

Design And Implementation of an Efficient And Modernized Technique of a Car Automation Using Spartan - 3 FPGA

* S. Raghupathi *S.Baskar

* Professor – Shivani institute of technology, Trichy, Tamilnadu, India.

* PG-Scholar-SNSCT, Coimbatore, Tamilnadu, India.

Abstract---This paper deals with the integration of various functions involved in the automation of car in to a single chip so that the cost and complexity of the system can be reduced. This paper also includes the concept of Low Power Optimization technique. In an existing CAN control system, the number of nodes required to control various parameters of an automation process are more and it may increase the number of controllers. Also separate modules are to be programmed for different nodes and the large numbers of interfaces are required for the CAN system. These requirements of the CAN control system makes the design and implementation of an Automation, a complex one.

The objective of this paper enumerates a new automation system in which FPGA is employed to integrate all the functions into a single chip. The FPGA chip acts as master controller, receiving the inputs from the various sensors and sending the proper command signals to particular system in time for controlling the specific action of the car. The paper mainly focuses on the following parameters such as Engine Speed, Engine temperature, Mirror adjustment, Wiper control, Anti lock brake system, RSB-Rotational Sensing Block, and Obstacle detection. Significance stands with the Anti lock brake system and the Obstacle detection.

In this paper the vehicle automation is done using FPGA to reduce the complexity and cost than the existing CAN mechanism. Here seven main control parameters are taken into account and these parameters are controlled successfully with the concept of low power optimization. In future more parameters (nearly 40) can be controlled to improve the performance of the car automation system. Also the system can be optimized for speed, area and power using optimization techniques.

Keywords---Automation, Sensor, Low power, Abs, Fpga Board.

features, FPGAs can be programmed and reprogrammed as needed

I. BACKGROUND OF AUTOMATION

Constantly growing requirements for environmental protection, as well as rising demands from the end-users to improve fuel economy, safety and driving comfort, have contributed significantly to the rapid development of automotive electronics.

In existing system such as CAN control system, in the automation of car, number of nodes required for the control of various parameters in the automation process are more and it may increase the number of controllers. Also separate modules are to be programmed for different nodes and the large numbers of interfaces are required in the CAN system. These requirements of the CAN control makes the design and implementation of Automation of vehicle, a complex one.

To overcome the aforesaid problem of CAN controller usage, a flexible microcontroller solution in silicon with field programmable gate arrays (FPGAs) is available for implementation of all the desired functions. FPGA devices offer a powerful, viable alternative to CAN microcontrollers because these significantly reduce engineering development time and the cost of multiple silicon iteration. Unlike fixed-silicon microcontrollers that may be missing required

During the design process, enabling more rapid prototyping and faster time-to-market. This can also be upgraded in the field if requirements change-even after the devices are developed in a product.

A new FPGA has been designed, developed and synthesized in Xilinx effectively to overcome the defects of the CAN controller. Hence in future the Field programmable gate arrays will prove to be a potential flexible master controller. This master controller is a promising system for the use of the new modern automation industry. Thus offering the latest electronic system that makes automobiles better performers and more entertaining for the occupants.

II. CAN

CAN is a message based protocol, designed specifically for automotive applications but now also used in other areas such as industrial automation and medical equipment. CAN is a multi-master broadcast serial bus standard for connecting electronic control units (ECUs). Each node is able to send and

receive messages, but not simultaneously. A message consists primarily of an ID which represents the priority of the message and up to eight data bytes. It is transmitted serially onto the bus. This signal pattern is encoded in NRZ and is sensed by all nodes. The devices that are connected by a CAN network are typically sensors, actuators, and other control devices. These devices are not connected directly to the bus, but through a host processor and a CAN controller.

III. VLSI

VLSI is the abbreviation of very large-scale integrated circuit. It is the process of creating integrated circuits by combining thousands of transistor-based circuit into a single chip. The VLSI technology has been successfully used to build microprocessors, signal processors, systolic arrays, large capacitive memories, memory controllers and interconnection networks.

IV. VHDL

VHDL is the abbreviation of VHSIC (Very High Speed Integrated Circuits) Hardware Description Language. It is commonly used as a design-entry language for field-programmable gate arrays and application-specific integrated circuits in electronic design automation of digital circuits. There are typically three phases of a design process, which use VHDL. The specification phase, the design phase and the gate level synthesis phase. System modeling is done during the specification phase while register-transfer modeling is done during the design phase. The net list from synthesis of the RTL (Register Transfer Level) model is obtained in the gate level synthesis phase. At each level, either a behavioral level or a structural model, or both can be developed according to the design needs.

V. BLOCK DIAGRAM OF CAR AUTOMATION

Block Diagram Description

I. Master N/W Controller

All the accessories are connected to individual modules with inbuilt FPGA module. The accessories are controlled by a master controller, which consists of the user interface and the display placed on the front panel of the automobile. The user can keep track of the status of all the individual nodes and the user can have liberty to either manually control the nodes or make it to work automatically. FPGA is a computer in the car. FPGA is programmed in such a way to detect the sensor outputs as inputs and it enables corresponding output ports. The logic can be programmed using behavioral description such as if then etc, or by developing look up tables, or by defining set of conditions. The Methodology involves in this paper is , the whole automation of the vehicle is coded in VHDL, simulated using Modelsim and

synthesized using Xilinx ISE environment and downloaded in Spartan-3 FPGA kit. The Xilinx FPGA is similar to CPLD, with smaller functional blocks but in large number spread out evenly across the entire chip along with programmable interconnects.

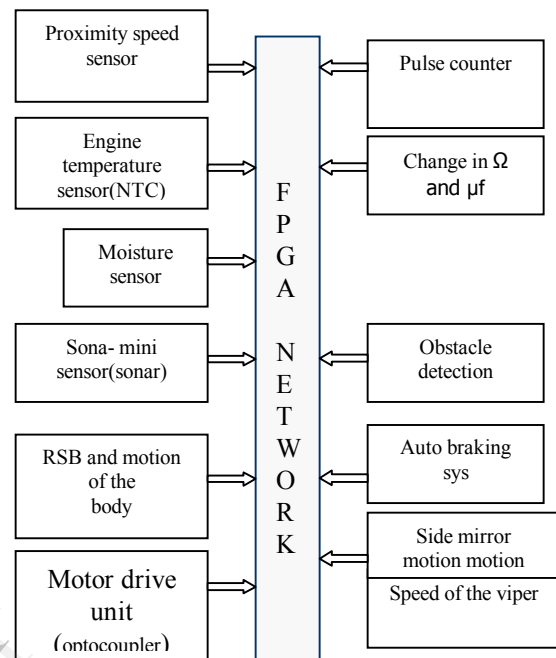


Fig.1. Block Diagram Description of car automation

ii. Speed sensor

The speed of a car can be monitored easily with the help of speed sensor. There are many types of speed sensors are available. Proximity sensor is the best one to determines the frequency of pulses as created from the engine speed sensor. These pulses are counted by the FPGA and it is monitored.

iii. Temperature and humidity sensor

Based on a unique capacitive cell for humidity measurement and a Negative Temperature Coefficient (NTC) thermistor for temperature measurement, this dual purpose relative humidity / temperature miniaturized sensor is designed for high volume, cost sensitive applications with tight space constraints. The change in resistance is measured as an engine temperature and the change in capacitance is measured as relative humidity. Thus the FPGA monitors both engine temperature and the relative humidity.

iv. Ultrasonic sensor

The SonaMini Sonar Sensors are complete sonar transmitter/receiver systems. They are designed to be used as rangefinders, but can be hacked for a variety of other purposes (communications, etc).

Determining range with the SonaMini Sensor is easy but requires careful measurement of the pulse width on the Sonar TOF Out pin. Every time the sonar "pings", a pulse is generated on the Sonar TOF Out pin. It measures the pulse width in seconds, and get the amount of time the sound pulse took to travel to the target and then travel back to the SonaMini. This time is called the sonar Time-of-Flight, or **TOF**. Multiply the TOF by the speed of sound, the result is the round-trip distance, or twice the range to the target. Thus the obstacle detection can be monitored by the FPGA.

v. *Anti lock braking system*

Stopping a car in a hurry on a slippery road can be very challenging. Anti-lock braking systems (ABS) take a lot of the challenge out of this sometimes nerve-wracking event. The FPGA will receive the input signal from the RSB and with respect to Speed and the angle of the rotation, and another input from the motion of the body it will provide alert to apply ABS.

vi. *Motor Drive unit*

The motor drive unit is used to provide control voltage to vary the speed of the dc motor used for wiper or other drives. The motor drive unit consists of an opto-coupler, which couples necessary voltage proportional to the speed of the dc motor through the control voltage available at the FPGA output pin.

vii. *Side mirror motion*

Giving the input to the motor drive unit, which drives a stepper motor, can change the side mirror motion. The motor drive unit receives the input from FPGA and makes the stepper motor to change the direction of the mirror. Thus the mirror motion can be controlled by the FPGA.

Parameters to be automated

- Engine speed
- Engine temperature
- Mirror adjustment
- Wiper control
- Antilock break system
- Rotational Sensing Block
- Obstacle detection
- Motion of the body

VI. SENSORS

A. *SONAMINI SONAR SENSOR*

The Sona Mini Sonar Sensors are complete sonar transmitter/receiver systems. They are designed to be used as rangefinders, but can be hacked for a variety of other purposes (communications, etc).

- Pin1:- 8-16VDC Power Input
- Pin2:- Ground

- Pin3:- External Trigger Input
- Pin4:- External Trigger Enable
- Pin5:- Sonar TOF Out
- Pin6:- Range Trigger 1
- Pin7:- Range Trigger 2

I. Determining Range using the SonaMini

Every time the sonar "pings", a pulse is generated on the Sonar TOF Out pin. It measures the pulse width in seconds, and get the amount of time the sound pulse took to travel to the target and then travel back to the SonaMini. This time is called the sonar Time-of-Flight, or **TOF**.

$$\bullet \text{ Actual TOF[s]} = \text{TTOF Pulse width[s]} - \text{TDet Delay[s]}$$

$$\bullet \text{ Range[m]} = (\text{Actual TOF[s]} * 340.29[\text{m/s}]) / 2$$

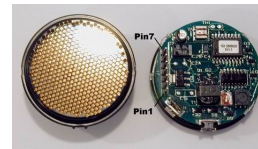


Fig-1: SonaMini Sonar Sensor

B. SPEED SENSOR

The automobile's engine contains a speed sensor. This speed sensor automatically sends the information to the computer as to how fast the car is traveling at the moment of driving. The engine's speed sensor is craftily designed to be able to record the rate at which the vehicle's crankshaft is spinning.



Fig-2: Toyota Matrix – Speed Sensor BECK ARNLEY
VEHICLE SPEED

C. TEMPERATURE AND RELATIVE HUMIDITY SENSOR

i. Humidity Sensor Specific Features

- Miniature Surface mount SMD package
- Compatible with automated assembly processes, including wave soldering, water immersion and reflow soldering.
- Full interchangeability with no calibration required in standard conditions
- High reliability and long term stability
- Available in tape and reel for automatic pick and place

- Individual marking for compliance to stringent traceability requirements
- Instantaneous desaturation after long periods in saturation phase
- Patented solid polymer structure
- Suitable for linear voltage or frequency output circuitry
- Fast response time

ii. Temperature sensor specific features maximum ratings characteristics

- High quality thermistor

Storage Temperature Tstg - 40 to 100 °C

Humidity Operating Range RH 0 to 100 % RH

Temperature Operating Range Ta - 40 to 100 °C

Maximum Electric Power to be supplied (continuous) @ 25°C
P25 2 mW

Humidity measuring range RH 1 99 %

Nominal capacitance @ 55 % RH* C 177 180 183 pF

Averaged Sensitivity from 33 % to 75 % RH? C/%RH 0.34
pF/%RH

Long term stability 0.5 %RH/yr

Recovery time after 150 hours of condensation 10 s

Response time (33 to 76 % RH, static, @ 63 %) τ 10 s



Fig -3: Temperature and Humidity Sensor

VII. FPGA INTRODUCTION

Field programmable gate arrays are programmable by the users.

With FGPA's debugging or prototyping of new design can be done as easily and quickly as software. But FGPA's performs much faster than software on computer. As the price of FGPA's goes down with higher speed FGPA's are replacing other semi custom design approaches in many applications.

A step above the PLD in complexity is the field-programmable gate array (FPGA). There is very little difference between an FPGA and a PLD-an FPGA is usually just larger and more complex than a PLD.

Specification

FPGA trainer contains the following parts

1. BASEBOARD
2. DAUGHTER BOARD
3. PROGRAMMING TOOL
4. POWER SUPPLY

Hardware description baseboard

Baseboard gives a detailed description of the hardware available in the Baseboard. It facilitates the user to allot

various I/Os of the FPGA to the hardware present on this board.

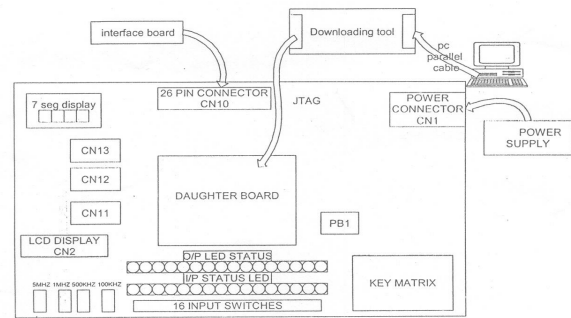


Fig-4: Connection Diagram of Base Board

DAUGHTER BOARD-1 (FPGA – XILINX)

The Spartan-3 family of Field-Programmable Gate Arrays is specifically designed to meet the needs of high volume, Cost-sensitive consumer electronic applications. The eight-member family offers densities ranging from 50,000 to five million-system gates. Because of consumer electronics applications, including broadband access, home networking, display/projection and digital television equipment

Connection diagram of FPGA

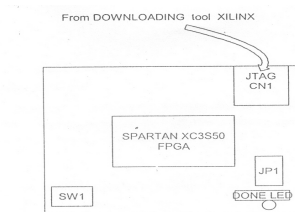


Fig-5: Connection diagram of FPGA

VIII. TOOL INTRODUCTION

Integrated Software Environment (ISE) is the Xilinx design software suite. ISE can be used by a full spectrum of designers, from the first time CPLD designer to the experienced ASIC designer transitioning to FPGA.

ISE enables to start the design with any of a number of different source types, including:

- HDL (VHDL, Verilog HDL, ABEL)
- Schematic design files
- EDIF
- NGC/NGO
- State Machines
- IP Cores

Design Entry

- ISE Text Editor - The ISE Text Editor is provided in ISE for entering design code and viewing reports.
- Schematic Editor - The Engineering Capture System (ECS) is a graphical user interface (GUI) that allows you to create, view, and edit schematics and

symbols for the Design Entry step of the Xilinx® design flow.

- CORE Generator - The CORE Generator System is a design tool that delivers parameterized cores optimized for Xilinx FPGAs ranging in complexity from simple arithmetic operators such as adders, to system-level building blocks such as filters, transforms, FIFOs, and memories.

- Constraints Editor - The Constraints Editor allows you to create and modify the most commonly used timing constraints.

- PACE - The Pinout and Area Constraints Editor (PACE) allows you to view and edit I/O, Global logic, and Area Group constraints.

- StateCAD State Machine Editor - StateCAD allows you to specify states, transitions, and actions in a graphical editor. The state machine will be created in HDL.

Synthesis

- XST - Xilinx Synthesis Technology.
- Integration with LeonardoSpectrum from Mentor Graphics, Inc.
- Integration with Synplify from Synplify, Inc.

Simulation

- HDL Bencher test bench waveform - The HDL Bencher tool automates creation of test benches and test fixtures using a graphical waveform editor.
- Integration with ModelSim Simulator from Model Technology, Inc.

Implementation

- Translate - The Translate process runs NGDBuild to merge all of the input netlists as well as design constraint information into a Xilinx database file.
- Map - The Map program maps a logical design to a Xilinx FPGA.
- Place and Route (PAR) - The PAR program accepts the mapped design, places and routes the FPGA, and produces output for the bitstream generator.
- Floorplanner - The Floorplanner allows you to view a graphical representation of the FPGA, and to view and modify the placed design.
- FPGA Editor - The FPGA Editor allows you view and modify the physical implementation, including routing.
- Timing Analyzer - The Timing Analyzer provides a way to perform static timing analysis on FPGA and CPLD designs. With Timing Analyzer, analysis can be performed immediately after mapping, placing or routing an FPGA design, and after fitting and routing a CPLD design.

Fit (CPLD only) - The CPLDFit process maps

Device Download and Program File Formatting

netlist(s) into specified devices and creates the JEDEC programming file.

- ChipViewer (CPLD only) - The ChipViewer tool provides a graphical view of the inputs and outputs, macrocell details, equations, and pin assignments.

- BitGen - The BitGen program receives the placed and routed design and produces a bitstream for Xilinx device configuration.

- iMPACT - The iMPACT tool generates various programming file formats, and subsequently allows you to configure your device.

- XPower - XPower enables you to interactively and automatically analyze power consumption for Xilinx FPGA and CPLD devices.

- Integration with ChipScope Pro.

IX. CONCLUSION

In this paper several micro controllers for each functions has been devised into a single master controller for performing various functions. This master controller is a promising system for the use of the new modern automation industry, thus offering the latest electronic system that makes automobiles better performers and more entertaining for the end users.

The vehicle automation is done using FPGA to reduce the complexity and cost of the existing CAN mechanism. In future more parameters (more than 40) can be added to improve the performance of the car automation system. Also the system can be optimized for speed, area and power using optimization techniques.

ACKNOWLEDGEMENT

The main author wishes to acknowledge the chairman of shivani institute of technology **Dr.P.Selvaraj** for his consistent encouragement towards the research activity and motivation given for the preparation of this article.

REFERENCES

- [1] Journal of the Institution of Engineers, Singapore Vol. 45.
- [2] International journal of mathematical models and methods in applied sciences Issue 1, Volume 2, 2008
- [3] Prof.Dr.-Ing. Klaus D. Müller-Glaser „ Challenges for reliable software design in automotive electronic control units ”
- [4] Shabiul Islam, NowshadAmin, Mukter, M.S.Bhuyan Md “ Fuzzy based PID Controller using VHDL for Transportation Application “.
- [5] Michael John Sebastian Smith Application Specific Integrated Circuits