IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

A Novel System for Monitoring and Controlling Industrial Engine Operation, using Vibration, Magnetic Field, Humidity and Temperature Sensors, Implemented with FPGAs and VHDL

Leonidas Dimitriadis[1], Dr Evangelos I. Dimitriadis[2]

Undergraduate student, Department of Information and Electronic Engineering, International Hellenic University, 57400, Sindos Thessaloniki, Greece[1]

Department of Computer, Informatics and Telecommunications Engineering, International Hellenic University, End of Magnisias Str, 62124 Serres Greece.[2]

Abstract: A novel system which uses FPGAs and VHDL for monitoring and controlling industrial engine operation, is presented here. The system monitors and controls four basic parameters exerting decisive effect on engine operation, using corresponding sensors. Engine's vibration, spontaneous fluctuations of magnetic field, humidity and temperature, are simultaneously monitored and controlled. Each sensor acts as input for DE10-Lite FPGA board used here and its values are presented in seven-segment displays. Every sensor is accompanied by indicator LEDs, of which green one indicates that input values are below critical set value limit, while red indicates that the input values of the corresponding parameter are greater than or equal to the critical upper value set by the programmer. Our system also uses three additional LEDs controlling the overall engine operation, from which the green one indicates that all sensor input values are within acceptable limits, while red one shows that at least one parameter's input values are above set limits, meaning that there is a danger for the engine because it is not operating in the appropriate way. The third and yellow LED is activated when the engine overcomes time set period of operation, thus our system implements an additional control and protection of the industrial engine. Both FPGA's board LEDs and buzzer are activated only if vibration input values are above upper set limit. Our system is cheap to manufacture, easy to use and can be also combined with IoT and AI systems, allowing it to share and process valuable information of all factory's engines. It also provides the flexibility of setting upper sensor input value limits, depending on monitored engines, thus making it useful in a variety of applications.

Keywords: Industrial engine monitoring, sensors, vibration, magnetic field, humidity, temperature, FPGA, VHDL, Buzzer, LEDs.

I. INTRODUCTION

It is well known that FPGAs have attracted attention of a great number of researchers in the recent years, for industrial as well as for a variety of other applications. [1-13] FPGAs have the main advantage of combining software and hardware, thus enabling hardware programming for a series of applications. The most used languages for FPGAs' programming are VHDL and Verilog and VHDL is the one used in our work.

Although a lot of work has been done concerning implementation of FPGAs in a variety of applications as mentioned above, there are few works [14-16] dealing with industrial engines monitoring and controlling. The above works present FPGAs in industrial control applications, FPGA-Based Motor Control Systems for Industrial Automation and the third work shows how to achieve the goal of Integrating FPGA-Based Acceleration in Industrial Motion Control System.

All of the above works do not solve the problem of presenting an FPGA-based, simple, portable, easy to use and low cost industrial engine monitoring and controlling system, capable of maintaining all factory engines in good operating conditions, thus avoiding unwanted repair costs and delays in production line, since it is so important to act before anything happens instead of reacting when the damage is already done.

© IJIREEICE This work is licensed under a Creative Commons Attribution 4.0 International License

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

We present here a system which monitors and controls simultaneously, four basic engine operation parameters by using corresponding sensors. Vibration, magnetic field, humidity and temperature are the most important parameters affecting industrial engine's operation and our system monitors and controls all of the above parameters. It can also provide information about engine's operation to a central AI based system or send useful data via 5G system to a remote controlling system.

Another benefit of our system is that it can work with a variety of sensors with different sensitivities, as long as they have an analog output. It can also operate to different value ranges set by the programmer, thus providing solution for various types of industrial engines.

II. DESIGN OVERVIEW AND OPERATION OF THE SYSTEM

Figure1 presents device overview and operational units of our system, using FPGA DE10-Lite board, while Figure 2 presents circuit diagram of the system. Subsequent Figure 3 presents the implemented industrial engine monitoring system of this work.

It is obvious from the above figures that our system, except from DE10-Lite FPGA board, contains also some basic circuit parts. It uses four sensor units with their corresponding green and red LEDs presenting appropriate or harmful operation, respectively. We can see SW1801P vibration sensor unit, KY035 analog Hall magnetic sensor unit, HR202 humidity sensor unit and LM35 temperature sensor unit. We also use the buzzer unit which is activated if at least one sensor values are higher than upper set limit and total alarm system unit, which gives information about overall system operation.

Time is the other input value used here and it is provided by FPGA's clock.

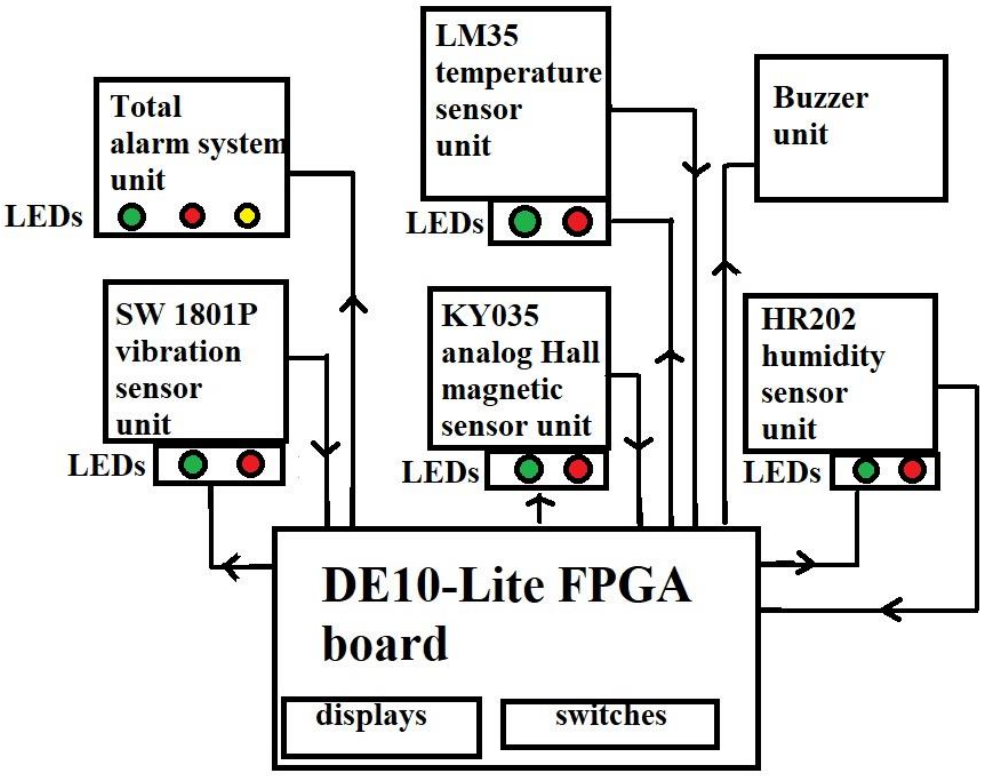

Figure 1: Device overview and operational units of our system.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

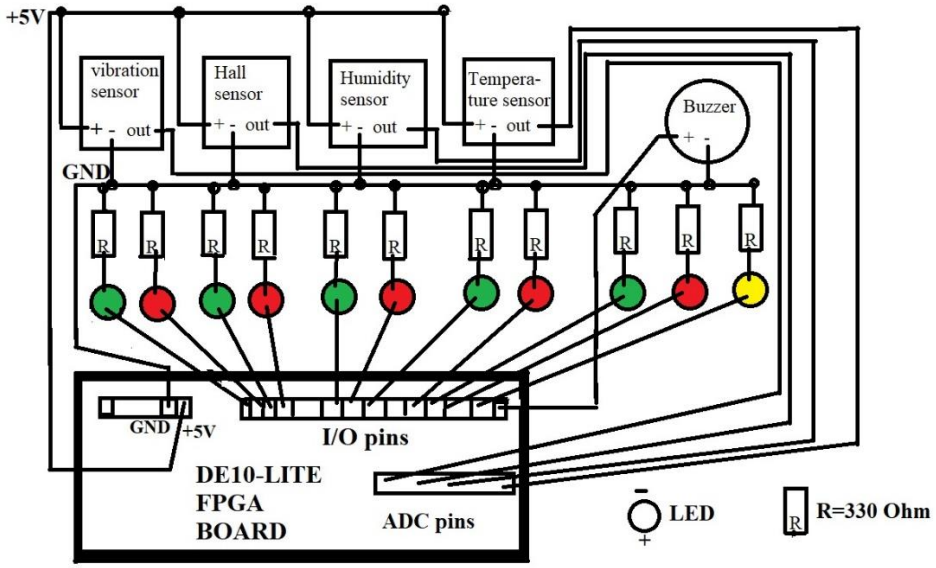

Figure 2: Circuit diagram of our system

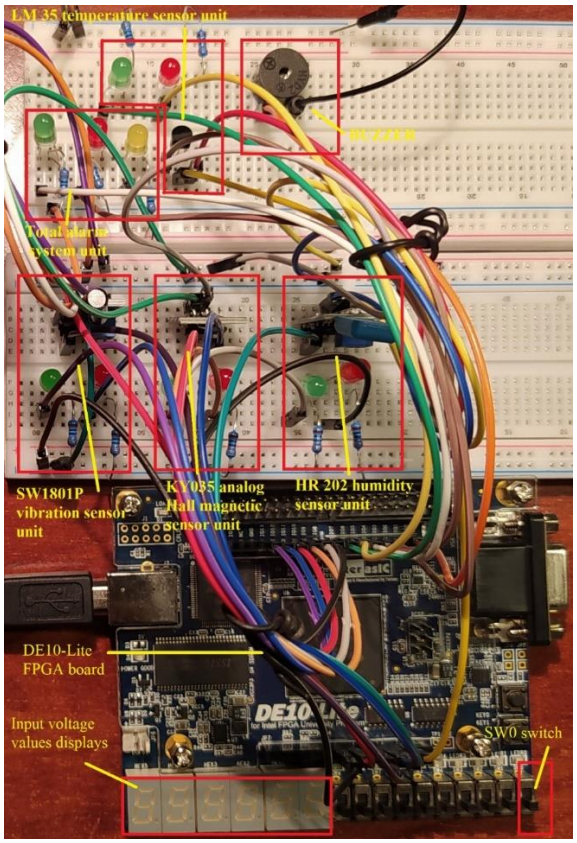

Figure 3: The implemented industrial engine monitoring system of this work.

Our industrial engine monitoring system starts operating as soon as power supply +5V is applied to the circuits via DE10-Lite appropriate pins and the VHDL program is sent via USB Blaster interface, to FPGA chip. Input values from all four sensor units and FPGA's clock are entered in our system. Analog to digital converters (ADC) of DE10-Lite proceed to conversion and finally input voltage values are presented in seven-segment displays of the FPGA board. Since we want to present four input values using two decimal digits for each sensor value, the use of SW0 switch is necessary. In case that SW0 is OFF seven-segment displays present from left to right digits, vibration sensor and Hall analog magnetic sensor input values. If SW0 goes to ON state then humidity and temperature sensors input values are presented. It must be mentioned here that for all sensors input voltage values, we set a specific upper value limit and if

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

input values are lower than set limit, industrial engine operates appropriately and no danger for damaging it exists. Consequently corresponding green LEDs light ON for each sensor. In the opposite case corresponding red LEDs light ON and there is a risk of damaging the engine if its operation continues.

We used calibration factors within VHDL program to set sensors input values to a reasonable range. Then we set upper limits for every sensor, which of course can vary depending on the application. Vibration input values (Vvibr) limit was set to 2.0 Volts, Hall analog magnetic input values (Vhall) limit was set to 3.0 Volts, humidity input values (Vhum) limit was set to 3.0 Volts and temperature input values (Vtemp) limit was also also set to 3.0 Volts.

Concerning the most important factor of machine vibration limits in industry, they are determined by standards like ISO 10816, which categorize machines and use velocity limits to define acceptable, restricted, and damaging vibration levels. For many industrial machines, an acceptable range for unrestricted operation is less than 2.8 mm/sec. Values between 2.8–4.5 mm/sec are considered restricted, while anything above 4.5 mm/sec indicates a potential for damage and requires immediate attention. We observed that for SW1801P vibration sensor the upper value set limit of 2.0 Volts was in accordance with the above restrictions.

Similar calibration procedures were used for the other three sensors, leading to the upper input values set limits, mentioned above.

Figure 4 presents the industrial engine monitoring system in operation mode. SW0 switch is OFF, so input values of vibration sensor and Hall sensor are displayed in seven-segment displays. All four sensor input values, according to the upper set limit values mentioned above, are below critical set values, thus four sensor green LEDs and overall alarm system green LED, are ON. Yellow alarm LED is also ON, indicating that normal set operation period of the engine, has expired. We must mention that the above operation time period was set here to a value of only 1 min, in order to check its functioning. The system provides the ability of setting this time period as long as it is needed, depending on the application that our system is used.

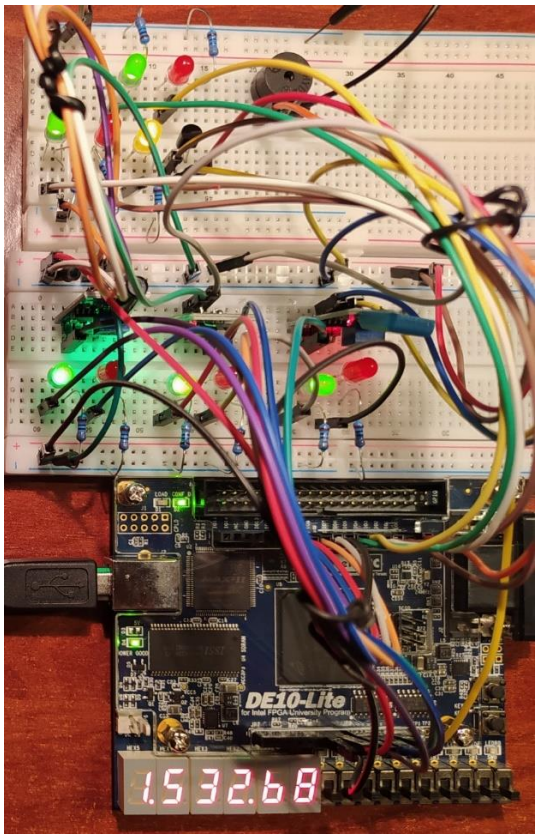

Figure 4: The industrial engine monitoring system in operation mode. SW0 switch is OFF, so input values of vibration sensor and Hall sensor are displayed. All four sensor input values are below critical set values, thus four sensor green LEDs and alarm system green LED, are ON. Yellow alarm LED is also ON, indicating that normal set operation period of the engine, has expired.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

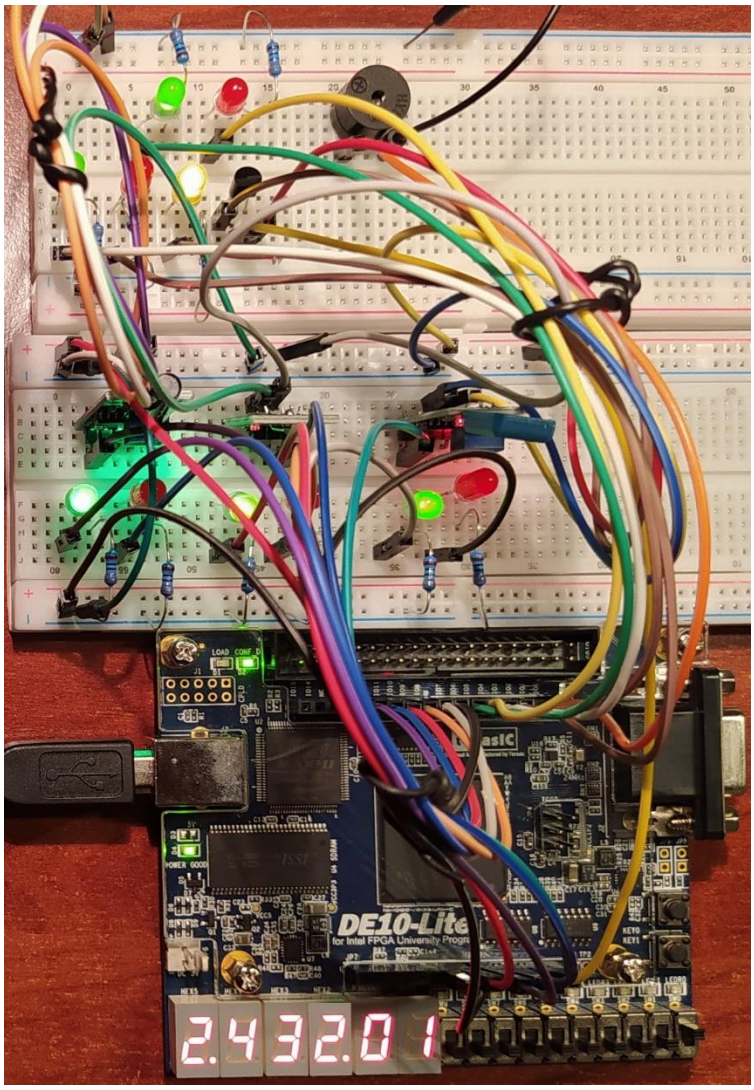

Figure 5: The industrial engine monitoring system in operation mode. SW0 switch is ON, so input values of humidity sensor and temperature sensor are displayed. All four sensor input values are below critical set values, thus four sensor green LEDs and alarm system green LED, are ON. Yellow alarm LED is also ON, indicating that normal set operation period of the engine, has expired.

Figure 5 presents the industrial engine monitoring system in operation mode and SW0 switch is now to ON state, so input values of humidity sensor and temperature sensor are displayed in seven-segment displays. All four sensor input values are below critical set values, thus four sensor green LEDs and alarm system green LED, are ON. Yellow alarm LED is also ON, similarly to Figure 4, indicating that normal set operation period of the engine, has expired. The use of SW0 switch is significant for the person who checks optically the engine's parameters input values.

It must be mentioned here that simultaneously with the signals send to all LEDs of our system, the above signals could be send to an AI system which could be able to process all engines signals in real time and provide useful information and also prevent significant engines operation, if there is danger for damaging them. All of the above mentioned external LEDs are connected to I/O pins of the FPGA board and act as outputs for our system.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

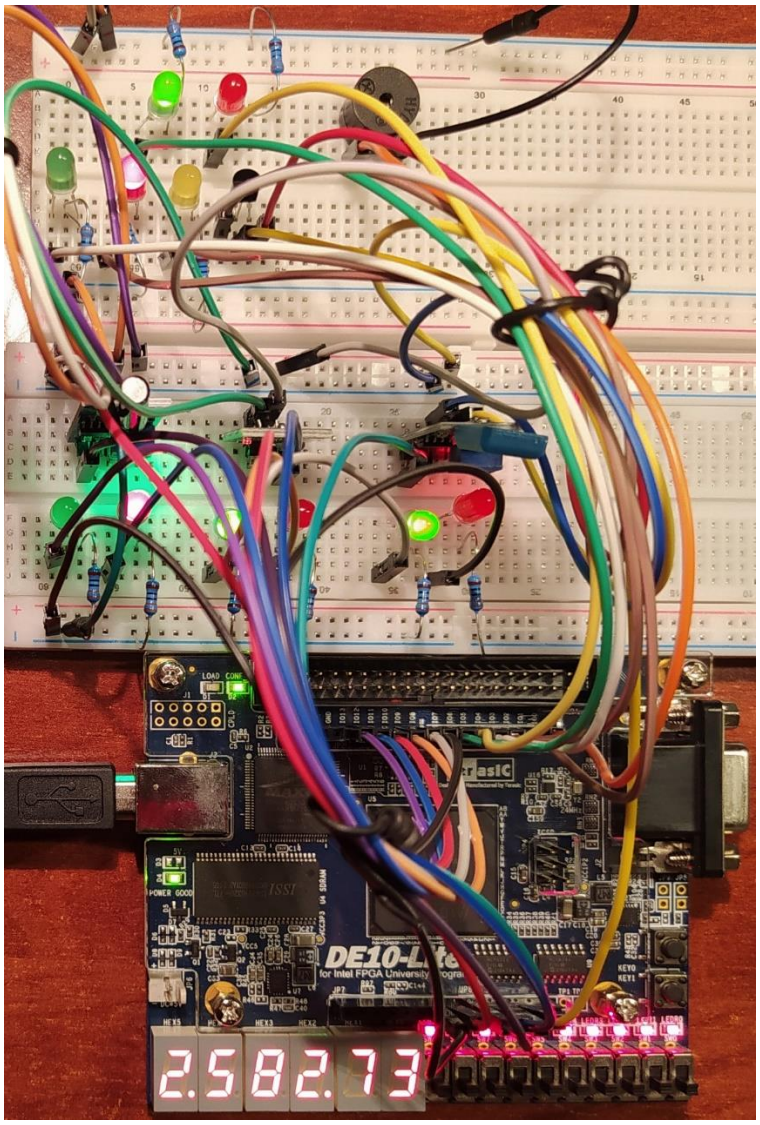

Figure 6: Our industrial engine monitoring system operating. Vibration input sensor value at left display exceeds upper set value of 2.00V, thus sensor red LED and system alarm red LED are ON. In addition board LEDs are ON and buzzer sounds.

Figure 6 shows our industrial engine monitoring system operating. SW0 switch is OFF so the first value displayed from left to right displays, corresponds to vibration sensor input values. It can be seen that vibration input sensor value at left display exceeds upper set value of 2.00V, thus sensor red LED and system alarm red LED are both ON. In addition FPGA board LEDs are ON, because vibration of an engine is the most significant parameter for its appropriate operation. Buzzer also sounds because at least one sensor input values are above set limits.

Concerning the other three parameters monitored here, Hall magnetic analog sensor values are the second parameter monitored here. These values are also important because knowing engine's magnetic field and consequently magnetic flux, lets you assess its energy consumption. By measuring and analyzing magnetic field and magnetic flux allows for the assessment of energy consumption and efficiency in electromechanical systems, such as motors, transformers, and magnetic filters. In conjunction, spontaneous sharp fluctuations of magnetic parameters are directly affected by current flow within the electric parts of the engine, thus providing useful information about engine's appropriate operation.

The third parameter monitored and controlled with our system is humidity of the engine's environment. It is very important to achieve appropriate humidity values in order to avoid machine wear and prevent possible short circuits in the electrical and electronic parts of the engine.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

Finally the fourth parameter monitored with our system is machine temperature. Needless to mention that engine's overheating may cause irreparable damage, thus machine temperature is required to be monitored and controlled.

Another important fact of our system's design is that it receives input sensor voltage values periodically, ensuring continuous parameters monitoring.

Our industrial engine monitoring system also offers the advantages of being easy to use and also cheap to manufacture.

III. PROGRAMING THE SYSTEM

We used Quartus Prime Lite Edition 21.1.1 to create the VHDL programs of our system. It must be mentioned here that before proceeding with the VHDL programming of our system, we had to set a series of parameters controlling the operation of DE10-Lite FPGA's Analog to Digital Converter (ADC). This converter plays a very important role in the whole system operation, since it converts the analogue input voltages from all sensors connected to FPGA board to digital values, acting as main input of the system. The files created by the above ADC parameters setting are imported into the final project of our system.

A flowchart diagram, presenting main functions of our system is presented in Figure 7, while the APPENDIX contains the whole VHDL program.

It is clear that the system operates seven basic functions. All of them use processes in VHDL programming language. These processes are running as long as the system is ON. All external or board's LEDs as well as buzzer system, are programmed to work as logic outputs, thus they are at the ON state if they receive binary '1' as logic output.

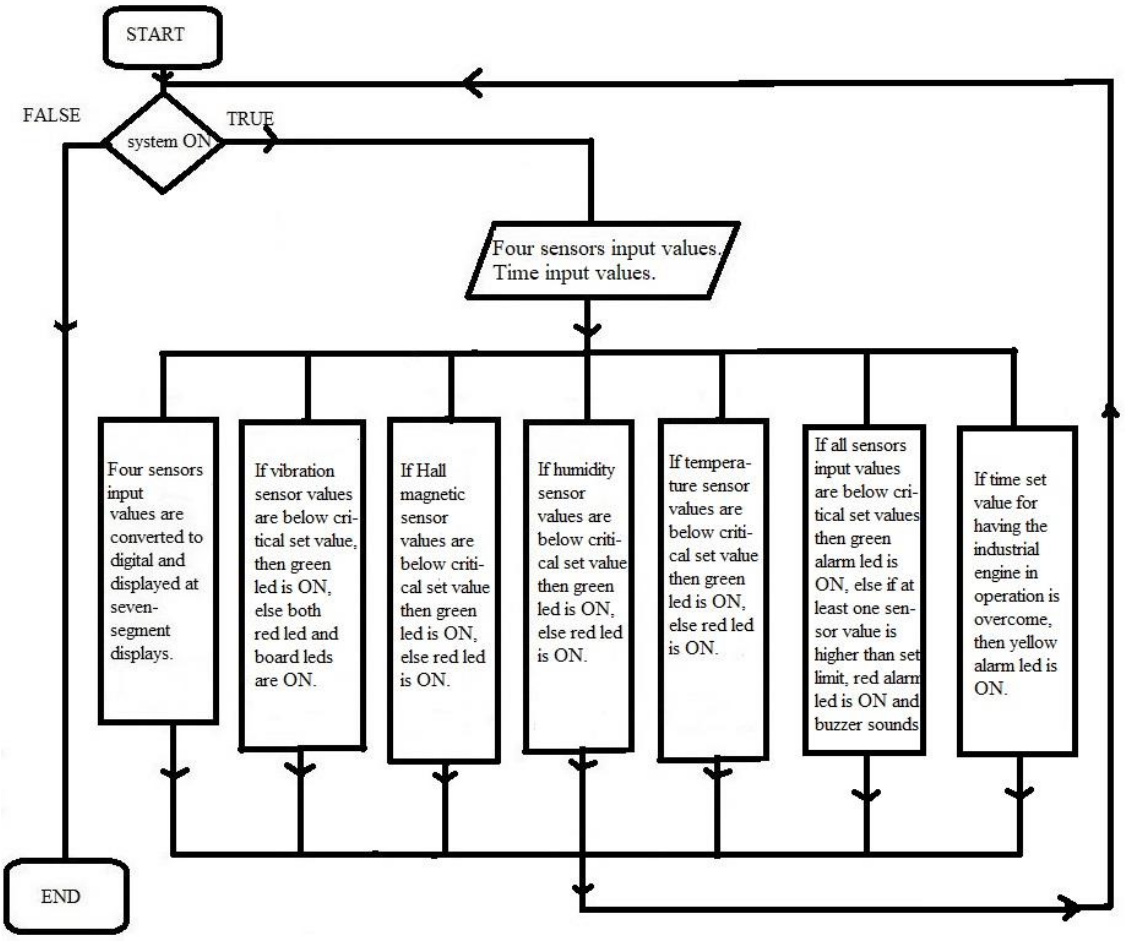

Figure 7: Flowchart diagram, presenting main functions-processes of our system.

DOI: 10.17148/IJIREEICE.2026.14501

First function of Figure 7 playing definitive role in system operation, uses analog input voltage values provided from all sensors as main input, convert them to digital values and present the final result in seven-segment displays of the FPGA board. Calculated input sensor voltage values are used in other processes, in order to activate external or internal LEDs and buzzer. A separate process based on FPGA board's clock, is also activated upon system's transition to ON state. It is a very important process whenever time parameter is needed, but it is not shown as a separate function in Figure 7.

Next four functions are related to four sensors used in the system. Every function uses a VHDL process in which the possible reaching or overcoming the upper input value set limit, is being checked. Engine's vibration, Hall magnetic field, humidity and temperature values are continuously monitored and all four processes are programmed to send logic '1' to corresponding external green LEDs if the upper set input value limit is not reached or overcome. In the opposite case logic '1' is send to red external LED notifying that engine is in danger of damage if its operation continues. Especially for vibration sensor input values reaching or overcoming, FPGA's board LEDs are also ON.

Sixth function is related to total alarm unit of our system. In case that at least one of four sensor's input value is equal or above set limit then total alarm external red LED lights ON and buzzer begins to sound. If all sensors input values are below critical set limits then green external LED of the total alarm unit lights ON and system's user is aware that the engine is working fine.

FPGA's clock values act as input and using the appropriate process shown in the APPENDIX of this paper but not in the above flowchart, are converted to time values measured with seconds. Seventh process of Figure 7 uses the above time values as input and activates via logic '1' the yellow external LED of the alarm unit, in case that engine's time period of operation exceeds upper set limit, thus there is a danger for damaging the engine.

IV. CONCLUSION

An FPGA-based industrial engine monitoring and controlling system, is presented in this work. Our system is able to monitor and control simultaneously four basic factors playing decisive role in engine's smooth operation. Vibration, internal magnetic field spike fluctuations, humidity and temperature, are monitored by our system and corresponding external LEDs are activated for each sensor, presenting good or false operation. A LED and buzzer alarm unit are also activated for presenting overall system operation status, depending on whether all input sensor values are within set limits or at least one of them measures dangerous input values for engine operation. All input sensor values are converted in FPGA's ADCs and the results are displayed in seven-segment displays. Our system also provides an additional engine protection by informing user whether time period of operation for the specific engine is above set limit, thus engine must pause its operation to avoid malfunction or destruction. Yellow external LED lights ON in the above case. The system can work with all commercial sensors that provide an analog output and it is easy to be manufactured, providing also the benefit of low cost. Additionally, all output logic signals for lighting LEDs and buzzer, could be sent to an AI system to achieve simultaneous monitor and control of a large number of engines.

REFERENCES

[1]. M. Yussup, M.M. Ibrahim, L. Lombigit, N.A.A. Rahman and M.R.M. Zin, "Implementation of data acquisition interface using on-board field-programmable gate array (FPGA) universal serial bus (USB) link", Advanc. in Nuclear Research and Energy Development, AIP Conf. Proc. 1584, 69-72 (2014), doi: 10.1063/1.4866106

[2]. A. Gujar, International Journal of Computer Science and Information Technologies, "Image Encryption using AES Algorithm based on FPGA", vol 5, (5), 2014, 6853-6859

[3]. S. Singh, A.K. Saini, R. Saini, I.J. Image, Graphics and Signal Processing, "Interfacing the Analog Camera with FPGA Board for Real-time Video Acquisition" 2014, 4, 32-38, DOI: 10.5815/ijigsp.2014.04.04

[4]. S. Muthukrishnan and R. Priyadharsini, International Journal of Computer Science and Mobile Computing, "32-Bit RISC and DSP System Design in an FPGA" vol 3, issue 12, Dec. 2014, pg. 361-368

[5]. L. Chen, Y. Chang, L. Yan, IEEE Transactions on Geoscience and Remote Sensing, "On-orbit real-time variational image destriping: FPGA architecture and implementation", 10.1109/tgrs.2022.3140428, 2022, pp. 1-1

[6]. S. Yarlagadda, S. Kaza, A. Tummala, E. Babu, R. Prabhakar, Information Technology in Industry, "The reduction of Crosstalk in VLSI due to parallel bus structure using Data Compression Bus Encoding technique implemented on Artix 7 FPGA Architecture", 10.17762/itii.v9i1.151, 2021, Vol 9 (1), pp. 456-460

[7]. C. Du, Y. Yamaguchi, Electronics, "High-Level Synthesis Design for Stencil Computations on FPGA with High Bandwidth Memory", 2020, 9(8), 1275; https://doi.org/10.3390/electronics9081275

DOI: 10.17148/IJIREEICE.2026.14501

[8]. X. Hao, C. Lin and Q. Wu, Electronics, "A Parallel Timing Synchronization Structure in Real-Time High Transmission Capacity Wireless Communication Systems", 2020, 9(4), 652; https://doi.org/10.3390/electronics9040652

[9]. P. A. Bawiskar, R.K. Agrawal, International Journal of Innovative Research in Science, Engineering and Technology, "FPGA Based Home Security System" vol.4, issue 12, Dec. 2015, p. 12865-12869, DOI: 10.15680/IJIRSET.2015.0412139

[10]. K. Saroch, A. Sharma, IOSR Journal of Electronics and Communication Engineering, "FPGA Based System Login Security Lock Design Using Finite State Machine" vol 5, issue 3, Mar.-Apr. 2013, pp 70-75

[11]. R.S. Parikh, Int. Journal of Engineering Research and Application, "Alarm System Implementation on Field Programmable Gate Array" vol 8, issue 1, Jan. 2018, pp 01-04.

[12]. E. I. Dimitriadis and L. Dimitriadis, "A Simple, Low Cost and Multiple Input Alarm System, Functioning as Finite State Machine (FSM), Using VHDL and FPGAs", Journal of Active and Passive Electronic Devices, vol. 17, pp. 307–315, 2024.

[13]. E. I. Dimitriadis and L. Dimitriadis, "A g-sensor based alarm system, for multiple tilt sensor applications, using VHDL and FPGAs", Journal of Active and Passive Electronic Devices, vol. 18, pp. 119-129, 2024.

[14]. Eric Monmasson, Lahoucine Idkhajine, Cirstea Marcian N, Imen Bahri, Tisan Alin, Naouar Mohammed Wissem, "FPGAs in industrial control applications", IEEE Transactions on Industrial Informatics, 2016, 7(2), pp.224-243.

[15]. Muthukumaran Vaithianathan, Shivakumar Udkar, Deepanjan Roy, Manjunath Reddy, Senkadir Rajasekaran, "FPGA-Based Motor Control Systems for Industrial Automation", 2024 International Conference on Sustainable Communication Networks and Application (ICSCNA), December 2024.

[16]. Claudio Rubattu, Antonio Ledda, Francesco Ratto, Chaitanya Jugade, Dip Goswami, and Francesca Palumbo, "Integrating FPGA-Based Acceleration in Industrial Motion Control System", IEEE Open Journal of the Industrial electronics Society, vol. 6, 2025 pp. 898-914.

APPENDIX

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
use ieee.std_logic_unsigned.ALL;

entity DE10_Lite_ADC_industrial_sensors is
generic(ClockFrequencyHz : integer:=50000000;
wait_count : natural := 1250000);  -- 50000000=1sec

port
(
rst : in std_logic; --SW8
nRst    : in std_logic; -- Negative reset
Seconds : inout integer;
led1: out std_logic;
led2: out std_logic;
led3: out std_logic;
led4: out std_logic;
led5: out std_logic;
led6: out std_logic;
led7: out std_logic;
led8: out std_logic;
led9: out std_logic;
led10: out std_logic;
led_green_vibr: buffer std_logic;
led_red_vibr: buffer std_logic;
led_green_out_vibr: out std_logic;
led_red_out_vibr: out std_logic;
led_green_hall: buffer std_logic;
led_red_hall: buffer std_logic;
led_green_out_hall: out std_logic;
```

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

```
led_red_out_hall: out std_logic;
led_green_hum: buffer std_logic;
led_red_hum: buffer std_logic;
led_green_out_hum: out std_logic;
led_red_out_hum: out std_logic;
led_green_temp: buffer std_logic;
led_red_temp: buffer std_logic;
led_green_out_temp: out std_logic;
led_red_out_temp: out std_logic;
buzzer:out std_logic;
buzzer_buff: buffer std_logic;
Vvibr: buffer integer;
Vhall: buffer integer;
Vhum: buffer integer;
Vtemp: buffer integer;

d2abuf :buffer integer range 0 to 9;
d1abuf :buffer integer range 0 to 9;
d0abuf :buffer integer range 0 to 9;
d2bbuf :buffer integer range 0 to 9;
d1bbuf :buffer integer range 0 to 9;
d0bbuf :buffer integer range 0 to 9;
d2cbuf :buffer integer range 0 to 9;
d1cbuf :buffer integer range 0 to 9;
d0cbuf :buffer integer range 0 to 9;
d2dbuf :buffer integer range 0 to 9;
d1dbuf :buffer integer range 0 to 9;
d0dbuf :buffer integer range 0 to 9;
SW0 : in std_logic;

red_led_alarm_out : out std_logic;
yellow_led_alarm_out: out std_logic;
green_led_no_alarm_out: out std_logic;
red_led_alarm_buff : buffer std_logic;
yellow_led_alarm_buff: buffer std_logic;
green_led_no_alarm_buff: buffer std_logic;
-- Clocks
ADC_CLK_10: in std_logic;
MAX10_CLK1_50: in std_logic;
MAX10_CLK2_50: in std_logic;
-- KEYs
KEY: in std_logic_vector(1 downto 0);
-- HEX
HEX0: out std_logic_vector(7 downto 0);
HEX1: out std_logic_vector(7 downto 0);
HEX2: out std_logic_vector(7 downto 0);
HEX3: out std_logic_vector(7 downto 0);
HEX4: out std_logic_vector(7 downto 0);
HEX5: out std_logic_vector(7 downto 0);

ARDUINO_IO: inout std_logic_vector(15 downto 0);
ARDUINO_RESET_N: inout std_logic);
end entity;
architecture DE10_Lite_ADC_industrial_sensors_Arch of DE10_Lite_ADC_industrial_sensors is

-- Analog to Digital Converter IP core
component myADC is
port
```

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

```
(
clk_clk: in std_logic := 'X';
modular_adc_0_command_valid: in std_logic := 'X';
modular_adc_0_command_channel: in std_logic_vector(4 downto 0) := (others => 'X');
modular_adc_0_command_startofpacket: in std_logic := 'X';
modular_adc_0_command_endofpacket: in std_logic := 'X';
modular_adc_0_command_ready: out std_logic;
modular_adc_0_response_valid: out std_logic;
modular_adc_0_response_channel: out std_logic_vector(4 downto 0);
modular_adc_0_response_data: out std_logic_vector(11 downto 0);
modular_adc_0_response_startofpacket: out std_logic;
modular_adc_0_response_endofpacket: out std_logic;
reset_reset_n: in std_logic
);
end component myADC;
signal modular_adc_0_command_valid: std_logic;
signal modular_adc_0_command_channel: std_logic_vector(4 downto 0);
signal modular_adc_0_command_startofpacket: std_logic;
signal modular_adc_0_command_endofpacket: std_logic;
signal modular_adc_0_command_ready: std_logic;
signal modular_adc_0_response_valid: std_logic;
signal modular_adc_0_response_channel: std_logic_vector(4 downto 0);
signal modular_adc_0_response_data: std_logic_vector(11 downto 0);
signal modular_adc_0_response_startofpacket: std_logic;
signal modular_adc_0_response_endofpacket: std_logic;
signal clk_clk: std_logic;
signal reset_reset_n: std_logic;
type state_machines is (sm0, sm1, sm2, sm3, sm4);
signal sm: state_machines;
-- signals to store conversion results
signal ADCIN1, ADCIN2, ADCIN3, ADCIN4: std_logic_vector(11 downto 0);
signal AD1, AD2, AD3, AD4: std_logic_vector(11 downto 0);
-- signals for BCD digits
signal digit2a, digit1a, digit0a: std_logic_vector(3 downto 0);
signal digit2b, digit1b, digit0b: std_logic_vector(3 downto 0);
signal digit2c, digit1c, digit0c: std_logic_vector(3 downto 0);
signal digit2d, digit1d, digit0d: std_logic_vector(3 downto 0);
signal digit5, digit4, digit3, digit2, digit1, digit0: std_logic_vector(3 downto 0);
-- signal to determine how fast the
-- 7-seg displays will be updated
signal cnt: integer;
signal state_LED: std_logic;
signal state_Vvibr: integer;
signal state_Vhall: integer;
signal state_Vhum: integer;
signal state_Vtemp: integer;
signal Ticks : integer;
signal count : natural range 0 to wait_count;


begin
-- ADC port map
adc1: myADC port map
(
modular_adc_0_command_valid => modular_adc_0_command_valid,
modular_adc_0_command_channel => modular_adc_0_command_channel,
modular_adc_0_command_startofpacket => modular_adc_0_command_startofpacket,
modular_adc_0_command_endofpacket => modular_adc_0_command_endofpacket,
```

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

```
modular_adc_0_command_ready => modular_adc_0_command_ready,
modular_adc_0_response_valid => modular_adc_0_response_valid,
modular_adc_0_response_channel => modular_adc_0_response_channel,
modular_adc_0_response_data => modular_adc_0_response_data,
modular_adc_0_response_startofpacket => modular_adc_0_response_startofpacket,
modular_adc_0_response_endofpacket => modular_adc_0_response_endofpacket,
clk_clk => clk_clk,
reset_reset_n => reset_reset_n
);
clk_clk <= MAX10_CLK1_50;
reset_reset_n <= KEY(0);
-- process for reading new samples
p1: process(reset_reset_n, clk_clk)
begin
if reset_reset_n = '0' then
        sm <= sm0;
elsif rising_edge(clk_clk) then
        case sm is
                when sm0 =>
                        sm <= sm1;
                        modular_adc_0_command_valid <= '1';
                        modular_adc_0_command_channel <= "00001";
                when sm1 =>
                        if modular_adc_0_response_valid = '1' then
                                modular_adc_0_command_channel <= "00010";
                                ADCIN4 <= modular_adc_0_response_data;
                                sm <= sm2;
                        end if;
                when sm2 =>
                        if modular_adc_0_response_valid = '1' then
                                --modular_adc_0_command_channel <= "00001";
                                modular_adc_0_command_channel <= "00011";
                                ADCIN1 <= modular_adc_0_response_data;
                                --sm <= sm1;
                                sm <= sm3;
                        end if;
                when sm3 =>
                        if modular_adc_0_response_valid = '1' then
                                modular_adc_0_command_channel <= "00100";
                                ADCIN2 <= modular_adc_0_response_data;
                                sm <= sm4;
                        end if;
                when sm4 =>
                        if modular_adc_0_response_valid = '1' then
                                modular_adc_0_command_channel <= "00001";
                                ADCIN3 <= modular_adc_0_response_data;
                                sm <= sm1;
                        end if;
                when others =>
        end case;
end if;
end process;

-- process for conversion from binary to BCD (first analog voltage_vibration)
p2: process(AD1, d2abuf, d1abuf, d0abuf)
variable vin: integer;
variable d2, d1, d0: integer;
begin
```

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

```
vin := to_integer(unsigned(std_logic_vector(to_unsigned(to_integer(unsigned(AD1)) * 500,
32))(31 downto 12)));
d2 := (vin / 100)/4;--/4 for calibration
d1 := vin mod 100 / 10;
d0 := ((vin mod 100) mod 10);
digit2a <= std_logic_vector(to_unsigned(d2, 4));
digit1a <= std_logic_vector(to_unsigned(d1, 4));
digit0a <= std_logic_vector(to_unsigned(d0, 4));
d2abuf<= d2;
d1abuf<= d1;
d0abuf<= d0;
end process;
-- process for conversion from binary to BCD (second analog voltage_Hall)
p3: process(AD2,d2bbuf,d1bbuf,d0bbuf)
variable vin: integer;
variable d2, d1, d0: integer;
begin
vin := to_integer(unsigned(std_logic_vector(to_unsigned(to_integer(unsigned(AD2)) * 500,
32))(31 downto 12)));
d2 := vin / 100;
d1 := vin mod 100 / 10;
d0 := ((vin mod 100) mod 10);
digit2b <= std_logic_vector(to_unsigned(d2, 4));
digit1b <= std_logic_vector(to_unsigned(d1, 4));
digit0b <= std_logic_vector(to_unsigned(d0, 4));
d2bbuf<= d2;
d1bbuf<= d1;
d0bbuf<= d0;
end process;
-- process for conversion from binary to BCD (third analog voltage_humidity)
p5: process(AD3,d2cbuf,d1cbuf,d0cbuf)
variable vin: integer;
variable d2, d1, d0: integer;
begin
vin := to_integer(unsigned(std_logic_vector(to_unsigned(to_integer(unsigned(AD3)) * 500,
32))(31 downto 12)));
d2 := (vin / 100)/2; --/2 for calibration
d1 := vin mod 100 / 10;
d0 := ((vin mod 100) mod 10);
digit2c <= std_logic_vector(to_unsigned(d2, 4));
digit1c <= std_logic_vector(to_unsigned(d1, 4));
digit0c <= std_logic_vector(to_unsigned(d0, 4));
d2cbuf<= d2;
d1cbuf<= d1;
d0cbuf<= d0;
end process;
-- process for conversion from binary to BCD (fourth analog voltage_temperature)
p6: process(AD4,d2dbuf,d1dbuf,d0dbuf)
variable vin: integer;
variable d2, d1, d0: integer;
begin
vin := to_integer(unsigned(std_logic_vector(to_unsigned(to_integer(unsigned(AD4)) * 500,
32))(31 downto 12)));
d2 := (vin / 100)/2;--/2 for calibration
d1 := vin mod 100 / 10;
d0 := ((vin mod 100) mod 10);
digit2d <= std_logic_vector(to_unsigned(d2, 4));
digit1d <= std_logic_vector(to_unsigned(d1, 4));
```

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

```
digit0d <= std_logic_vector(to_unsigned(d0, 4));
d2dbuf<= d2;
d1dbuf<= d1;
d0dbuf<= d0;
end process;
state_Vvibr<= (d2abuf*100)+(d1abuf*10)+(d0abuf);
Vvibr<= state_Vvibr;
state_Vhall<= (d2bbuf*100)+(d1bbuf*10)+(d0bbuf);
Vhall<= state_Vhall;
state_Vhum<= (d2cbuf*100)+(d1cbuf*10)+(d0cbuf);
Vhum<= state_Vhum;
state_Vtemp<= (d2dbuf*100)+(d1dbuf*10)+(d0dbuf);
Vtemp<= state_Vtemp;
-- determine how fast the 7-seg displays will be updated
p4: process(reset_reset_n, clk_clk)
begin
if reset_reset_n = '0' then
cnt <= 0;
elsif rising_edge(clk_clk) then
if cnt < 20_000_000 then
cnt <= cnt + 1;
else
cnt <= 0;
AD1 <= ADCIN1;
AD2 <= ADCIN2;
AD3 <= ADCIN3;
AD4 <= ADCIN4;
end if;
end if;
end process;
--time-seconds
process(MAX10_CLK1_50) is
   begin
      if rising_edge(MAX10_CLK1_50) then -- IF CLK'EVENT AND CLK='1'
         -- If the negative reset signal is active
         if nRst = '0' then
            Ticks   <= 0;
            Seconds <= 0;
         else
            -- True once every second
            if Ticks = ClockFrequencyHz - 1 then
               Ticks <= 0;
                  Seconds <= Seconds + 1;
            else
               Ticks <= Ticks + 1;
            end if;
         end if;
      end if;
   end process;
--critical Vvibr value exceeded board LEDs and external red led lights up
--(100->1Volt)
process(state_LED,led_red_vibr,Vvibr, led_green_vibr)
begin
  IF Vvibr>=200 THEN
    led_red_vibr <= '1';
            state_LED <= '1';
    led_green_vibr <= '0';
    else
```

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

```
   led_red_vibr <= '0';
            state_LED <= '0';
            led_green_vibr <= '1';
  end if;
end process;
led_red_out_vibr<= led_red_vibr;
led_green_out_vibr<= led_green_vibr;
led1 <= state_LED;
led2 <= state_LED;
led3 <= state_LED;
led4 <= state_LED;
led5 <= state_LED;
led6 <= state_LED;
led7 <= state_LED;
led8 <= state_LED;
led9 <= state_LED;
led10 <= state_LED;
--Time-seconds exceeds critical value of industrial engine operation then yellow led lights up
process(yellow_led_alarm_buff,Seconds)
begin
  IF Seconds>=60 THEN
    yellow_led_alarm_buff <= '1';
    else
    yellow_led_alarm_buff <= '0';
  end if;
end process;
yellow_led_alarm_out<= yellow_led_alarm_buff;
--critical Vhall value exceeded
process(led_green_hall,led_red_hall,Vhall)
begin
  IF Vhall>=300 THEN
    led_red_hall <= '1';
    led_green_hall <= '0';
    else
    led_red_hall <= '0';
            led_green_hall <= '1';
  end if;
end process;
led_red_out_hall<= led_red_hall;
led_green_out_hall<= led_green_hall;
--critical Vhum value exceeded
process(led_green_hum,led_red_hum,Vhum)
begin
  IF Vhum>=300 THEN
    led_red_hum <= '1';
    led_green_hum <= '0';
    else
    led_red_hum <= '0';
            led_green_hum <= '1';
  end if;
end process;
led_red_out_hum<= led_red_hum;
led_green_out_hum<= led_green_hum;
--critical Vtemp value exceeded
process(led_green_temp,led_red_temp,Vtemp)
begin
  IF Vtemp>=300 THEN
    led_red_temp <= '1';
```

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

```
   led_green_temp <= '0';
   else
   led_red_temp <= '0';
            led_green_temp <= '1';
  end if;
end process;
led_red_out_temp<= led_red_temp;
led_green_out_temp<= led_green_temp;
--alarm level
process (Vvibr,Vhall,Vhum,Vtemp, green_led_no_alarm_buff, red_led_alarm_buff)
begin
if (Vvibr>=200 OR Vhall>=300 OR Vhum>=300 OR Vtemp>=300) THEN
red_led_alarm_buff <= '1';
green_led_no_alarm_buff <= '0';
end if;
if (Vvibr<200 AND Vhall<300 AND Vhum<300 AND Vtemp<300) THEN
red_led_alarm_buff <= '0';
green_led_no_alarm_buff <= '1';
end if;
end process;
red_led_alarm_out <= red_led_alarm_buff;
green_led_no_alarm_out <= green_led_no_alarm_buff;
 -- buzzer sounds
Process(buzzer_buff,Vvibr,Vhall,Vhum,Vtemp)
BEGIN
IF (Vvibr>=200 OR Vhall>=300 OR Vhum>=300 OR Vtemp>=300) THEN
buzzer_buff <= '1';
end if;
IF (Vvibr<200 AND Vhall<300 AND Vhum<300 AND Vtemp<300) THEN
buzzer_buff <= '0';
end if;
end process;
buzzer<= buzzer_buff;
process( digit2a,digit1a,digit0a, digit2c, digit1c, digit0c, SW0,
                              digit2b,digit1b,digit0b, digit2d, digit1d, digit0d)
begin
  IF SW0='0' THEN
  digit5 <= digit2a;
  digit4 <= digit1a;
  digit3 <= digit0a;
  digit2 <= digit2b;
  digit1 <= digit1b;
  digit0 <= digit0b;
  elsif SW0='1' THEN
  digit5 <= digit2c;
  digit4 <= digit1c;
  digit3 <= digit0c;
  digit2 <= digit2d;
  digit1 <= digit1d;
  digit0 <= digit0d;
  end if;
end process;
  WITH digit5 SELECT
HEX5 <= "01000000" WHEN "0000", -- display 0
"01111001" WHEN "0001", -- display 1
"00100100" WHEN "0010", -- display 2
"00110000" WHEN "0011", -- display 3
"00011001" WHEN "0100", -- display 4
```

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

```
"00010010" WHEN "0101", -- display 5
"00000011" WHEN "0110", -- display 6
"01111000" WHEN "0111", -- display 7
"00000000" WHEN "1000", -- display 8
"00011000" WHEN "1001", -- display 9
"01111111" WHEN OTHERS; -- blank display
WITH digit4 SELECT
HEX4 <= "11000000" WHEN "0000", -- display 0
"11111001" WHEN "0001", -- display 1
"10100100" WHEN "0010", -- display 2
"10110000" WHEN "0011", -- display 3
"10011001" WHEN "0100", -- display 4
"10010010" WHEN "0101", -- display 5
"10000011" WHEN "0110", -- display 6
"11111000" WHEN "0111", -- display 7
"10000000" WHEN "1000", -- display 8
"10011000" WHEN "1001", -- display 9
"11111111" WHEN OTHERS; -- blank display
WITH digit3 SELECT
HEX3 <= "11000000" WHEN "0000", -- display 0
"11111001" WHEN "0001", -- display 1
"10100100" WHEN "0010", -- display 2
"10110000" WHEN "0011", -- display 3
"10011001" WHEN "0100", -- display 4
"10010010" WHEN "0101", -- display 5
"10000011" WHEN "0110", -- display 6
"11111000" WHEN "0111", -- display 7
"10000000" WHEN "1000", -- display 8
"10011000" WHEN "1001", -- display 9
"11111111" WHEN OTHERS; -- blank display

 WITH digit2 SELECT
HEX2 <= "01000000" WHEN "0000", -- display 0
"01111001" WHEN "0001", -- display 1
"00100100" WHEN "0010", -- display 2
"00110000" WHEN "0011", -- display 3
"00011001" WHEN "0100", -- display 4
"00010010" WHEN "0101", -- display 5
"00000011" WHEN "0110", -- display 6
"01111000" WHEN "0111", -- display 7
"00000000" WHEN "1000", -- display 8
"00011000" WHEN "1001", -- display 9
"01111111" WHEN OTHERS; -- blank display
WITH digit1 SELECT
HEX1 <= "11000000" WHEN "0000", -- display 0
"11111001" WHEN "0001", -- display 1
"10100100" WHEN "0010", -- display 2
"10110000" WHEN "0011", -- display 3
"10011001" WHEN "0100", -- display 4
"10010010" WHEN "0101", -- display 5
"10000011" WHEN "0110", -- display 6
"11111000" WHEN "0111", -- display 7
"10000000" WHEN "1000", -- display 8
"10011000" WHEN "1001", -- display 9
"11111111" WHEN OTHERS; -- blank display
WITH digit0 SELECT
HEX0 <= "11000000" WHEN "0000", -- display 0
"11111001" WHEN "0001", -- display 1
```

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 5, May 2026

DOI: 10.17148/IJIREEICE.2026.14501

```
"10100100" WHEN "0010", -- display 2
"10110000" WHEN "0011", -- display 3
"10011001" WHEN "0100", -- display 4
"10010010" WHEN "0101", -- display 5
"10000011" WHEN "0110", -- display 6
"11111000" WHEN "0111", -- display 7
"10000000" WHEN "1000", -- display 8
"10011000" WHEN "1001", -- display 9
"11111111" WHEN OTHERS; -- blank display
end architecture;
```

© IJIREEICE This work is licensed under a Creative Commons Attribution 4.0 International License