soon as data arrives at the input. On the other hand, sequential statements are executed in the sequence that they are specified. VHDL allows both concurrent and sequential signal assignments that will determine the manner in which they are executed.

VHDL uses reserved keywords that cannot be used as signal names or identifiers. Keywords and user-defined identifiers are case insensitive. Lines with comments start with two adjacent hyphens (--) and will be ignored by the compiler. VHDL also ignores line breaks and extra spaces.

VHDL is a strongly typed language which implies that one has always to declare the type of every object that can have a value, such as signals, constants and variables.

Once an entity is modelled it needs to be validated by a VHDL system. Once the sample circuit is completed, design is to be simulated to verify that it actually does what it is intended to do. Test benches apply some sequence of inputs to the circuit being tested (the unit under test, or UUT) so that its operation can be observed in simulation.

A test bench must consist of a component declaration corresponding to the UUT, and a description of input stimulus being applied to the UUT. Thus the test bench can be defined as virtual environment which is used to verify the correctness or soundness of the hardware model.

ModelSim is a verification and simulation tool for VHDL, Verilog, System Verilog, and mixed language designs. Mentor Graphics was the first to combine single kernel simulator (SKS) technology with a unified debug environment for Verilog, VHDL, and System C.
The combination of industry-leading, native SKS performance with the best integrated debug and analysis environment make ModelSim the simulator of choice for both ASIC and FPGA design. The best standards and platform support in the industry make it easy to adopt in the majority of process and tool flows.

In ModelSim, all designs are compiled into a library. You typically start a new simulation in ModelSim by creating a working library called "work". "Work" is the library name used by the compiler as the default destination for compiled design units. After creating the working library, you compile your design units into it.

The ModelSim library format is compatible across all supported platforms. You can simulate your design on any platform without having to recompile your design. With the design compiled, you load the simulator with your design by invoking the simulator on a top-level module (Verilog) or a configuration or entity/architecture pair (VHDL).

Assuming the design loads successfully, the simulation time is set to zero, and you enter a run command to begin simulation. If you don’t get the results you expect, you can use ModelSim’s robust debugging environment to track down the cause of the problem.

The Wave window allows you to view the results of your simulation as HDL waveforms and their values. The Wave window is divided into a number of window panes. All window panes in the Wave window can be resized by clicking and dragging the bar between any two panes.

If you close the Wave window, any configurations you made to the window (e.g., signals added, cursors set, etc.) are discarded. However, you can use Save Format command to capture the current Wave window display and signal preferences to a DO file. You open the DO file later to recreate the Wave window as it appeared when the file was created.

CONCLUSION

The paper “VHDL Implementation of Advanced Manchester Encoder and Decoder” presented here gives an easy and less expensive way to implement Manchester coding system. First the different components that constitute the coder have been designed with required inputs, outputs & control signals. The interface between the designed components has been formulated. This was then realized in VHDL language using MODEL SIM Software.

The design was thoroughly simulated to test all the instructions in VHDL simulator software. As the coding system has high accuracy, the system gives more accuracy and efficiency in data sending. This coding system is a universally accepted digital coding system. Also it provides more security for data transmission.

REFERENCES

[9] Technical papers from VSSC Library
[10] www.aldec.com

★★★