International Journal of Combined Research & Development (IJCRD)
eISSN:2321-225X; pISSN:2321-2241 Volume: 2; Issue: 5; May -2014

Hardware Design of Safety Arming Mechanism using FPGA

Manoj Kumar Reddy.T.D

ECE Department, Jawaharlal Nehru Technological University, Anantapur, Andhra Pradesh, India.
manoj.deshai@gmail.com,

Analogic Controls India Private Limited

Survey no 23/2,Gundlapochampally,Medchal mandal, Rangareddy Dist, Hyderarabad-500025,
Andhra Pradesh, India www.analogiccontroles.com

Abstract: In this Paper **"Hardware Design of Safety Arming Mechanism using FPGA" involves sequence detection and safety interlock based safety system with different types of sensors like depth measurements, 'g' measurements and the operational sequences which the equipment should follow to have good safety. The FPGA programmed with all these inputs and the sequences are adjusted as per the requirement. VHDL code was developed for performing these activities and loaded in to the FPGA. An UART module is also developed and linked with the sequence of operations. The required inputs and outputs are continuously logged and displayed on to the RS-422 port using this developed UART in a defined format.**

***Keywords*— SAM, FPGA, UART, NVRAM,VHDL**

I. INTRODUCTION

Safety Arming Mechanism (SAM) is a modular sub-system of safety device to ensure safety of the device during handling, transport, storage, launch phases of the device. SAM consists of An electronic circuitry and an Arming Mechanism. The device consists of required hardware and software. Necessary Safety Interlocks are provided in H/W and S/W which are removed at different stages in a sequence and finally trigger the further circuits at desired time. Electronic circuitry and an Arming Mechanism are interfaced by means of a connector terminated on a rigid flexi PCB. The system operational logic will be designed as per the logic flow chart Arming Mechanism essentially consists of a solenoid based mechanism with dual redundancy along with other mechanical components Arming Safety rod, a Hydrostatic switch, two DC solenoids, two MEMS Impact switches to sense deceleration on impact. Each DC solenoid will be capable of driving two Interrupters for redundancy.

Arming Mechanism will have a provision to check alignment status of the interrupters. A mechanical safety in the form of a "Safety pin" which locks the plunger of both the solenoids. Before deployment this safety pin is to be removed. A Hydrostatic switch will be provided which will remain initially closed for a below a fixed length and gets opened when it senses a length of more than or equal to that fixed length. In aligned condition, the Interrupters shall be locked and short across the circuit will be removed. Arming Mechanism has a feature such that it can not be assembled in armed condition and will also have indication of alignment of each interrupter .Impact of the order of certain 'g' will be sensed by MEMS based impact sensor.

The UART design is referred from the Xilinx document reference[1] and reference[8]. Xilinx Spatran-3 FPGA datasheet referred from the reference[2].NVRAM datasheet referred from the reference[3].RS-422/485 datasheet referred from reference[4]. Reset IC datasheet referred from the reference[5] .DC-DC converter datasheet referred from the reference[6]. Operational amplifier LM124 referred from the reference[7]. Logic for the circuit is referred from the reference[9] . VHDL programming is referred from the reference[10] .

II. BLOCK DIAGRAM

The block diagram of power supply scheme as shown in.Fig1 . Mainly it consisting of DC-DC converter, voltage regulator. The DC-DC converter converts input 28V to 5V.By using this 5V the voltage regulator converts in to 3.3.V and 1.2V and in turn used by the FPGA. Fig2.refers to the block diagram of hardware of safety arming mechanism consisting of Xilinx-SPARTAN- FPGA, opto-coupler, power-fail logic, 10MHZ clock source, reset IC, NVRAM for data logging, RS-422 transceiver MAX1490B used for UART communication. The explanation of the block diagram is explained as follows. . 28V inputs like DEPTH, OBC-1, OBC-II, OBC-III, IMPACT signals are electrically isolated by the opto-couplers and the output of the opto-coupler is .V signals given to the Schmitt trigger, where pulse shaping is done and these signals are interfaced to FPGA. The VHDL code is implemented in the FPGA and programmed by using JTAG programmer. The internal signals of the code can be monitored on the test points As per the logic the outputs like ENABLE, ARM, STR, CHEN, GATE1, GATE2 are generated by giving the inputs. Fig.3 refers to the block diagram of pulse generation. It consists of relays, pulse transformers. . The outputs CHEN, GATE1 AND GATE2 and STR intern fed to the opto-couplers and relays respectively.

The resultant signals are fed to the pulse transformer and the required operation will be performed.

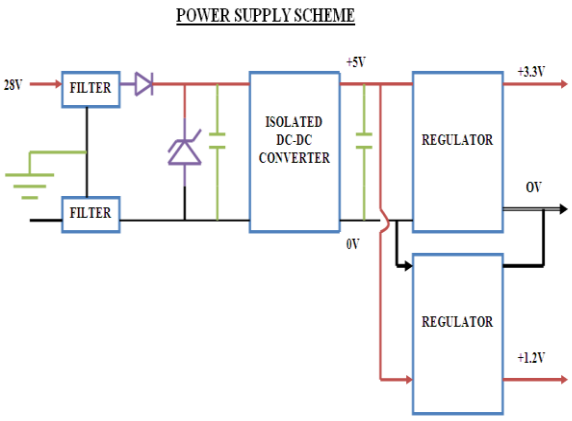


Fig1:Block diagram of the power supply scheme

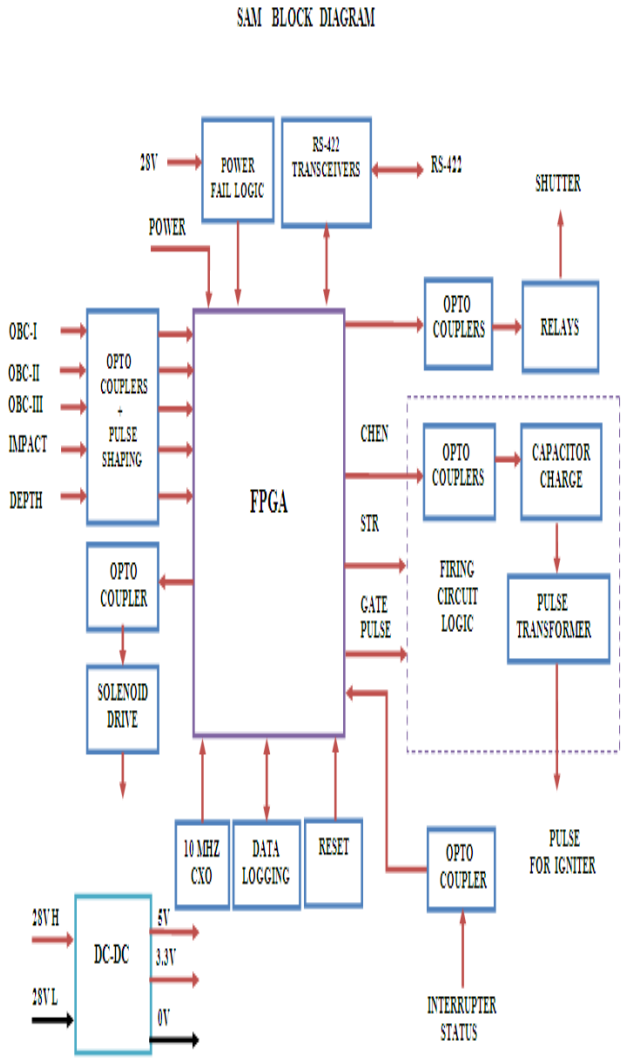


Fig2:Block diagram of the hard ware of SAM

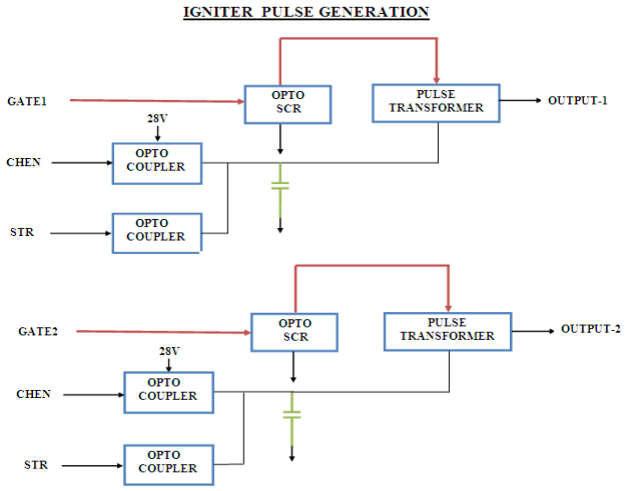


Fig:Block diagram of the igniter pulse generation

The block diagram of Universal Asynchronous Receiver Transmitter (UART) is as shown in the Fig.4.1 The data from data bus is accessed by the transmit data register(TDR). With the transmit control operation, the serial data TxD is shifted out from the transmit shift register (TSR) .Baud clock is generated by Baud generator and same baud clock is used for both transmitter and receiver. The serial data from input RxD is shifted to the Receive shift register(RSD) by using receive control logic and the data is placed on the data bus by using Receive data register(RSR).

UART data out format is as shown in the fig4.2. The sequence is Idle condition , one start bit ,eight data bits(LSB first) followed by one or two stop bits. Considering Fig 2, the block diagram of hardware of safety arming mechanism, the UART is developed in the FPGA, the single ended signals Tx and Rx are converted in to 5v and fed to RS-422 transceiver and the outputs are differential signals and communicated to the external world.RS-422 to USB is used for communicating with the computer. With the use of Hyper terminal the communication can be monitored in the computer. By selecting parameters like Baud rate, data bits, parity, number of stop bits, flow control communication can be established. The Received data can be captured in a file and can be viewed by using HEX pert software tool.

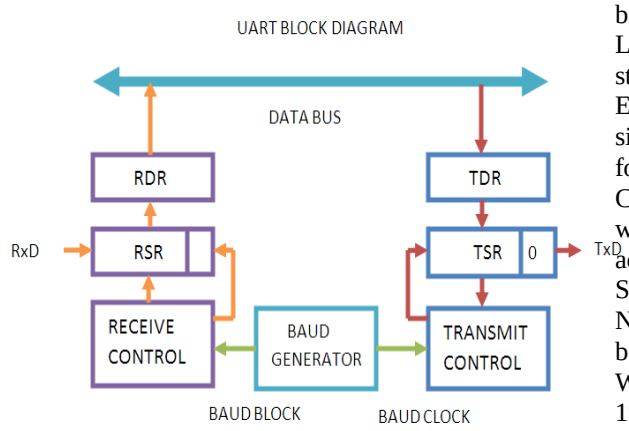


Fig4.1:Block diagram of UART

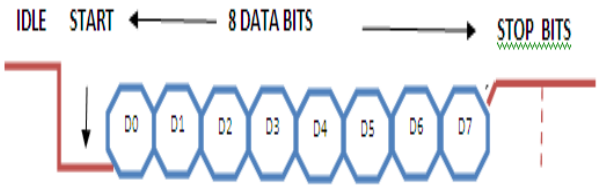


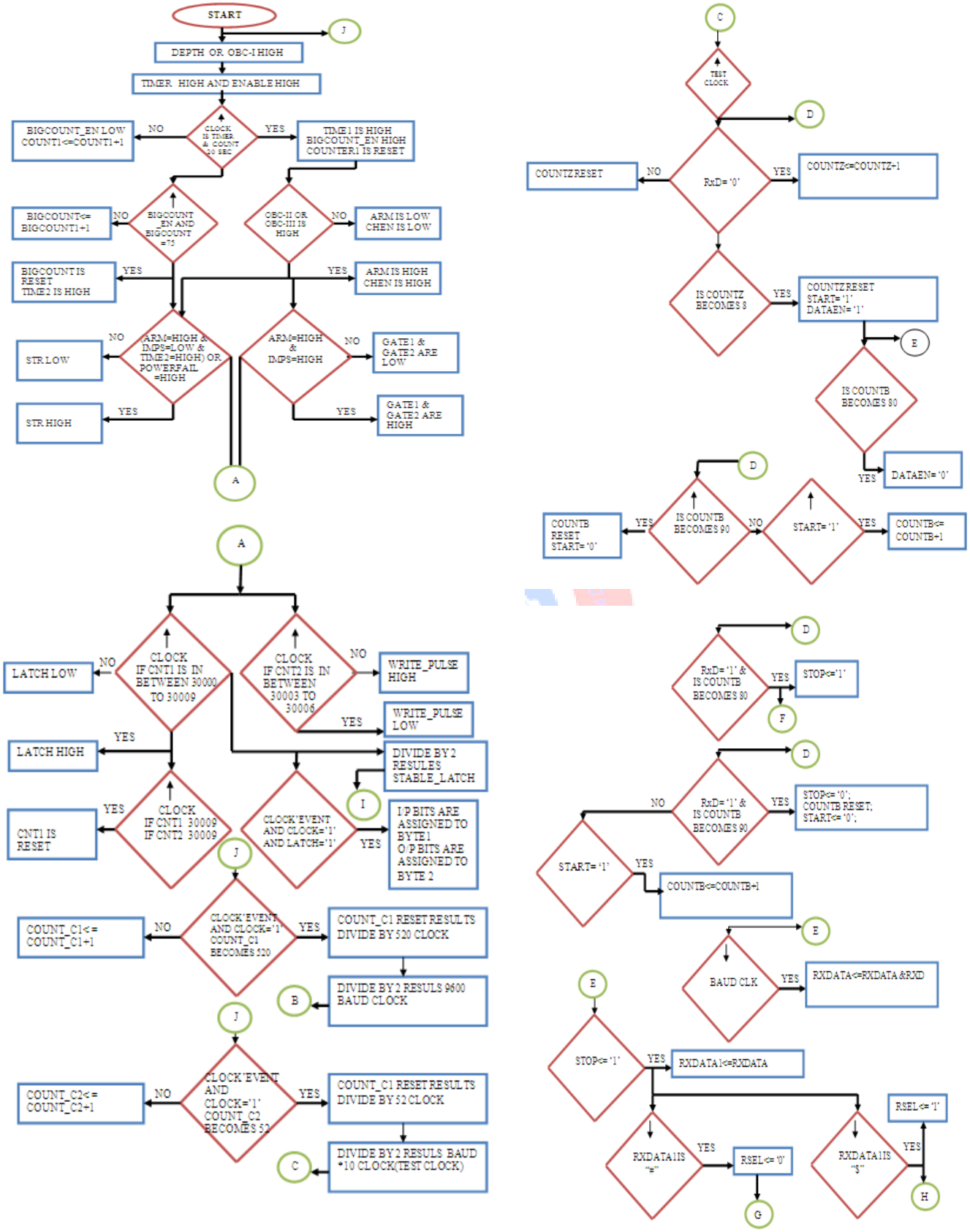
Fig4.2:UART data out format

III. DIGITAL LOGIC USED IN FPGA

The digital logic of this project is explained with reference to the flow chart shown in the Fig 5. Input to the FPGA are from opto-couplers which are DEPTH, OBC-1, OBC-II, OBC-III, IMPACT, POWERFAIL and RESET IC. Outputs are ENABLE, ARM, STR ,CHEN, GATE1,GATE2 will be generated when DEPTH or OBC-1 are given. Timer will start when DEPTH or OBC-1 are present. When TIMER becomes HIGH the counter will start counting, when it reaches to 20 seconds TIMER1 signal is generated. When TIMER1 becomes high and either of OBC-II or OBC-III are present the ARM and CHEN signals are generated otherwise it will be deactivated.TIMER2 will be generated after 25 minutes using the counter. When ARM signal is activated and IMPACT signal is deactivated and timer2 is activated or POWER FAIL becomes high STR will be generated, otherwise deactivated. When ARM signal is activated and IMPACT signal is activated GATE1 and GATE2 will be generated, otherwise deactivated.

LATCH signal is generated for every 3 ms, using this LATCH signal STABLE LATCH will be generated which will be divided by 2 of the LATCH .BYTE1 assigned to Input bits and BYTE2 assigned to Output Bytes, will be done when LATCH occurs with respect to the clock. The data will be stored in NVRAM for every 3 ms. Active low WRITE ENABLE pulse(WE) will be generated when the LATCH signal occurs. Complement of WRITE ENABLE can be used for the OUTPUT ENABLE(OE). A low signal is generated for CHIP SELECT(CS).When RSEL signal is low write operation will be performed, default condition is write mode. 19 bit address will be incremented with respect to LATCH. When STABLE LATCH becomes high input byte will be stored in NVRAM and STABLE LATCH becomes low output byte will be stored in NVRAM. This process repeated for every 3 ms. When RSEL signal is high read operation will be performed, 19 bit address will be incremented with respect to LATCH. Data will be read from the NVRAM.CHIP SELECT (CS) will be low, WRITE ENABLE will be high and OUTPUT ENABLE (OE) will be low during the read operation. A 10 MHZ reference clock has been using for the design.9600 baud rate will be generated with the 10 MHZ, the clock is first divide by 520 and then divided by 2.This can be used for the transmit as well as receive in the UART .

In the UART receive mode the 10 MHZ clock is first divide by 52 and then divided by 2.This test clock is used for the detection of the start bit. Counterz is used for the counting of the test clocks when RxD bit becomes low. When the RxD becomes high counter will be reset. When RxD is low and counterz becomes 8 the start bit is zero generate the signals start and Dataen and the counterz will be reset. When start becomes high, counterb is used to count 80 test clock pulses and Dataen signal becomes low. When count becomes 90 and the RxD becomes high counterb counter will be reset. When counterb becomes 80 the Stop signal becomes high and counterb becomes 90, Stop signal becomes low. When dataen is high, falling edge of baud clock,8 bit SISO shift register 1 to 7 bits are concadded with RxD and assigned to same serial in serial out shift register 0 to 7 bits. When Stop becomes high the shift register output assigned to other shift register. when $ symbol is given on RxD, RSEL becomes high, read process will occur. when # comes on RxD, RSEL becomes low it will be in writing mode RSEL becomes high UART transmission occurs ,start bit(zero) concadded with 8 bit data concadded with one stop bit(one) and assigned to transmit data shift register. When stable latch is high as well as low, LSB bit of transmit data shift register is assigned to TxD. This process carried out with respect to the baud clock.


START
J
DEPTH OR OBC-I HIGH
TIMER HIGH AND ENABLE HIGH
CLOCK IS TIMER & COUNT 20 SEC
NO
BIGCOUNT_EN LOW COUNT1<=COUNT1+1
YES
TIME1 IS HIGH BIGCOUNT_EN HIGH COUNTER1 IS RESET
BIGCOUNT EN AND BIGCOUNT =75
NO
BIGCOUNT<= BIGCOUNT1+1
OBC-II OR OBC-III IS HIGH
NO
ARM IS LOW CHEN IS LOW
YES
BIGCOUNT IS RESET TIME2 IS HIGH
YES
ARM IS HIGH CHEN IS HIGH
(ARM=HIGH & IMPS=LOW & TIME2=HIGH) OR POWERFAIL =HIGH
NO
STR LOW
YES
STR HIGH
ARM=HIGH & IMPS=HIGH
NO
GATE1 & GATE2 ARE LOW
YES
GATE1 & GATE2 ARE HIGH
A
C
TEST CLOCK
D
COUNTZ RESET
NO
RxD= '0'
YES
COUNTZ<=COUNTZ+1
IS COUNTZ BECOMES 8
YES
COUNTZ RESET START= '1' DATAEN= '1'
E
IS COUNTB BECOMES 80
YES
DATAEN= '0'
D
COUNTB RESET START= '0'
YES
IS COUNTB BECOMES 90
NO
START= '1'
YES
COUNTB<= COUNTB+1
A
CLOCK IF CNT1 IS IN BETWEEN 30000 TO 30009
NO
LATCH LOW
YES
LATCH HIGH
CLOCK IF CNT2 IS IN BETWEEN 30003 TO 30006
NO
WRITE_PULSE HIGH
YES
WRITE_PULSE LOW'
DIVIDE BY 2 RESULES STABLE_LATCH
I
CLOCK'EVENT AND CLOCK='1' AND LATCH='1'
YES
I/P BITS ARE ASSIGNED TO BYTE1 O/P BITS ARE ASSIGNED TO BYTE 2
CLOCK IF CNT1 30009 IF CNT2 30009
YES
CNT1 IS RESET
J
CLOCK'EVENT AND CLOCK='1' COUNT_C1 BECOMES 520
NO
COUNT_C1<= COUNT_C1+1
YES
COUNT_C1 RESET RESULTS DIVIDE BY 520 CLOCK
B
DIVIDE BY 2 RESULS 9600 BAUD CLOCK
J
CLOCK'EVENT AND CLOCK='1' COUNT_C2 BECOMES 52
NO
COUNT_C2<= COUNT_C2+1
YES
COUNT_C1 RESET RESULTS DIVIDE BY 52 CLOCK
C
DIVIDE BY 2 RESULS BAUD *10 CLOCK(TEST CLOCK)
D
RxD= '1' & IS COUNTB BECOMES 80
YES
STOP<='1'
F
D
RxD= '1' & IS COUNTB BECOMES 90
YES
STOP<= '0'; COUNTB RESET; START<= '0';
NO
START= '1'
YES
COUNTB<=COUNTB+1
E
BAUD CLK
YES
RXDATA<=RXDATA &RXD
E
STOP<= '1'
YES
RXDATA1<=RXDATA
RXDATA1IS "8"
YES
RSEL<= '0'
G
RXDATA1IS "3"
YES
RSEL<= '1'
H

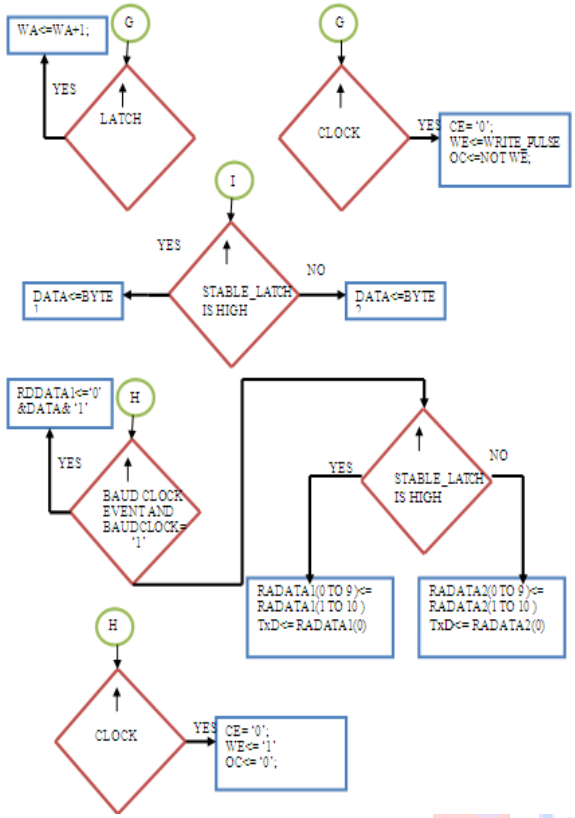


Fig: 5 : Digital Logic Flow chart

IV. CRITICALITY OF THE PROJECT

The requirement in this project is the total design should be in a small size of circular PCB dia 75mm and the interconnectivity should not occupy more space. This was achieved by soldering the components on two sides of the PCB and using the flexi rigid cables. These flexi-rigid cables will come along with the PCB and the interconnectivity of the PCB is achieved while making the PCB itself. With this all the three PCBs can sit one above the other with reasonable gap and occupy less space. Since the component density is very high, testing is very critical. The testing of internal logic of FPGA is carried out by configuring some of the pins as test points and routing the required monitoring signal to that test point since monitoring on the high density FPGA is not possible.

The most critical part in the software designing is interlocking. Some of the operations are synchronous and some are asynchronous. The sequence of events has to be followed as per the flowchart. To achieve this different latch signals were implemented in different stages. The chain processing has to be monitor after completing total activity. For this a non-volatile static RAM is provided which is loaded with the processed data at the fixed interval of time. The input byte which is coded in fixed format and the output byte which is also coded in a fixed format and written in to the static RAM continuously.

As and when, a request come to the FPGA on the UART, what ever data will be available in the NVRAM to be transmitted to the computer through RS-422 port .The UART logic was fully implemented and tested with in the FPGA. This method saved an external UART and interconnection logics and the space .

V. SYNTHESIS,SIMULATION ,FPGA PROGRAMMING AND HARDWARE REULTS

The digital logic for SAM is implemented by using VHDL language in XILINX-ISE tool. The VHDL code is complied and synthesized. The simulation was done by ISE-SIMULATOR. The simulation results are as shown in the Fig 6,7 and 8. The signals like CLOCK, ADDRESS, DATA, LATCH, CE, OE, WE, LATCH, STABLE_LATCH, Tx, Rx, EN ETC., are observed in the simulator as well as in the hardware. The UCF file was created for Translate, mapping and place and root and bit file has been generated. The generated bit file was programmed in FPGA by using XILINX-ISE JTAG Programmer.

The 28V DC power supply is used for the hardware. The hard ware test has been carried out by giving inputs and monitored the outputs at corresponding points by using oscilloscope and digital multimeter. The data can be logged in the NVRAM and when it is in read operation, the results of the data was monitored at the HyperTerminal by using RS-422 to USB converter. The results can be captured in text file and corresponding binary values of the ASCII code can be verified by using converter tool. The captured results of the data are as shown in the fig.9 ,10 and 11.

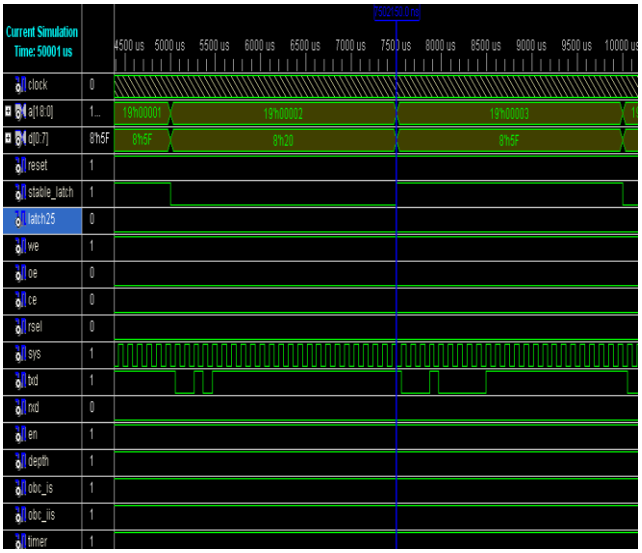


Fig 6 :Simulation results for baud clock and Tx

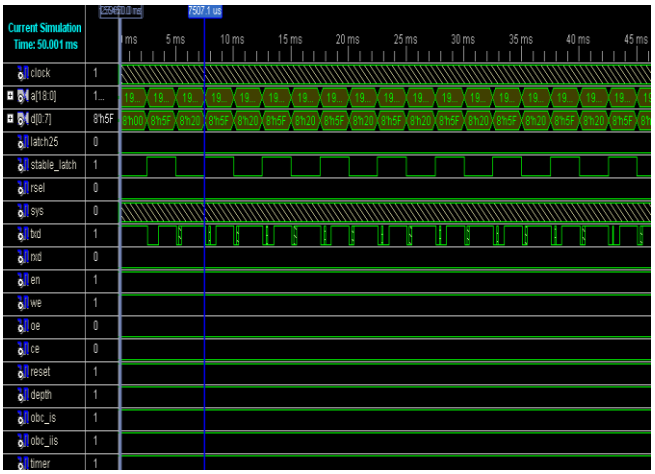


Fig 7 :Simulation results for stable latch and Tx

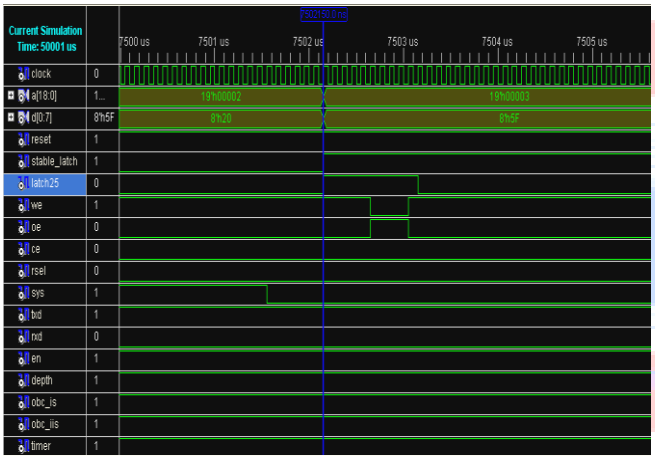


Fig 8 :Simulation results for latch ce, oe and we

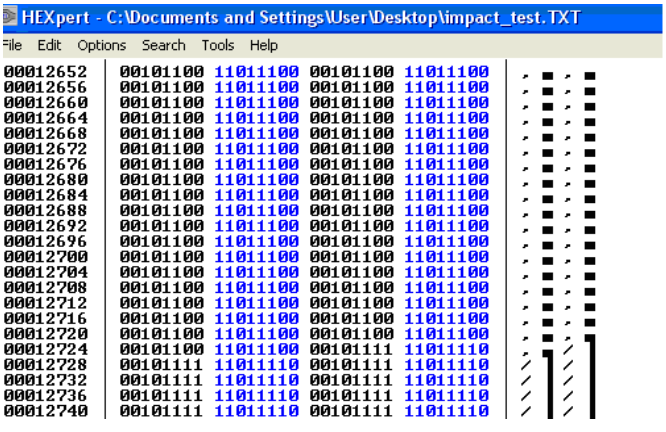


Fig9: NVRAM data at HyperTerminal when impact signal is given

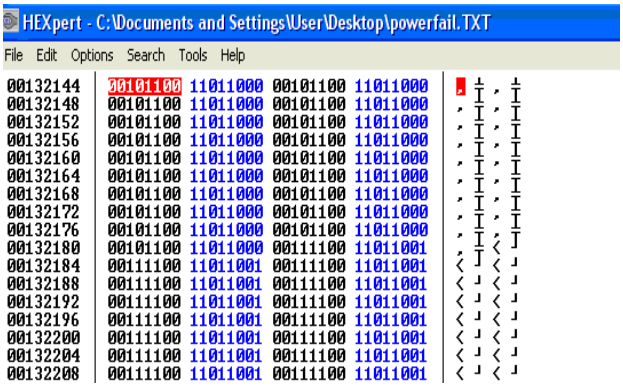


Fig10: NVRAM data at HyperTerminal when power fail occurs

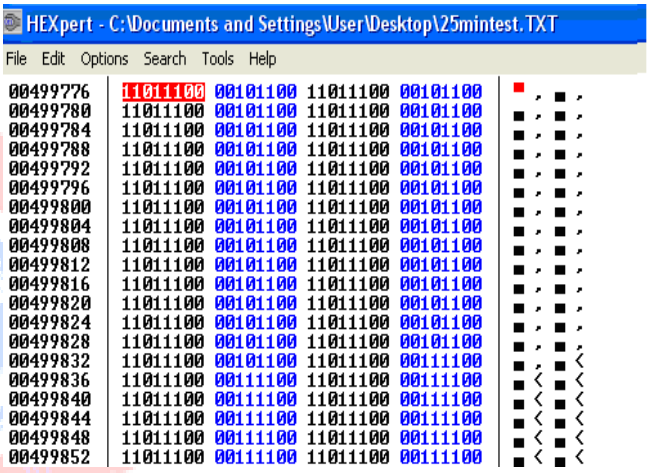


Fig11: NVRAM data at HyperTerminal after 25 minutes

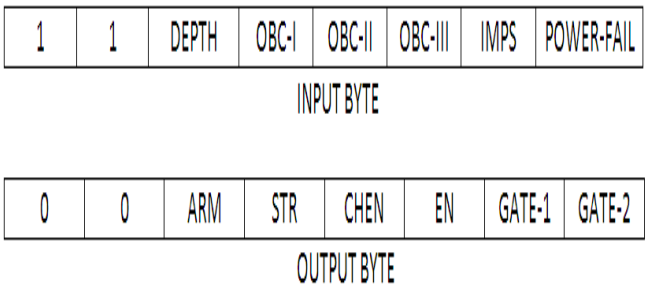


Fig12: Input and Output byte format

Note: In the fig.9,10 and 11, the byte which staring MSB bits with 11 that corresponds to input byte and the byte which staring MSB bits with 00 that corresponds to output byte and last column corresponds to ASCII character. The format for Input byte and output byte is as shown in fig.12.

VI. CONCLUSIONS

The required flowchart as per the specified inputs is implemented in the FPGA. The FPGA is operated at 10Mhz basic clock and the required signals are verified with appropriate timings. These results are transmitted to the computer system on a RS-22 port using an UART developed with the FPGA as per the limited requirement. The outputs on the computer are verified. The design is limited to one baud rate. This needs to be expanded further in case a full-fledged design of UART is to be carried out. The FPGA will have enough free macro cells available and can be reconfigured for any new application or additions for future design.

ACKNOWLEDGMENT

I sincerely thank Mr.K.Prabhakar, Director (Tech.), ANALOGIC CONTROLES INDIA LIMITED for giving me an opportunity and supporting me in carrying out this research work. I also thank the HOD of ECE Department, and faculty of ECE Department of JNTUA for the support provided.

REFERENCES

[1] "Irda and UART Design" –Dec-200- http://www.xilinx.com

[2] "Spatran-3 AN FPGA Family Data sheet" Dec-2007- http://www.xilinx.com

[3] "bq4015ly NON VOLATILE SRAM"-may1999-Revised-january2010 -http://www.ti.com

[4] "MAXIM-RS-485,RS-422 TRANSCIVERS" –Revised-Dec-1994- http://www.datasheetcatalog.com

[5] "Low power microprocessor supervisory circuits-SP705"-2009 copyright 2000- http://www.datasheetcatalog.com

[6] "Gaia converter HI-REL DC-DC CONVERTER MGDM-04:4W POWER"-2006 Revised Aug-2006-- http://www.gaia-converter.com

[7] "National Semiconductor –low power quad operational amplifiers"-August-2000. - http://www.national.com

[8] "Micro-processors and interfacing"-Revised-2006-Douglas v. Hall

[9] "Digital Design"-Fourth edition -2008-John- F warlkerly

[10] "A VHDL PRIMER"-Third edition-2010-J.Bhasker.