Logic Design for Linear Regression Model Using ASIC in Engine Oil Degradation Monitoring System

M. F. M. Idros[1,2], A. H. A. Razak[1,2], M. F. Azmi[1,2], S.I. Suliman[2], A. K. Halim[1,2], N. Khairudin[1,2]
[1]*Electronic Architecture and Application (EArA) Research Group*
[2]*Faculty of Electrical Engineering,*
Universiti Teknologi MARA Malaysia, 40450 Shah Alam, Selangor, Malaysia
faizul_san287@salam.uitm.edu.my

***Abstract*—A degradation analysis in automotive engine oil is concerned with the unrespectable cost of equipment for data storage. System-on-Chip gives possible cost effective in reducing the bulky equipment and reliance on labor. This article discusses a new technique of degradation monitoring where an engine oil degradation model is used and translated into the logic gate based on the Least Square Method of statistical analysis. The degradation model is based on the optical properties where the percentage transmittance of light is varied due to the increase of contaminates contents in the engine oil at a certain period. A linear regression model is chosen in register-transfer level (RTL) development of the digital circuit design. In the algorithm development, the data set are collected at every one hour up to 300 hours and stored in a temporary register. Linear regression is implemented at every 5 data to obtain the degraded condition based on the variation of the slope.**

***Index Terms*—Oil Condition Monitoring; Resistive Sensor; Indium Tin Oxide; Oil Degradation Sensing.**

I. INTRODUCTION

Vehicle engine uses lubricant to lubricate, seal, cool and clean the metal parts in order to maintain the performance of the engine [1]. Two main materials namely base oil and the chemical additives are blended together and resulting in the grade and specific duty. The additives contain the metallic detergents, ashless dispersants, zinc dithiophosphate, anti-oxidant/anti-wear, friction modifier, antifoam and pour point depressants. While the base oil does not wear out, the lubricant will be degraded by itself due to the chemical reaction between the additives and the surrounding parts in engines after a certain limit of time [2]. Total Acid Number (TAN), Total Base Number (TBN) and viscosity have been identified as the common analysis to determine the remaining quality of engine oil. From these three parameters, TAN has been considered as the most important indicator of engine oil quality [3],[14]. It is because the reaction between oil and oxygen at an elevated temperature made an oxidation between them to form a variety of carbonyl compound. The carbonyl compound that normally containing esters, ketones and carboxylic acid will accelerate the oxidation process and increase the acidity of the engine oil [2, 4]. This will cause the increasing value of TAN up to the warning limit.

Currently, the degradation monitoring of engine oil is monitored by using an offline equipment and data analysis. The samples of data are collected using equipment and passed to a laboratory for analysis. The capability of an optical sensor to detect the variation of covalent bonding at a certain optical wavelength makes the optical spectroscopy is widely used in lubricant analysis. Oxidation compound analysis including viscosity, Total Acid Number, ZDP depletion and hydroxyl band can be seen at band location of $1050cm^{-1}$ – $1730cm^{-1}$ [5-7]. The problem of this offline analysis was the result of the analysis was turning up later and it could very hard to get the relation between the data with the actual running-in machine at the sampling time [8].

The high maintenance cost of equipment and a reliance on labor led to the sensor development for the online monitoring system. There were many types of sensors have been introduced recently. A sensor to monitor corrosion in lubricating oil had been introduced in [9]. The sensor was developed by multiple resistive copper films with different thickness on the glass substrate. The corrosive material can be detected when there was the variation of the film's resistance electrifying. A new technique to measure the changing content of antifreeze material in lubricant had been proposed in [10]. The concept of Zinc Alkyldithiophosphate (ZDP) as an anti-wear additive which can react with antifreeze material made the sensor was realized by using Alumina plates. A sensor that made from Alumina plates have been introduced where it operated by measuring the resistance value of engine oil samples.

The need of an intelligent system to decide on the time to change the engine oil is the main motivation of this research. Although there are extensive researches done in this area, most do not have a standalone system to tell the warning or critical limit of the engine oil, hence has no ability to predict the condition of the engine oil. Most of the current monitoring techniques are used as offline analysis where it requires bulky equipment and the user had to have a technical knowledge in lubricant technology to make a decision making.

II. METHODOLOGY

In this work, the samples of engine oil being used are SAE 10W40, 20W/50, 15W/40, CD-SE50 and SJ15W/40 (diesel). Optical properties the listed oil have been converted to binary code for simulation and implemented on FPGA for hardware validation. FPGA is normally used in any digital design as reported in [15-20], before converted into physical design. Figure 1 shows the flow of overall works in logic design generation using Quartus II and layout preparation.

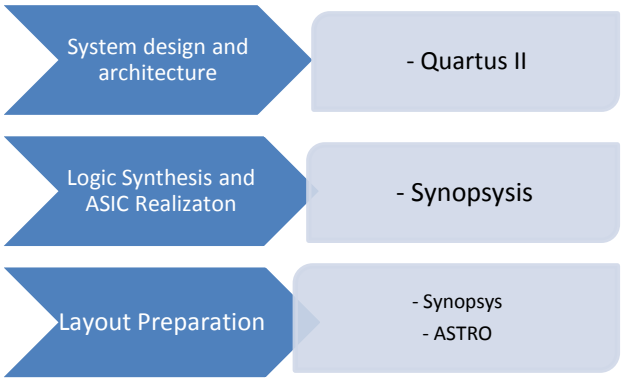

Figure 1: Logic Design generation and ASIC preparation work flow

This is the back-end system design based the optical data input. The front-end input is the optical sensor with the analog to digital converter. The back-end system developments have been performed using Quartus II software environment where the algorithm is coded in Verilog language. Quartus The LSM is the linear regression which is capable to be used as the prediction algorithm. The LSM is written in Verilog language to generate the Register Transfer Level (RTL) where the logic gates are generated. The system consists of clock divider, arithmetic and logic unit, register and input/output unit as shown in Figure 2. The system is composed of five blocks, namely, clock divider, data input processor, arithmetic unit, prediction and decision maker, and a display unit. The input data is fed from *Data in* in 9-bit binary number representing the percentage of transmittance (%*T*) in the *y*-axis, whereas the hours in the *x*-axis is taken from the *Clock Divider* input. The 9-bit as an n-bit data are chosen to be counted at the maximum number 2^n of 512 that represent the counting hours. *Clock Divider* is used to generate the 1-hour pulse because the data are sampled at every 1 hour and have a capability to sample the data up to 512 hours due to the bit count of 9-bit. The maximum generating time is depending on the engine oil condition.

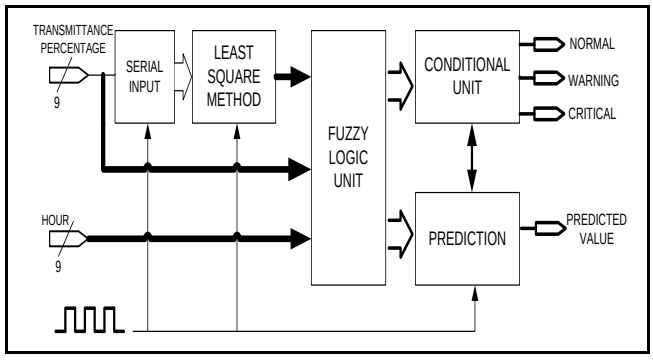

Figure 2: Block diagram of overall design

For logic synthesis analysis, Design Compiler of Synopsys tool has been used by taking the RTL hardware description written in Verilog and standard cell library as an input to the system as shown in Figure 3.

The result of the synthesis process is the technology dependent gate level-netlist. The structural gate level-netlist are represented in the standard cells based on the standard cell library of the synthesis tools.

Lastly, ASTRO tool in Synopsys has been used for layout preparation in this smart SIC controller. It performs the basic timing driven flow for placement, CTS, routing, optimization and timing closure achievement. There were important keys

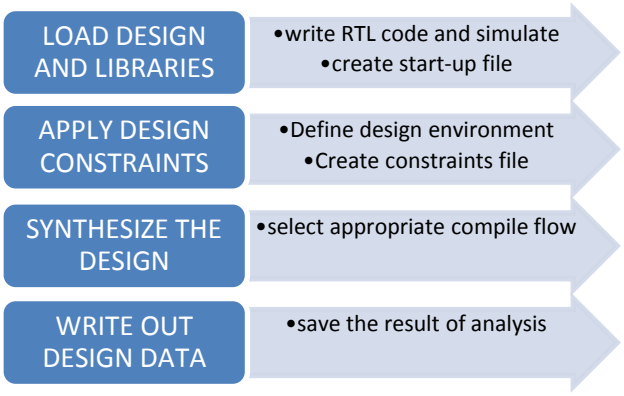

Figure 3: RTL generation flow

of the layout preparation, including connectivity, basic logic elements and sub designs.

III. Result And Discussion

In this work, the capability of ASIC tool in Synopsys has been demonstrated for the design compilation process. Then, optimization process was performed before the fabrication stage for library matching, floor plan setting and area constraint. This is due to the requirement of the IC foundry for any design to be matched with their design standard.

A. *Serial Input*

Figure 4 shows the *serial input* block that received the 9-bit data converted from %T. The input data from both transmittance percentage (%T) and hour (H) is converted into digital forms of 9-bit to get the maximum counter up to 512 hours and the maximum value of 512 for %T. The block consisted of 10 registers and data shifted from the first *register* to the last *register* for arithmetic operation. The time interval for each *t* is 1 hour. The first data *D1* at a time *t1* stored temporarily in a *register*.

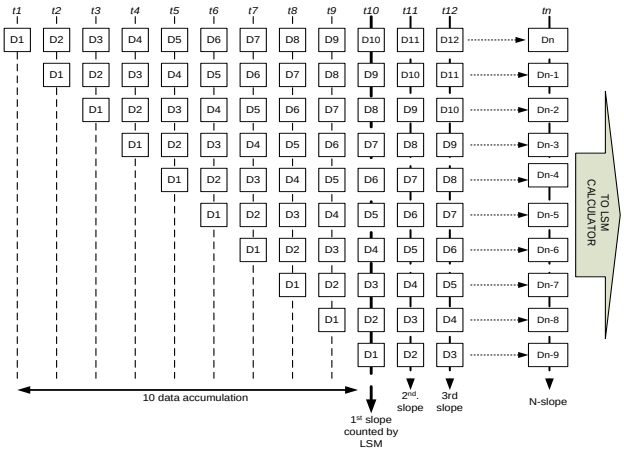

Figure 4: *serial input* block

At *t2*, *D2* shifted down to and replaced by *D2*. This process continues for the next value of *t* and *D* repeatedly. At every ten data, the new value of the slope is produced by using the Least Square method which is fed to the Fuzzy Logic unit for decision-making process.

The FL unit received three types of data, fuzzifies and defuzzified before feeding into the conditional unit. The predicted value is controlled by the conditional unit where it

will appear during the warning limit. All systems shown in Figure 4 are controlled by the clock which is generated every 1 hour. This system has been tested with the actual data and synthesized for gate-level realization.

B. *Data Registers*

The time series data collector is designed to have temporary storage before the arithmetic operation. The 9-bit D Flip-Flop (D-FF) is connected in series and data are fed and shifted to up to 10 D-FF. Figure 5 shows the implementation of the D-FF series as a register in this system. In the beginning of the operating system, the data are shifted to the right and the arithmetic operation starts to activate after 10 data are completely fed into the register memory. Then, the first 10 data become one frame where it will be shifted to the right to obtain new data and will be controlled by a clock.

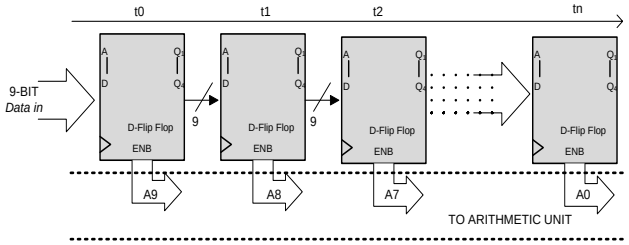

Figure 5: Implementation of the D-FF series as a register

Data in is taken from the ADC, which is not discussed in this article. At $t = 0$, data are taken from virgin oil and will be stored as a reference point. The next $t = 1, \ldots, 10$ are shifted to 10 different FF at 1 h timeout intervals. These stored data will then be replaced by other data from $t = 2, \ldots, 11$ until t_n, where n is the time of the critical limit for the engine oil condition.

The warning limit is set at the time when the engine oil condition given by the degradation slope is decreasing drastically at a certain period. The time to change the engine oil is set when the oxidation value from $\%T$ is decreased by half from the original value.

C. *Arithmetic Unit*

This is the main structure of the system that implementing LSM algorithm to monitor the variation of the transmittance percentage from the engine oil sample. There are two inputs into this arithmetic unit, *Time in* and *Data in*. The arithmetic unit processes the data and performs all of the necessary arithmetic operations to complete the calculation of degradation. Each arithmetic operation is implemented in one sub-block for multiple usage purposes, as shown in Figure 6.

The input data consisted of 9-bit binary which will be providing 512 sequence number. The *Time in* produced the sequence counting hours from 0 hour to 512 hours in binary form while the *Data in* was using the number as the binary form of the transmittance percentage. Both *Time in* and *Data in* were fed from the data registers.

D. *Overall System of the Engine Oil Monitoring and Prediction*

Figure 7 shows the RTL generation for the overall system. Quartus II software has been used to code the algorithm and create the top-level module. This system consists of three main blocks including serial data collector, Least Square block and arithmetic unit. The system is designed to have a fixed interval of time. Here, the interval of samples was a 1 hour per-sampled

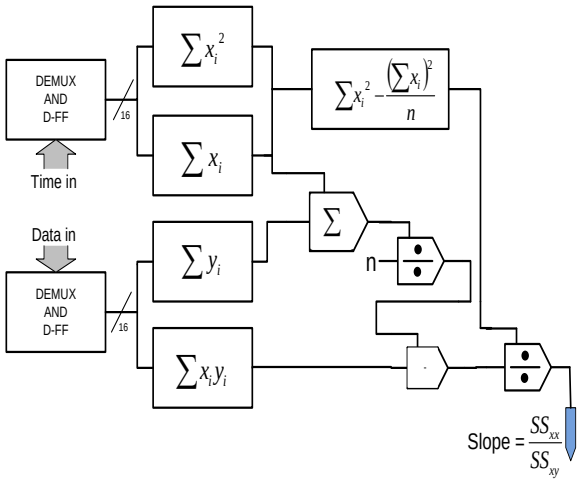

Figure 6: Sub-block of Arithmetic Unit

and each sample representing 16-bit. The interval of data from *Data in* is controlled by *Clock divider* where it will activate the input data at every 1 hour.

The implementation of the algorithm is based on the LSM equation. The input data were programmed to be fed at a constant interval of every 1 hour. The main reason for using the LSM is to obtain the value of the slope where it's able to show the variation degradation at every interval of time. Each value was stored temporarily in a Registers for further process.

For the compact design, this work focuses on a single input where the oxidation parameter is chosen with respect to the running hours. It is because the effects of oxidation of the degradation are higher than the other parameters. Therefore, it is good for the monitoring system to be operated because of the low input parameters and it can reduce the power consumption of the overall system.

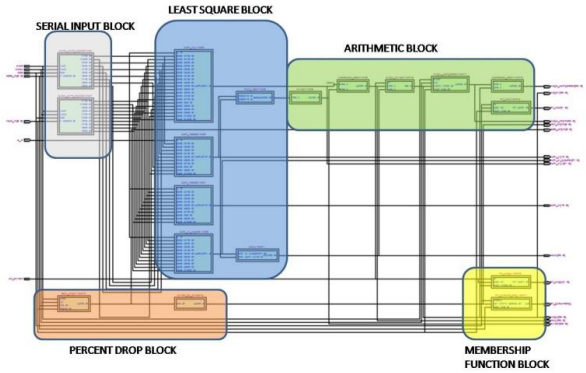

Figure 7: RTL Generation for Overall System

E. *Verification of Monitoring and Prediction with FPGA Implementation*

Here, the features of signal tab in Quartus tools have been used to perform the debugging and verification process. Altera board DE-5 has been used in performing the verification of the structural design of the engine oil monitoring system. The membership function of slope, percentage drop and running hour are the parameters have been monitored in this task where the condition of normal, warning and critical were displayed as a plotted graph on the Quartus window.

The warning limit for engine oil CD-SE15W/40 occurred at the time of 84 hours as shown in Figure 8. Before this stage,

the *predicted_x* for the lifetime prediction shows the unstable time such as 895, 896, 897 and so on. Just after the *warning_on* is activated, the time prediction shows the real prediction. Here, the remaining engine oil lifetime started at 54 hours and decreased until the critical condition is reached. The critical condition reached at 135 hours as shown in Figure 8. At this stage, we can see that the predicted lifetime decreased until 0.

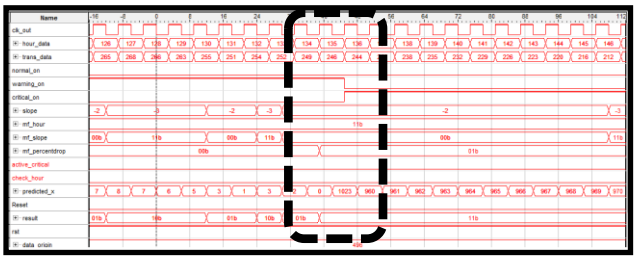

Figure 8: 'Critical condition' reached

F. *ASIC Realization*

Inside the main block is the combination of sub-block generated by the structural design before the optimization process as shown in Figure 9. This process also called as the RTL generation in DC tools for library matching. The synthesis process has been optimized when the block in the system is transformed into the logic gate. In this stage, the design compilation from the Quartus is linked with the internal library of the logic gate. The component from the Quartus environment will be failed to proceed the design compiler stage because of using the internal design. Therefore, the design must be using the common library of the logic gate to be successfully compiled in the DC stage.

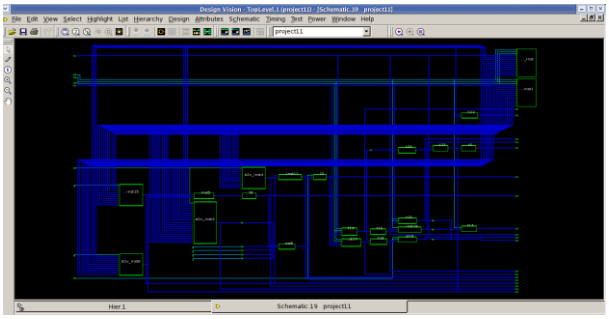

Figure 9: combination of sub-block

G. *Layout Generation*

The floorplan was created as a Milky Way (MW) database design cell which is the starting stage of placement. This task was performed after the timing setup meets the design requirement. The area of floorplan was determined manually by placing the cells and predicts the overall area. In this design, the initial area was set to 20µm^2 x 20µm^2 which is suit to the total amount of the cells placement and routing. Figure 10 shows the sub module layout from the overall design.

The input data located at the left side of the floorplan while the output located at the right-hand side. The net name, pad cell master name, pad cell name and pad pin name are inserted to create the power/ground pad and corner pads in order to create the TDF file for floorplanning setup. At this stage, the cells are connected to the power/ground nets first. This is to ensure the power ground net was completely connected to the cells. To protect the congestion, the standard cell was placed for global routing congestion estimation.

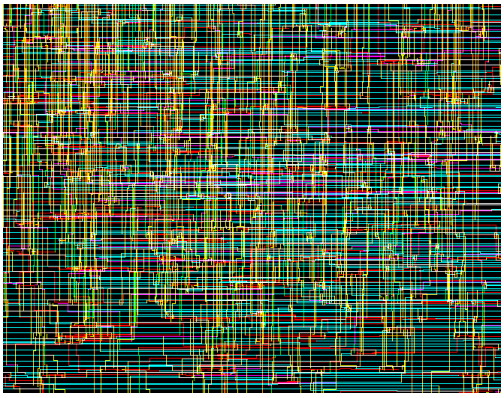

Figure 10: Sub module layout from the overall design.

The functionality and the physical implementation of the design have been given a high priority in the presented work. The approach of LSM in hardware language have been simulated in Quartus II software and implemented on Altera DE2-115 FPGA board. The implemented techniques provide the types of engine oil condition, normal, warning and critical. The degradation monitoring algorithm was implemented and suitable for any type of engine oil because of the interval based monitoring. The objective of the work to get the predictive time to change the engine oil was achieved. The physical implementation up to the post layout level of the system was done successfully using ASIC and ASTRO tools.

IV. CONCLUSION

The monitoring and prediction system was built based on the LSM. The LSM consisted of the combinational logic gate from adder, subtractor, multiplier and divider. Then, it has been combined with another sub-system including serial shift register, membership function, condition monitoring and lifetime prediction. The functional simulation has been tested in Quartus software and then implemented on FPGA for real hardware implementation. The result shows that the usage of engine oil in time-based slope monitoring achieved the objective to prolong the usage of engine oil for more than 100 hours. The LSM was designed with the combination of other system and implemented in FPGA for hardware design. As a result, a user will have a system that will give an alert about the exact time to change their engine oil based on the real parameter such as TAN, TBN, viscosity and oxidation rather than mileage based with the Fuzzy based decision making of time to change the engine oil. Hence, the second objective of this work is achieved successfully.

A significant portion of the work presented in this work has been devoted to the complete implementation of the system on FPGA. The actual data have been used to validate the capability of the overall system in monitoring and predicting the condition of engine oil. This is to fulfill the third objective of this work.

The condition-based monitoring system for engine oil was successfully implemented in the ASIC design flow to achieve the final objective of this work. The model was implemented in hardware description language before being synthesized into the gate-level design. The design was verified and optimized for area, power, and speed by following the standard rules of Silterra 0.18µm technology. The layout of the overall system has been generated for the completion of this work.

ACKNOWLEDGMENT

The author would like to acknowledge Institute of Research Management & Innovation (IRMI) UiTM for the financial support of this research. This research is supported by IRMI under LESTARI Research Grant with project code: 600-RMI/DANA 5/3/LESTARI (45/2015) in completing this work.

REFERENCES

[1] M.F.M.Idros, S. H. Ali, and S. Islam, "Quantitative Analysis of Spectroscopy's Study for Engine Oil Degradation Monitoring Due to Temperature Effect," in *2012 Third International Conference on Intelligent Systems Modelling and Simulation*, 2012, pp. 278 - 282.

[2] M. A. Al-Ghouti and L. Al-Atoum, "Virgin and recycled engine oil differentiation: A spectroscopic study," *Journal of Environmental Management*, vol. 90, pp. 187-195, 2009.

[3] Y. Felkel, N. Dörr, F. Glatz, and K. Varmuza, "Determination of the total acid number (TAN) of used gas engine oils by IR and chemometrics applying a combined strategy for variable selection," *Chemometrics and Intelligent Laboratory Systems*, vol. 101, pp. 14-22, 2010.

[4] S. S. Wang, "Engine oil condition sensor: method for establishing correlation with total acid number," *Sensors and Actuators B: Chemical*, vol. 86, pp. 122-126, 2002.

[5] S. Shirley and S. Donald, "Automotive Engine Oil Condition Monitoring," in *Tribology Data Handbook*, ed: CRC Press, 1997.

[6] M.F.M.Idros, S. H. Ali, and S. Islam, "Optical analysis for condition based monitoring of oxidation degradation in lubricant oil," in *Intelligent and Advanced Systems (ICIAS), 2012 4th International Conference on*, 2012, pp. 735-740.

[7] M.F.M.Idros, S. H. Ali, and S. Islam, "Design of Intelligent SoC Controller for Engine Oil Sensing and Monitoring System," *Asian Journal of Scientific Research*, vol. 5, pp. 70-77, 2012.

[8] P. S. M. Saurabh Kumar, "Online condition monitoring of engine oil," *Industrial Lubrication and Tribology*, vol. 57, pp. 260-267, 2005.

[9] A. Agoston, N. Dörr, and B. Jakoby, "Corrosion sensors for engine oils—laboratory evaluation and field tests," *Sensors and Actuators B: Chemical*, vol. 127, pp. 15-21, 10/20/ 2007.

[10] S. S. Wang and Y. Lin, "A new technique for detecting antifreeze in engine oil during early stage of leakage," *Sensors and Actuators B: Chemical*, vol. 96, pp. 157-164, 11/15/ 2003.

[11] G. Cocco. (2004) Motorcycle Design and Technology Handbook. *Motor Books Workshop*.

[12] M.F.M.Idros, H. Hashim, S. Islam, and S. H. Ali, "Capability of Optical Approach in Condition Based Monitoring of Lubricant Oil," *Sensors & Transducers Journal*, vol. 17, pp. 125-134, 2012.

[13] A. Mujahid and F. Dickert, "Monitoring automotive oil degradation: analytical tools and onboard sensing technologies," *Analytical and Bioanalytical Chemistry*, vol. 404, pp. 1197-1209, 2012/09/01 2012.

[14] B. L. De Rivas, J. Vivancos, J. Ordieres-meré, and S. F. Capuz-rizo, "Determination of the total acid number (TAN) of used mineral oils in aviation engines by FTIR using regression models," *Chemom. Intell. Lab. Syst.*, 2016.

[15] S. A. M. Al Junid, Z. A. Majid, and A. K. Halim, "Development of DNA sequencing accelerator based on Smith Waterman algorithm with heuristic divide and conquer technique for FPGA implementation," in *2008 International Conference on Computer and Communication Engineering*, 2008, pp. 994–996.

[16] S. a. M. Al Junid, M. a. Haron, Z. A. Majid, A. K. Halim, F. N. Osman, and H. Hashim, "Development of Novel Data Compression Technique for Accelerate DNA Sequence Alignment Based on Smith-Waterman Algorithm," in *2009 Third UKSim European Symposium on Computer Modeling and Simulation*, 2009, pp. 181–186.

[17] S. A. M. Al Junid, N. Md Tahir, Z. Abd Majid, Z. Othman, and K. K. Mohd Shariff, "Reducing memory complexity using data minimization technique on FPGA," in *2012 International Conference on Computer & Information Science (ICCIS)*, 2012, pp. 431–434.

[18] S. A. M. Al Junid, Z. A. Majid, and A. K. Halim, "High speed DNA sequencing accelerator using FPGA," in *2008 International Conference on Electronic Design*, 2008, pp. 1–4.

[19] S. A. M. Al Junid, N. M. Tahir, Z. A. Majid, F. N. Osman, and K. K. M. Shariff, "Comparative study for DNA data minimization technique on FPGA," in *2012 4th International Conference on Intelligent and Advanced Systems (ICIAS2012)*, 2012, pp. 765–767.

[20] S. A. M. Al Junid, N. M. Tahir, Z. A. Majid, A. K. Halim, and K. K. M. Shariff, "Improved data minimization technique in reducing memory space complexity for DNA local alignment accelerator application," in *2012 International Symposium on Computer Applications and Industrial Electronics (ISCAIE)*, 2012, pp. 153–156.