

# IMPLEMENTATION OF SENSOR DIAGNOSIS BASED ON AUTOSAR

Banushree<sup>1</sup>, Dr. S Gayathri<sup>2</sup>

**Abstract-** In the automotive industry, increased competition and market regulatory pressures are forcing the original equipment manufacturers (OEMs) to reduce the cost of their products. As a result, focus on cost control and implementation of effective system is expected. In this regard, it is necessary to implement a suitable diagnosis of the sensor circuit based on which automobile is essentially driven. The incorrect value of sensor may lead to erroneous prediction of the situation and that might lead to malfunction of the module and also the noise immunity protections for the read sensor values is also becoming an ever more challenging task to be accomplished.

This paper focuses on the analysis and implementation of sensor diagnosis with an input filter for ADC driver for reliable input, based on AUTOSAR standard. This filter helps to obtain steady analog voltage through input and avoid the effect of sudden high frequency spike that might occur resulting in maintenance of the quality of the designed module for diagnosis of sensor.

**Keywords –** Automotive Open System Architecture (AUTOSAR), Electronic Control Unit (ECU), Micro Controller Abstract Layer (MCAL), Input Output Layer (I/O Layer), Runtime Environment Layer (RTE).

## I. INTRODUCTION

AUTOSAR stands for Automotive Open System Architecture[1]. The enormous development in the state-of-the-art of applications in vehicles directed to the growth of complex drivers for electronic systems in automotive era because of which technical movement is required in design and implementation of automotive electronics systems driver software.

Fig 1 shows the layered architecture of AUTOSAR. The basic layers of AUTOSAR are Application layer, Run time Environment layer , Basic software layer and Micro controller abstraction layer.

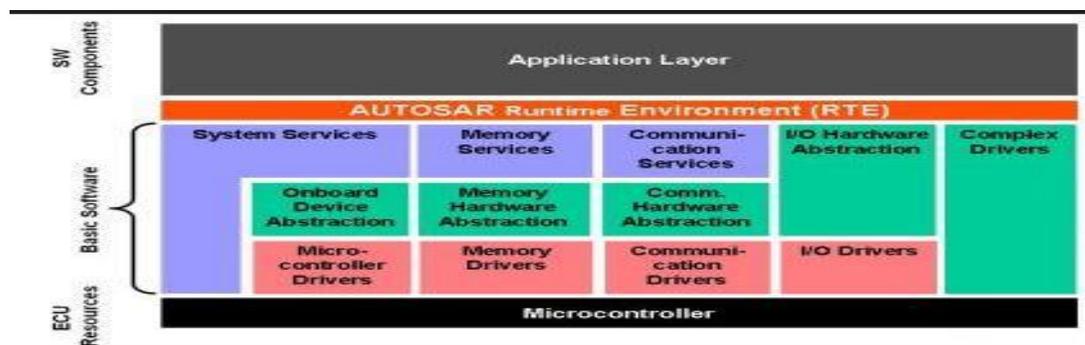


Fig 1 Layered Software Architecture

<sup>1</sup> M Tech, Industrial Electronics, Sri Jayachamarajendra College of Engineering, Mysore, Karnataka, India

<sup>2</sup> Associate professor, E & C, Sri Jayachamarajendra College of Engineering, Mysore, Karnataka, India

Application layer consists of various software modules which are responsible for specific application. RTE layer which acts as interface between the application layer and the basic software layer so that application layer is independent of below layers. BSW layer is responsible for communication between device drivers in MCAL layer and application layer[1]. MCAL layer helps to communicate with the hardware. This layer has software drivers for each device present.

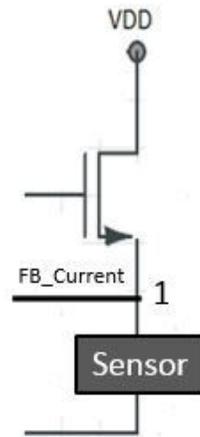
While directing on electronic systems in the automotive domain, the inputs are controlled either by the end user or by a feedback system. These inputs are processed by the controller and are delivered as outputs. The inputs emerge from a myriad of sources like buttons, switches, sensors, relays, and communication devices.

Designing sensitive electronic interface with the inputs coming from unfavourable conditions is a vital feature. Such interfaces can pose a threat to the electronic system which are designed without any protection. Sensors being the key for automotive functionalities, it is very important to develop the noise immunity protections for sensor inputs so that inputs are reliable and diagnostic functionalities are performed based on the obtained reliable input. This paper focuses on AUTOSAR layered development for sensor diagnosis based on reliable input.

## II. PROPOSAL FOR SENSOR DIAGNOSIS

An Analog-Digital Converter (ADC) is a widely used electronic component that converts an analog electric signal (usually a voltage) into a digital representation. This ADC circuit is used to get the value of the sensor via continuous conversion mode. Thus the continuous ADC converted data is used to ensure the integrity of the sensor circuit.

The typical connection of sensor to the microcontroller through high side driver is as shown in the Fig 2.



**Fig 2 Layered Software Architecture**

The feedback current received from the sensor circuit (point 1 in the Fig 2) is analysed in the ADC driver to achieve sensor diagnosis.

In the OFF state of sensor circuit, Short to battery is detected and in ON state Short to ground is detected through received feedback current in ADC driver software.

## III. DESIGN AND IMPLEMENTATION OF SOFTWARE LAYER OF AUTOSAR

The sensor diagnosis is based on AUTOSAR standard where control flows from application layer to the hardware layer. The layers are independent of each other.

### 1. Application layer

The application layer of AUTOSAR where it consists of different software components and each application drives the actuators based on input received from the sensors. This layer gets data from IOHWAB layer through RTE buffers.

The flow of application layer till MCAL layer of AUTOSAR for sensor diagnosis is as shown in Fig 3.

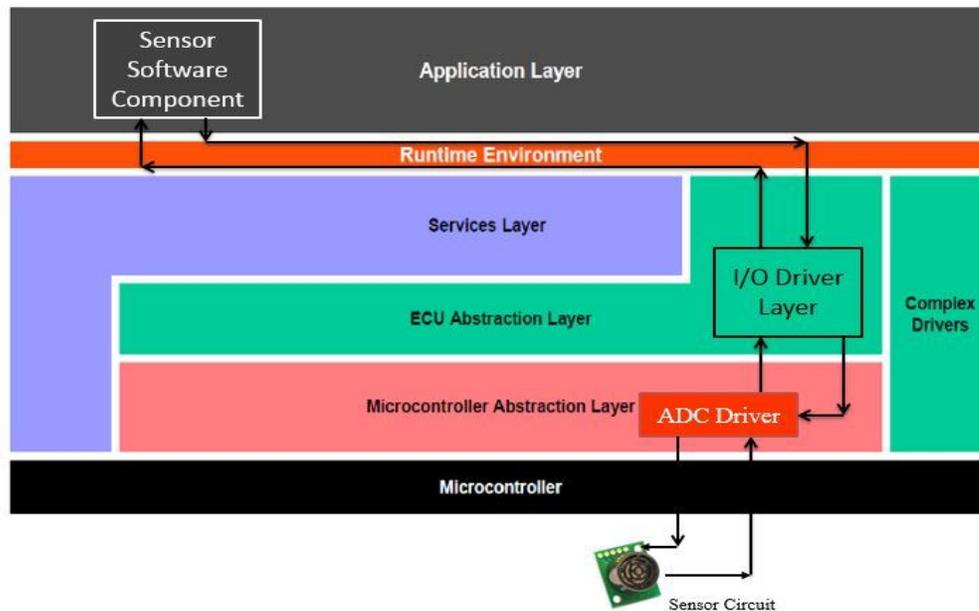


Fig 3 Layered Software Architecture

### 2. IO driver layer :

IOHWAB layer is responsible for receiving/sending data from/to microcontroller abstraction layer drivers[2]. The sensor data is received through ADC driver and this data is stored in the IOHWAB buffer through which application layer reads the value via RTE buffer.

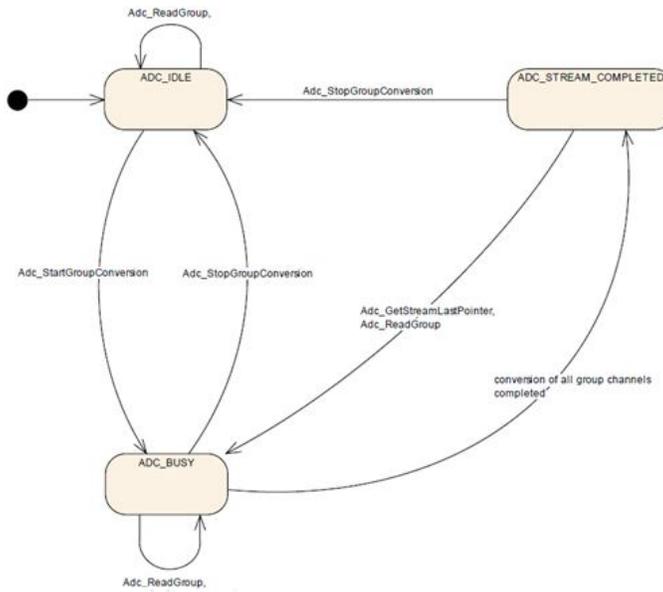
### 3. MCAL ADC driver :

The design describes the functionality, application and the configuration of the AUTOSAR Basic Software module ADC Driver. The ADC module initializes and controls the internal Analogue Digital Converter Unit(s) of the microcontroller. It provides services to start and stop a conversion respectively to enable and disable the trigger source for a conversion. Furthermore it provides services to query the status and result of a conversion. The ADC module works on ADC Channel Groups, which are built from ADC Channels.

An ADC Channel Group combines an analogue input pin (ADC Channel), the needed ADC circuitry itself and conversion result register into an entity that can be individually controlled[3].

## IV. SOFTWARE FLOW OF ADC DRIVER

Once there is an ADC trigger, ADC driver starts converting value of analog channels configured and it is done only if ADC is in idle state[3]. The software flow of ADC driver is as shown in Fig 4.



**Fig 4 Software flow of the ADC driver**

## V. ANALYSIS AND IMPLEMENTATION OF INPUT FILTER FOR SENSOR DATA

In ADC device driver software, a filter is a program or section of code that is designed to examine each input request for certain qualifying criteria and then process or forward it accordingly. The analog low-pass filter can remove high frequency noise and interference from the signal path prior to the ADC conversion to help avoid contaminating the signal with aliased noise. In case of input overvoltage, the analog filter also limits the input current and attenuates the input voltage. Thus, it can protect the ADC's input circuitry.

Noise in ADC conversions can be introduced from many sources. Examples include: thermal noise, variations in voltage supply, variation in the reference voltage, phase noise due to reproduced properly, sampling clock jitter and noise due to quantization error.

Under the correct conditions, averaging will reduce noise and improve the SNR. This will effectively increase the number of bits of a measurement's resolution. Such a system can be implemented with an ADC and a software routine that takes a set of samples and averages (filters) them for the result.

Averaging is a simple technique where an analog input is sampled several times and average of the results is taken. This technique is helpful in eliminating the effect of noise on the analog input or wrong conversion. The buffer is maintained by using an array to fill the ADC results in a periodic task.

The algorithm followed is as stated below:

1. Maintain a variable which contains an index for array.
2. After every new conversion, the converted value is stored in the array.
3. Average will be taken at the end of the max Configured counter value.
4. Once the index reaches the end of array, reset the index to the start of array.

The sequence diagram involving operations for the averaging technique used in ADC can be as depicted in the Fig 5

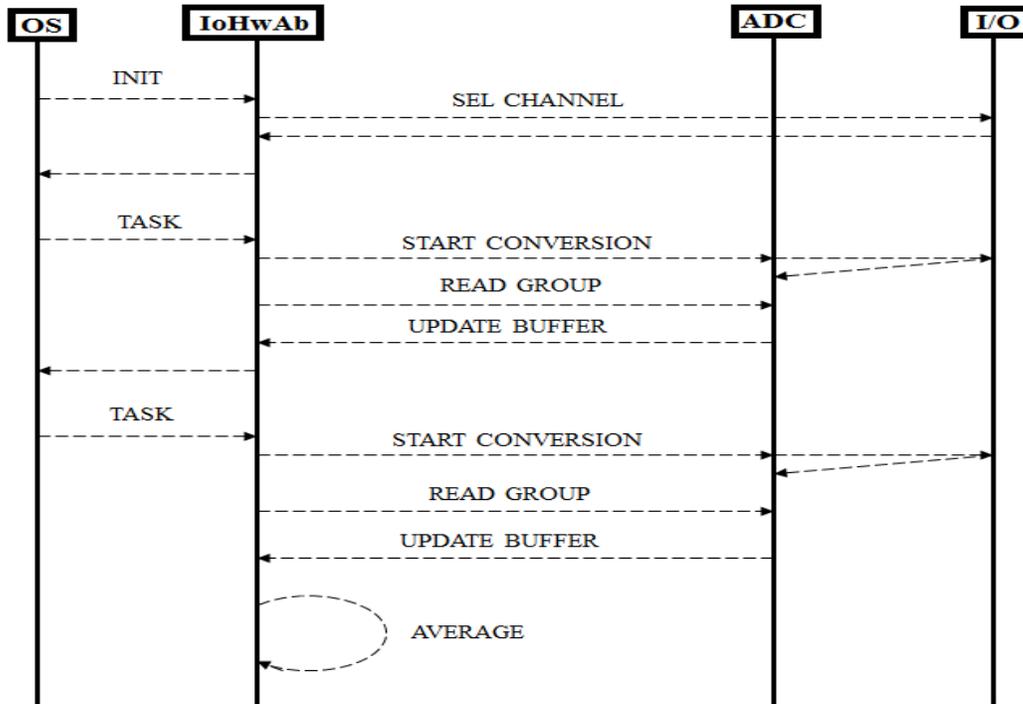


Fig 5 Sequence diagram of ADC filtering logic

## VI. RESULTS AND DISCUSSION

### 1. Sensor Diagnosis:

The Table 1 shows the result of sensor diagnosis based on the input feedback current with filter logic.

Table 1 Result of Sensor Diagnosis

Feedback Current	Application Buffer
> 0.3A	STB (0x03)
< 0.9mA	STG (0x02)
0.3A to 0.9mA	Normal Operation

For the input feedback current as shown in the Fig 6, the results of ADC values are logged in IoHwAb\_Record\_ADC\_Value buffer as shown in Table 2 without input filter logic and Table 3 with filter logic and a graph is plotted for the same as shown in Fig 7 and Fig 8.

It can be observed from the graphs that the analysis of input filters has helped to obtain steady ADC values to avoid the effect of noise. This also ensures the protection against the effect of sudden high frequency spike that might occur resulting in maintenance of the quality of the designed module and proper detection of faults at the sensor circuit.

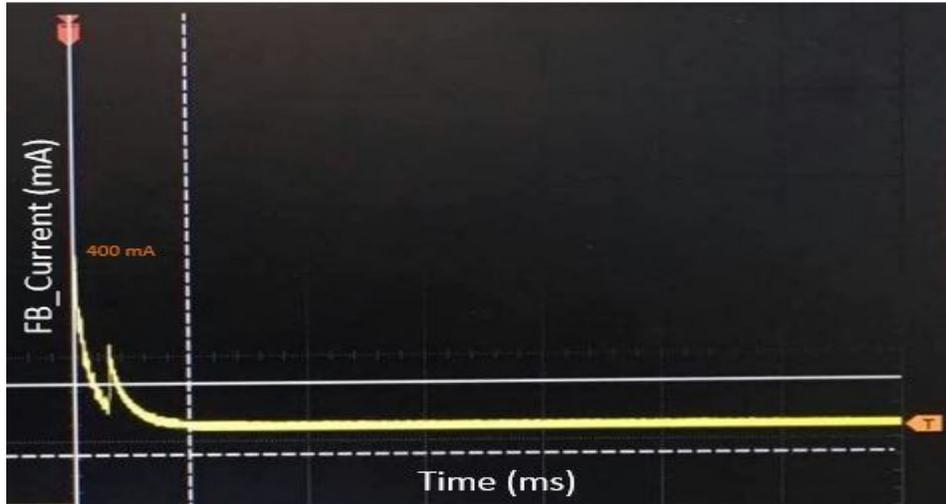


Fig 6: Feedback current of sensor

Table 2 ADC value of sensor without filter logic

IOHWAB_Record_ADC_Value	ADC_Value
[0]	4095
[1]	3803
[2]	3712
[3]	3841
[4]	3779

Table 3 ADC value of sensor with filter logic

IOHWAB_Record_ADC_Value	ADC_Value
[0]	3836
[1]	3789
[2]	3782
[3]	3783
[4]	3782

**2. Sensor diagnosis Without filter logic :**

When diagnosis is done based on sensor feedback current without the filter logic, at Fb\_current 400mA (as shown in Fig 6) > 0.3A due to noise, the application reads the value as short to battery from ADC driver and it will not consider the sensor circuit for further analysis and failure code set. Which results into malfunction of the software module.

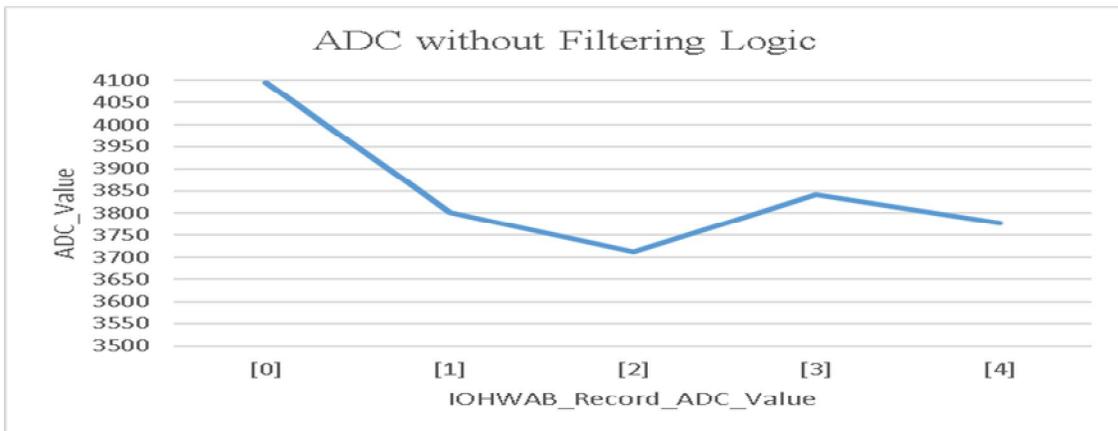


Fig 7 ADC value without Filter logic

### 3. Sensor diagnosis With filter logic :

When diagnosis is done based on sensor feedback current with the filter logic, at Fb\_current 400mA (as shown in Fig 6) > 0.3A due to noise, the ADC driver reads value continuously and takes an average which is 250mA and application reads as normal operation and it will consider the sensor circuit for further analysis and prevents malfunction of the software module.

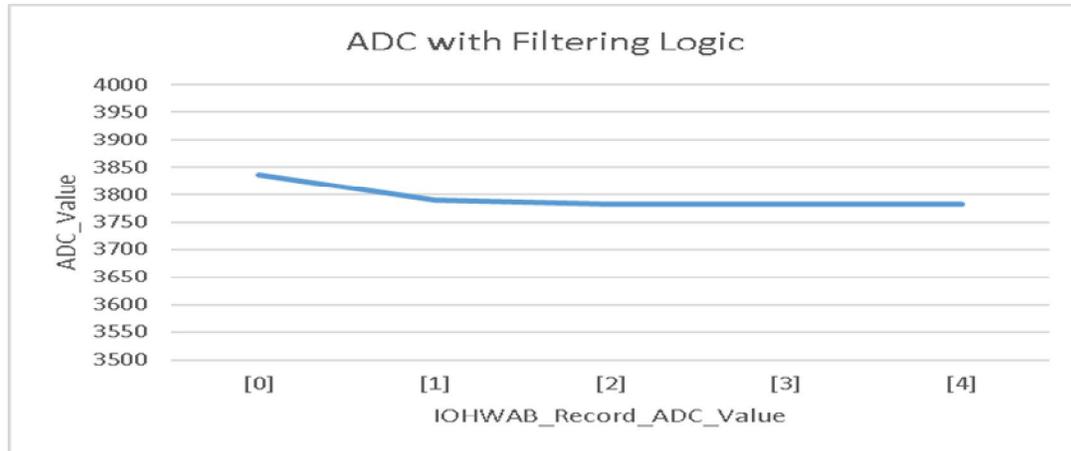


Fig 8 ADC value with Filter logic

## VII. CONCLUSIONS

The analysis and implementation of Sensor diagnosis is successful. The effect of sudden high frequency spike which might result in wrong diagnosis of sensor is eliminated. Different filter logic can also be implemented against the averaging technique used in this paper for future to enhance the integrity of the sensor circuit.

## VIII. REFERENCES

- [1] AUTOSAR Layered Software Architecture, 2014, AUTOSAR Specification Release 4.2.1, Retrieved on 12/10/2014.
- [2] AUTOSAR I/O hardware Abstraction, 2014, AUTOSAR Specification Release 4.2.1, retrieved on 20/11/2014.
- [3] AUTOSAR ADC Driver, 2014, AUTOSAR Specification Release 4.2.1, retrieved on 20/11/2014.
- [4] Gwangmin Park, Daehyun Kum, Sungho Jin<sup>3</sup> and Wooyoung Jung, "Implementation of AUTOSAR I/O Driver Modules for a SSPS System" in International Conference on Control, Automation and Systems 2008 Oct. 14-17, 2008 in COEX, Seoul, Korea
- [5] Woong-Jae Won, Jangkyung Son, Gwangmin Park, Daehyun Kum and Seonghun Lee, "Design and Implementation Procedure of the AUTOSAR I/O Driver Cluster" in ICROS-SICE International Joint Conference 2009 August 18-21, 2009, Fukuoka International Congress Center, Japan.
- [6] Nethravathi SH, "Implementation of AUTOSAR ASW and I/O Abstraction Layer" in IEEE Sponsored 2nd International Conference on Innovations in Information Embedded and Communication Systems ICIIECS' 15