FPGA Implementation of a PCM Decoder for Range Telemetry System with Block Code

Abdelmonem A. S.

Assistant Professor, ECE Dept., Elshorouk Academy, Cairo, Egypt

Hussein S.T.

Associated Professor, Military Technical College, Cairo, Egypt

Essam A.E

Graduate Student, ECE Dept., Military Technical College, Cairo, Egypt.

Abstract: *Pulse Code Modulation (PCM) Telemetry is a way of acquiring data in one location, converting the data samples to digital words, encoding the data in a serial format, and transmitting it to another location for decoding and analysis. Field programmable circuits now have capacities that enhance the telemetry systems capabilities. This paper demonstrates the simulation and implementation of a dual coding algorithm for PCM decoder. The hardware implementation is based on Industry Standard Architecture (ISA) interface. Dual coding capability that switches between NRZ and Manchester coding algorithm is adopted to enhance security performance. Small size and high performance are obtained using ACTEL FPGA. Simulation results are compared with practical measurements of the designed decoder.*

Key words — Decoder, PCM, FPGA, ACTEL and IS**.**

1. INTRODUCTION

Telemetry is generally defined as a communication process by which measurements are made and/or data are collected at remote or inaccessible points, and transmitted for monitoring, display, and recording. Telemetry has become an enabling technology for large complex systems and applications such as agriculture [10], defense [2, 5, 6, 18], motor racing [16], medicine [4, 7, 12], and others [9, 17].

PCM systems transmit data as a serial stream of digital words consisting of logic ones and zeros that are easy to transmit and record. The PCM encoder samples the input data and inserts the data words into a PCM frame. Words are assigned specific locations in PCM frame so the decoder can recover the data samples corresponding to each input signal. Over the years, a number of PCM codes have been designed to represents logic one and zero level while achieving the greatest performance for a given application [1, 8, 11, 15].

PCM and FPGA allow easy configuration of the designed telemetry solution and easy customization of all aspects of the communication stream for the given application. It also directly enables the integration of wide range of sensors and data sources by implementing the required digital PCM decoder right on the chip [3, 13].

Unlike general telemetry systems that utilize single decoding algorithm in their operations like NRZ, Manchester ….. etc [14], the first contribution of this work is to develop a PCM decoder with dual decoding capability that switches between NRZ and Manchester coding algorithms based on a programmable control signal. Therefore, the designed PCM is vital for those highly secure applications. The second contribution of this work is the simulation and implementation of the proposed PCM decoder.

The decoder consists of three major modules: Data extraction module, Operation timer module, and Ground computer – interface module. The hardware implementation is based on Industry Standard Architecture Interface. The proposed decoder is responsible to extract channel timing and data, apply the block codes to detect and correct one bit every four bits. One programmable ACTEL FPGA is used for the implementation of the proposed decoder.

This work begins by introducing the architecture of the proposed PCM decoder in section 2, where the functions of the individual modules are explained. In section 3, the experimental results and performance of the proposed system are analyzed. Finally, in section 4, the conclusion is presented.

2. SYSTEM ARCHITECTURE

Refer to Figure1. The proposed decoder is divided into three main modules:

1. Data extraction module.
2. Operation timer module.
3. Ground computer- interface module

In the following subsections, each module is explained.

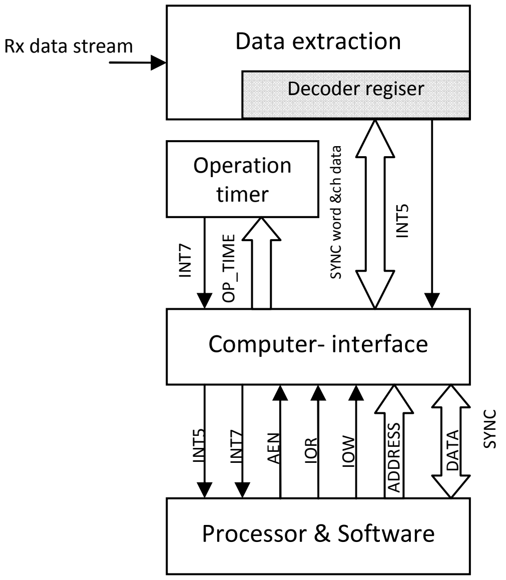

Figure 1. Decoder Architecture

2.1 DATA EXTRACTION MODULE IMPLEMENTATION

As illustrated in Figure 2. The received data stream is fed to a 16-bit serial-in parallel-out shift register where the output is continuously compared with a preset SYNC word with the same length. The SYNC word is previously preset in the decoder SYNC register via the decoder software. Whenever the 16-bit comparator locks on the same SYNC word, it outputs SYNC flag which initiates the timing and byte locator circuitry. This circuitry is responsible for detecting the start and stop bits that are sent with each channel data, thus recognizing the useful channel data and sending a flag signal to byte latch register so that the useful data will be ready to be sent whenever the channel data is selected. Another data type, 2 time bytes, is available through a synchronized 16-bit timer which feeds 2 bytes of the equivalent frame data. Two bytes of status data are also available from different timing and informative points within the decoder hardware are located to be read by the software whenever needed. A five input multiplexer is fed with the data bytes mentioned before, the selected byte to be read by the software is selected by means of direct addressing through the ISA bus interface, where the software address the required data byte to be read. Hamming code error detection and correction is used in the decoder software based on the following formula:

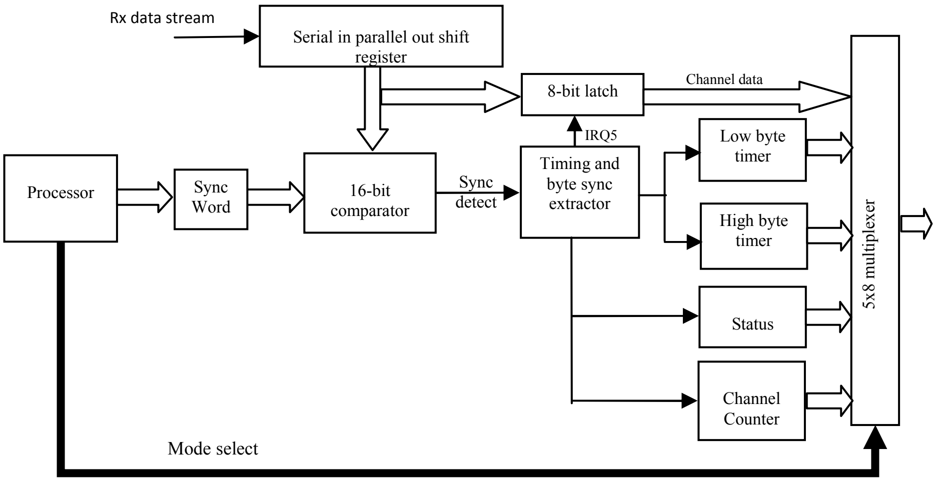

Figure 2 : Data Extraction Block Diagram

$$(n,k) = (2^m - 1, 2^m - 1 - m)$$

Where k is the number of information bits used to form an n bits codeword. The (7, 4) hamming code is used in this work, such that (m=n-k) is the number of parity symbols and m = 3. These binary codes have a minimum distance of 3, and they are capable of correcting all single error or detecting all combinations of two or fewer errors within the block. This application describes a technique for detecting and correcting 1-bit error in a block of data ranging from 1 to 4 bits in length. The technique applied is a modified version of a Hamming code and has been implemented entirely in C ++. Additional functions have been provided to program and read in one Disk on Chip (DOC) in microprocessor using the error encoding and decoding algorithms.

2.2 OPERATION TIME MODULE IMPLEMENTATION

As illustrated in Figure 3. The operation time module is defined as the maximum expected time for the required reception. This time is stored in the OP_TIME register found in the decoder circuitry by means of the operating software. The OP_TIME consists of 18 bits; the 16 least significant bits comprise the required time to quit the decoder operation when it is reached, while the two most significant bits comprise the time interval to increment the running 16-bit timer which will be compared with the 16 least significant bits. The 16-bit register can count to a maximum of 65536 then it will reset, thus a time scale (2 MSB) must be used to reduce the precision timer input clock such that the mission duration lies within the 16 bits count margin. The clock divider utilizes 2 bits to select between different frequency divider settings to cope with the required mission duration. This structure gives the flexibility to use the decoder to small or relatively long time durations according to the nature of the monitored operation. The output of this block is an OP_TIME flag which starts with the decoder operation through software control, and ends automatically when the preset time is reached where the decoder interrupts the processor to terminate the running processes, close the log files and secure the logged data.

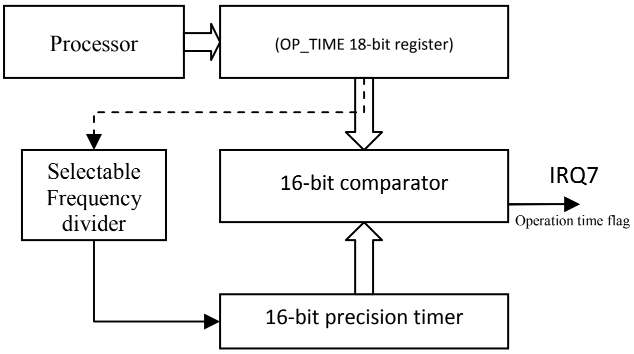

Figure 3: Operation Timer Block Diagram

2.3 PC INTERFACING MODULE

The bus interfaces 12-bit ISA address bus and 8-bit ISA data bus, the memory map of the decoder hardware with the associated function is listed as shown in the next table.

Figure 4 illustrates the decoder ISA bus interface structure. Based on the address map of the registers, the software can control and fully operate the decoder hardware. The control word is stored in the control register where each bit is responsible for a preset operation. The control action includes:

1. Resetting the channel receiver
2. Resetting the frame number
3. Start/stop operation
4. Enable/clear interrupt requests 5 and 7.

Moreover status words can be read during operation, reflecting the current value for different locations inside the hardware circuit, this status word can help the programmer to navigate through the decoder operation smoothly.

Address	$\overline{\text{AEN}}$	$\overline{\text{IOR}}$	$\overline{\text{IOW}}$	Function
0x300	0	1	0	Write control words and clear interrupts
0x301	0	1	0	Write operation time low byte
0x302	0	1	0	Write operation time high byte
0x303	0	1	0	Write operation time resolution (2 bits)
0x304	0	1	0	Write SYNC word low byte
0x305	0	1	0	Write SYNC word high byte
0x300	0	0	1	Read status word from decoder
0x301	0	0	1	Read channel number
0x302	0	0	1	Read channel data
0x303	0	0	1	Read frame number low byte
0x304	0	0	1	Read frame number high byte

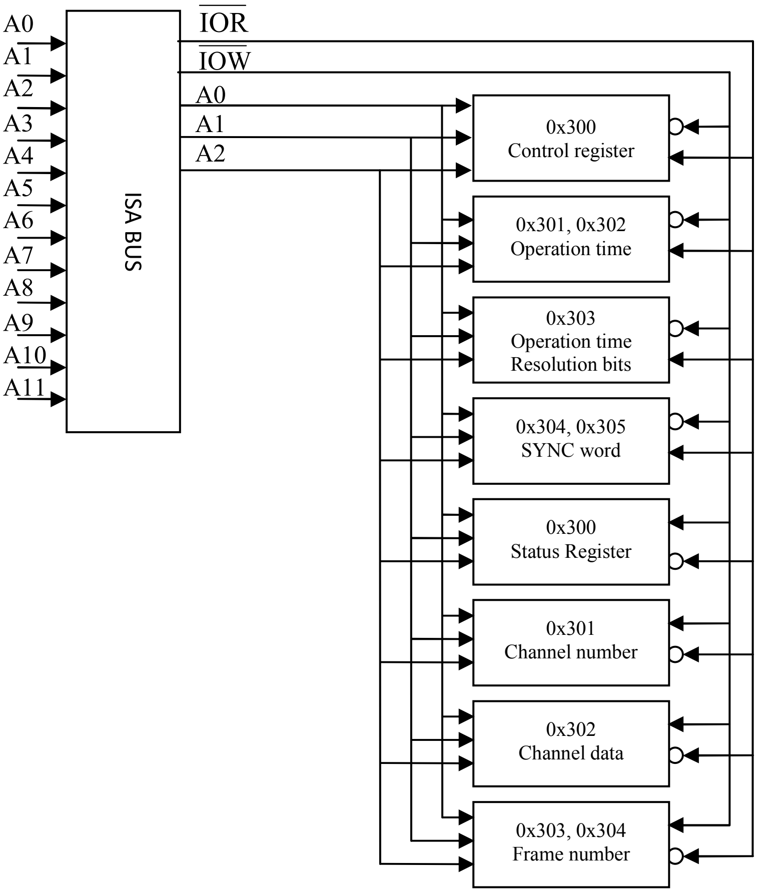

Figure4: Decoder's ISA Bus Interface Structure

2.4 THE ROLE OF THE EMBEDDED PC

The Ground Computer (GC) is interfaced to the decoder block through the ISA interface layer where the GC mainly controls every action in the decoder. Primarily, the GC extracts channels included in the received telemetry frame. The process of extraction begins by comparing the received data bits with the previously stored sync word (0x90 and 0xeb) which marks the beginning of each frame. Once the mark is captured, the 16-bit comparator initiates the timing and byte sync module. The timing and byte sync module generates the IRQ5 signal that controls the timing of each of the 8-bit channels. Thus, the received channels bits are extracted and stored under control of the 8-bit latch. Since the frame time lasts for 5×10-6 (bit duration) ×11 (bits/channel) ×32 (number of channels /frame) =

1.76 msec, and each frame is sent every 10 msec, the expected experiment duration won't exceed 5 seconds. This five seconds experiment duration is detected through comparing the predetermined experiment time with the value that is set in the 16-bit comparator of the operation timer module. As long as the end of the experiment is not deselected by the comparator, the process of reading and storing channel bits will be repeated. The operation time flag is used to initiate the IRQ7.

3. EXPERIMENTAL RESULTS AND ANALYSIS

The multilayer Printed Circuit Board (PCB) of the decoder is designed using the PCAD package. The board is manufactured and tested to carry the decoder Actel chip with PC ISA interface as shown in Figure 5. Each block in the decoder is simulated separately using associated software with the ACTEL design package. Moreover, the whole decoder system is simulated in a similar way. The design is synthesized on an A1280A chip with 62% for the decoder. The timing signals due to simulation process of the decoder are shown in Figure 6. The FR_INP is the input frame signal to the decoder. The AEN, IOR, IOW, ADDR and IRQ7 are ISA bus interface signals to manage data between PC and input frame. The CH_CLK, CH_GATE are temporary signals used in design process where DATA signal is the output channels.

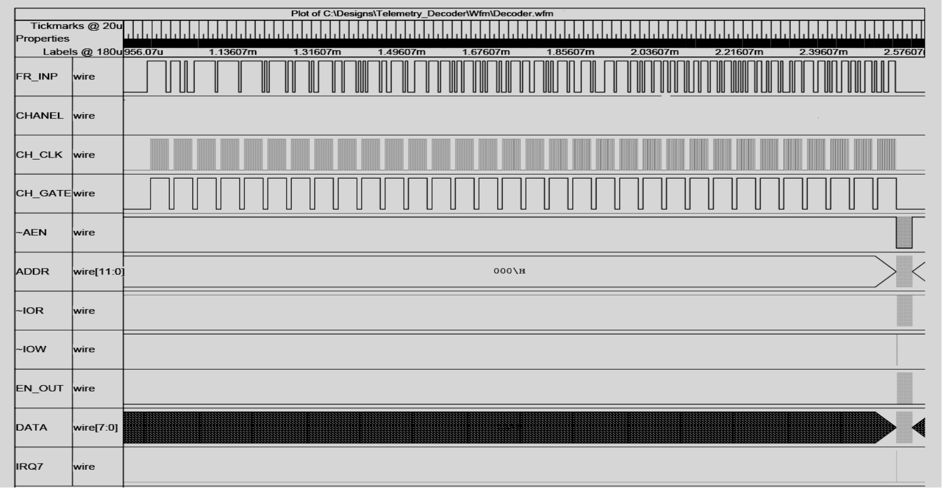

Figure 5: Decoder PCB Layout And Netlist

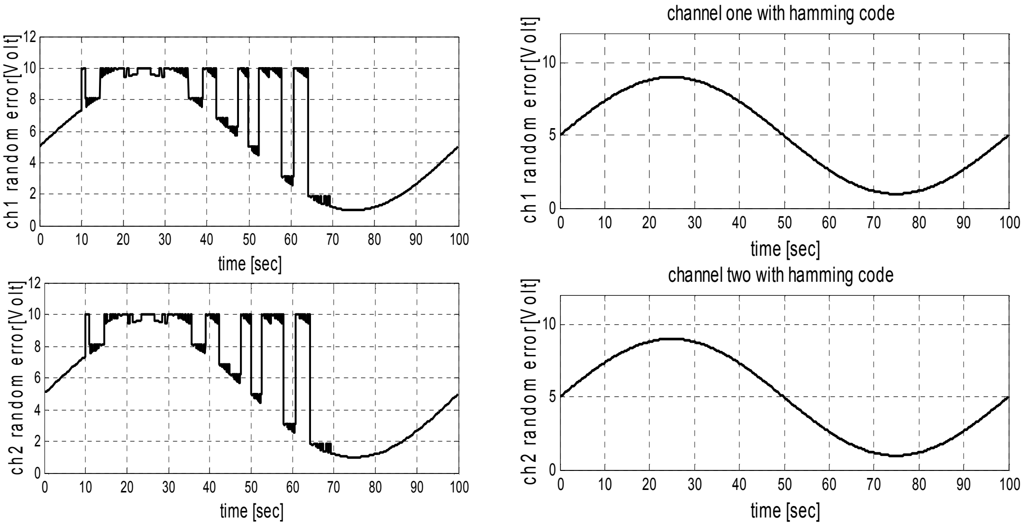

Figure 6: The Simulated Decoder With Error And The Corrected Ones

Experimental results are obtained by utilizing Logic analyzer and Oscilloscope devices to verify that the system is functioning properly. Figure 7 represents the Manchester decoding of the input frame signal Enc_MAN_received and associated frame_clock to get the extracted G_clock clock and G_data_pll channel data. Also, Figure8 presents the header of the detected frame with start, stop and sync patterns bits. The output of the hamming code error detection and correction algorithm is presented in Figure 9 and Figure 10. The left column of the figure represents data with random error for two channels before correction and right one illustrates correction of the two error bits frame data. The decoder is tested using all possible combinations of the 8-bits values of the input frame channels. A high level language program is written in C++ to manage the data flow between the PC and the decoder. The flow chart of the testing algorithm is shown in Figure11 that depends on detecting the two syncs and start receiving data with IRQ 5 occurrences.

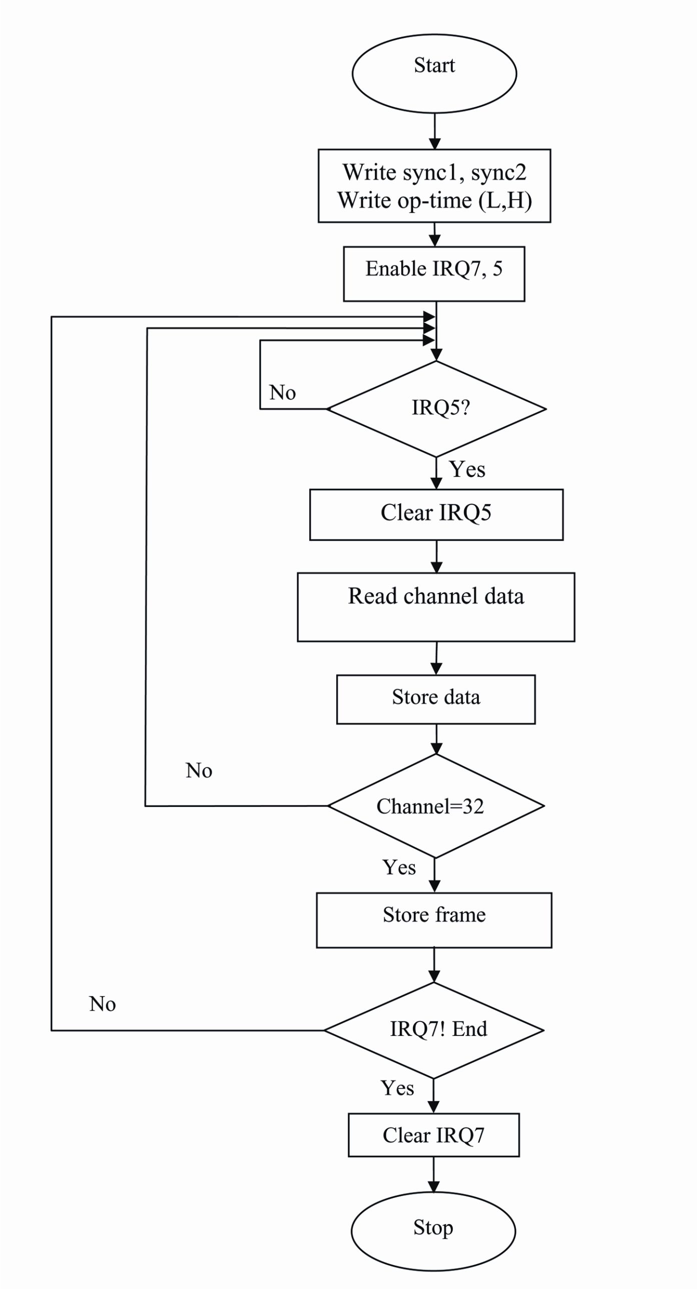

Figure11 : Flow Chart Of The Ground Computer With The Decoder

Simulation and experiment measurements show that the system performs and satisfies the design objectives. The design approach used in this work is considered more reliable than others techniques. The main advantages of this design are using ACTEL FPGA ensure system security, fast and support full module control via the ISA bus PC interface programming. Moreover the approach provides a baud rate of 200Kbps with dual coding Manchester and NRZ channel to cope with AM/FM transmitters for narrow or wide bandwidths.

4. CONCLUSIONS

The developed PCM decoder for ground station utilizing the ISA Interface are developed and implemented on ACTEL chip hardware. The design is synthesized on an A1280A chip with 62% chip occupation for the decoder. Using this kind of chips results in a compact and reliable design of PCM decoder for telemetry, systems utilized for tracking purposes. The ISA Interface is employed to enable data transfer and control between the ACTEL decoder and the central processor. A software program is developed and implemented in high level language to fully control all operations of the decoder modules. The overall system is tested and validated in three different ways: simulation, lab test and field test. The field test is performed in wiring and wireless situations. The decoded outputs were displayed and it was identical to the transmitted data. The error detection and correction capability is investigated by introducing some errors into the transmitted bit stream and a Hamming decoder for error detection and correction is used successfully to eliminate this error. It is verified that the simulated results coincide with the practical measurements.

References

[1] Bruce, A. and Wooley, D.and Colin F, “An integrated interpolative PCM decoder”, IEEE transactions on communications, vol. 27, (1979).

[2] Chandrasekaran, A. and. Gutshall, J. ,“The design of the communication and telemetry System used by the ionospheric observation nanosatellite formation mission” ,in Proceedings of the 15th Annual AIAA/UTAH State University Conference on Small Satellites, pp. 1—6, (Aug 2001).

[3] Coggins, T.J.and Cirstea, M.N., "FPGA implementation of a fast flex supervisory control and data acquisition (SCADA) remote telemetry unit", Thirty-First Annual Conference of the IEEE Industrial Electronics Society, (2005).

[4] Cook , T.A., "A Custom Microprocessor for Implantable Telemetry Systems", Third Ann. IEEE Symposium. Computer-Based Medical Systems, Computer Society Press, Los Alamitos, Calif.,(1990).

[5] Dapos, W.P., "Telemetry systems and electric gun projectiles", IEEE Transactions on Magnetics, vol.37, Issue 1, (Jan 2001).

[6] Hepner, J.and Borgen, G., " High-G flight experiment of a Field-Programmable Gate Array (FPGA) Pulse Code Modulation (PCM) encoder " ,46th International Instrumentation Symposium., Vol .397, PP 99-108, (2000).

[7] Jeutter, D. C. “Overview of biomedical telemetry techniques,” IEEE. Eng. Med. Biol. Mag., vol. 2, pp. 17–24, 1983.

[8] Michael J. Alexander, G., “New Performance driven FPGA routing algorithms”, IEEE transactions on computer aided design vol. 15, No23, (1996).

[9] Palanivelu, S. and Shanmugam, J., "Design and development of PCM decommutator with PCI interface", 24th Digital Avionics Systems Conference, October 30, (2005).

[10] Pierce, F.J. "Regional and on-farm wireless sensor networks for agricultural systems in Eastern Washington", Regional and on-farm wireless sensor networks for agricultural systems in Eastern Washington, vol. 61 Issue 1, (April 2008).

[11] Quadrini , E. F, "PCM-formatter: a powerful, low cost system for PCM telemetry decoding and quick look support", IEEE Transactions on Nuclear Science, vol. 40, issue 4, pp. 905-908, (1993).

[12] Riley, Joseph L., "Digital Frequency-to-Temperature Converter For Use in Radio Telemetry Systems in Which Temperature is Transmitted Proportional to a Subcarrier Frequency", IEEE Transactions ob Biomedical Engineering, vol. (2007).

[13] Sadhukhan, G.and Sandhu, M., "VLSI based implementation of PCM MUX encoder for range telemetry system ", Proceedings of the IEEE INDICON 2004. First Indian Annual Conference, (2004).

[14] Tien Manh Nguyen; Hen-Geul Yeh, "Behavior of PCM/PM/NRZ receivers in non-ideal channels-the separate effects of imperfect data streams and band limiting channels on performance", IEEE Global Telecommunications Conference, GLOBECOM, vol. 3, 1994.

[15] Trigg,, H. W. , "Burst Retransmission of PCM Telemetry Data", IEEE Transactions on Aerospace and Electronic Systems, vol. AES-2, (1996).

[16] Waldo,J., "Embedded computing and Formula One racing", IEEE Pervasive Computing, volume 4, Issue 3, (2005).

[17] Yakata, H.and Toya, N., "Performance analysis of optical telemetry using synchronous CDMA", Electronics and Communications in Japan, Part 1 (Communications), Vol .88,PP23-32, (2005)

[18] Zetterquist, P., "Development of PCM telemetry and tracking systems at the Swedish Space Corporation " , Journal of the British Interplanetary Society, Vol. 27, Issue 8, PP 590-5 ,(1974).