

<b>Document Title</b>	Design Specification of Digital Input SCM_HW_Abstraction_Component
<b>Document Owner</b>	CSAR IOHWABs team, Cummins Electronics, Cummins Inc.
<b>Document Responsibility</b>	CSAR IOHWABs, Cummins Electronics, Cummins Inc.
<b>Document Identification No</b>	XXXXXXXXXX (integrity id)

<b>Document Status</b>	Draft
<b>Compatible with AUTOSAR Standard</b>	Classic Platform
<b>Compatible with Standard Release</b>	4.3.1

## Disclaimer

This work (specification and/or software implementation) and the material contained in it, as released by Cummins Inc, is for the purpose of internal use only.

This work shall not be utilized or reproduced without any modification, in any form or by any means, for informational purposes only. For any other purpose, no part of the work may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

The work has been developed for Cummins applications only.

The word CUMMINS and the AUTOSAR logo are registered trademarks.

## Table of Contents

### Contents

1	Introduction .....	4
2	Acronyms and Abbreviations .....	5
3	Bibliography .....	6
3.1	Input documents, related standards and norms .....	6
4	Interfaces to other components .....	6
5	Requirements Tracing .....	7
6	SWC File structure .....	8
7	Functional specification .....	9
7.1	General Functionality .....	9
7.2	Error classification .....	9
7.2.1	Development Errors .....	9
7.2.2	Production Errors .....	9
8	API specification .....	10
8.1	Function List .....	10
8.2	Function : Dio_ReadChannel .....	10
8.2.1	Function attributes: .....	10
8.2.2	Function Call Tree: .....	11
8.3	Digital Input Flow Chart .....	12
9	Lessons Learnt .....	13
10	Revision History .....	13

## 1 Introduction

The I/O Hardware Abstraction is to provide access to MCAL drivers by mapping I/O Hardware Abstraction ports to ECU signals. The data provided to the software component is completely abstracted from the physical layer values. Therefore, the software component designer does not need detailed knowledge about the MCAL driver's API and the units of the physical layer values anymore.

The I/O Hardware Abstraction is always an ECU specific implementation because the requirements of the software components to the basic software must be fitted to the features of a certain MCAL implementation. The I/O Hardware Abstraction shall provide the service for initializing the whole I/O Hardware Abstraction.

IOHWAB component named SCM\_HW\_Abstraction\_Component is created at IOHWAB layer, and the 14 Digital Inputs are mapped into it. SCM\_HW\_Abstraction\_Component is responsible for providing the requested services related to Digital Inputs as requirements specified.

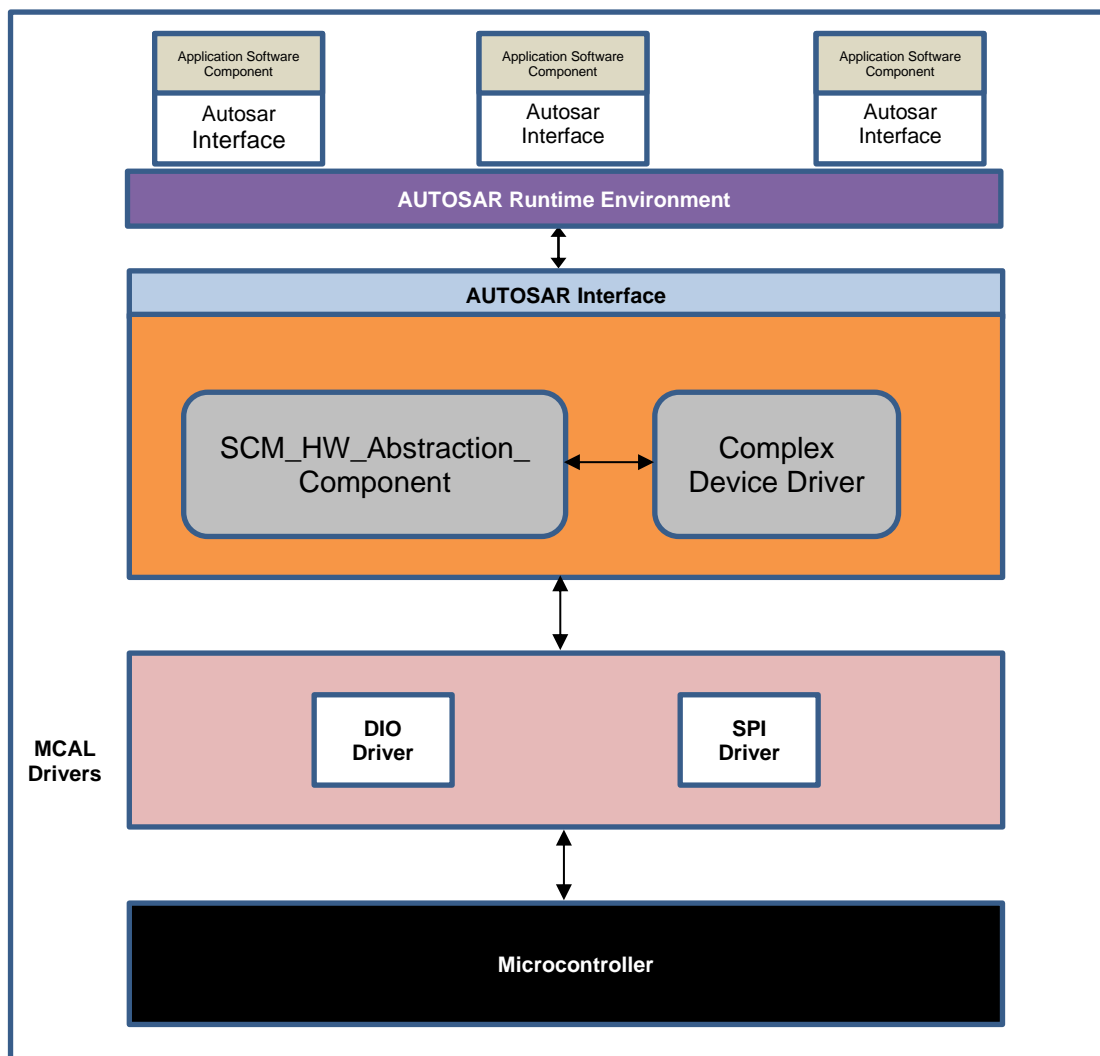


Figure 1: Autosar Layered Architecture

## 2 Acronyms and Abbreviations

The glossary below includes acronyms and abbreviations relevant to SCM\_HW\_Abstraction\_Component that are used in this document.

Abbreviation / Acronym:	Description:
AUTOSAR	Automotive Open System Architecture
API	Application Programming Interface
BSW	Basic Software
CDD	Complex Device Driver
DIO	Digital Input Output
DET	Default Error Tracer
ECU	Electronic Control Unit
HW	Hardware
IoHwAb	Input/Output Hardware Abstraction
MCAL	Microcontroller Abstraction Layer
RTE	Run Time Environment
S/R	Sender/Receiver
SDD	Software Design Document
SPI	Serial Peripheral Interface
SW	Software
SWC	Software Component
SWS	Software Specification

### 3 Bibliography

#### 3.1 Input documents & related standards and norms

[1] Specification of I/O Hardware Abstraction  
AUTOSAR\_SWS\_IOHardwareAbstraction.pdf

[2] Requirements on I/O Hardware Abstraction  
AUTOSAR\_SRS\_IOHWAbstraction.pdf

[3] Specification of DIO Driver  
AUTOSAR\_SWS\_DIODriver.pdf

[4] Requirements on DIO Driver  
AUTOSAR\_SRS\_DIODriver.pdf

[5] Vector Technical Reference  
TechnicalReference\_IoHwAb

### 4 Interfaces to other components

Digital Input IOHWAb component will communicate with CDD for virtual digital inputs.

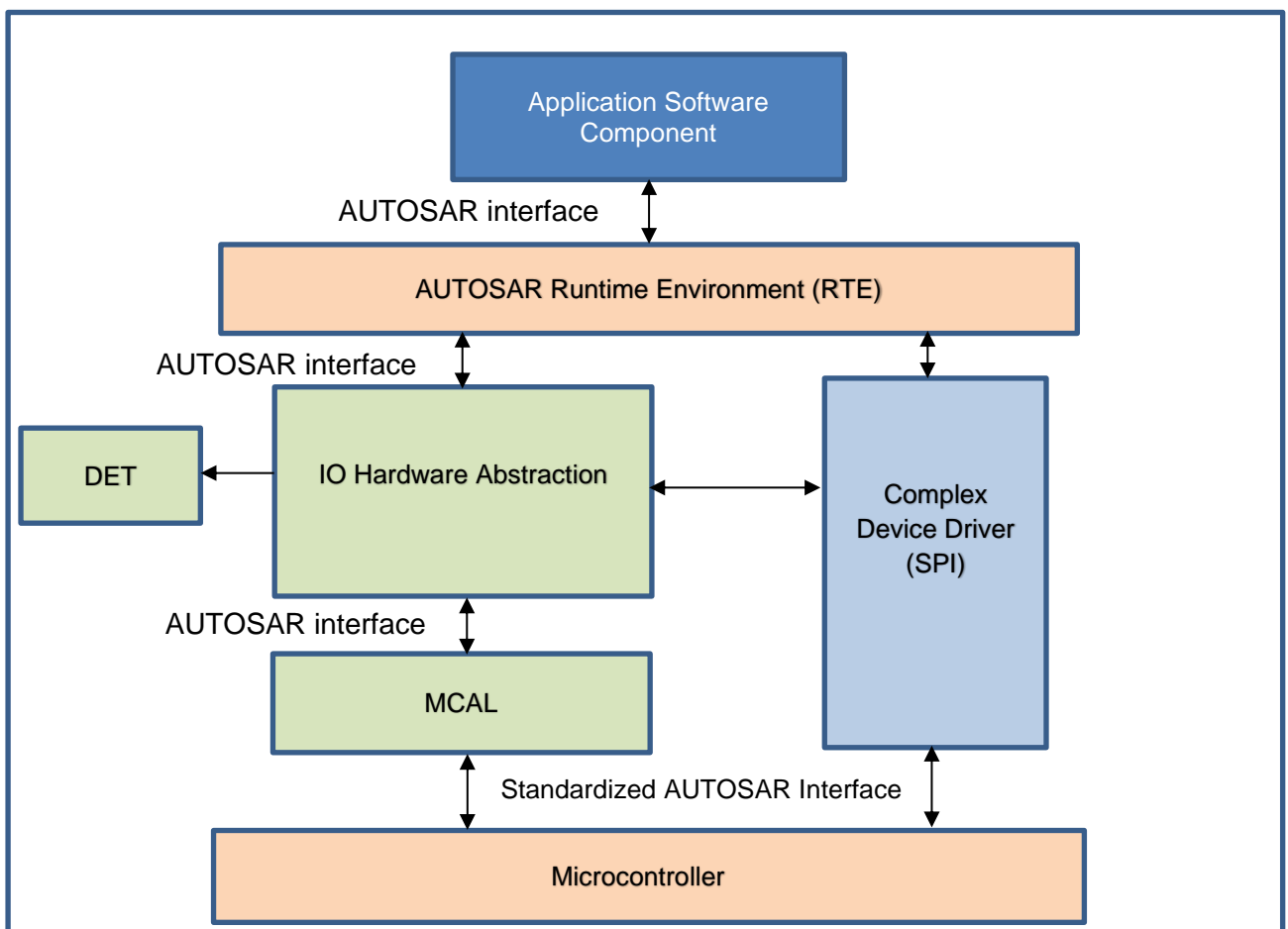


Figure 2: Inter Module Interfaces

## 5 Requirements Tracing

Requirement	Description	Satisfied by
6430713	Digital input hardware abstraction shall create a single component for digital inputs	
6430721	Digital Input Hardware Abstraction shall define if the digital input state needs to be inverted at pre-Compile	
6430720	Digital Input Hardware Abstraction component shall have an init runnable to initialize all variables to be used	
6430719	Digital Input Hardware Abstraction shall have a sender port for providing the values for the digital input channels: *. Digital Input value (ON or OFF) *. Digital Input status with the following values *. Available (supported in SCM base software) *. Not Available (not supported in SCM base software) *. OORH (not supported in SCM base software) *. OORL (not supported in SCM base software) *. Network Failure (not supported in SCM base software) *. Hardware Failure (supported in SCM base software)	
6430717	Digital Input ports data types shall be of explicit read and write	
6430712	Digital Input Hardware Abstraction component shall group periodic runnables based on update rates	
6430709	The digital input read runnable shall perform the following steps in order: *. Read the digital input state from driver layer for digital inputs *. Apply inversion if needed *. Provide data via the sender port	

6430718	Digital input periodic runnable shall execute before the components that uses its values	
6430710	Digital input hardware abstraction component shall report development errors to DET when invalid data is received from driver layer	

## 6 SWC File structure

1. IOHWAbs header file:  
IOHWAbs.h
2. IOHWAbs configuration file:  
IOHWAbs \_Cfg.h
3. IOHWAbs Callout Stubs file:  
IOHWAbs \_Callout\_Stubs.c
4. Dio Header file:  
Dio.h
5. Development Error Tracer header file:  
Det.h



## 7 Functional specification

### 7.1 General Functionality

The DIO Driver abstracts the access to the microcontroller's hardware pins. Furthermore, it allows the grouping of those pins.

The DIO Driver provides services for reading and writing to/from:

- DIO Channels (Pins)
- DIO Ports

The DIO driver can be used to modify the levels for e.g., we can read and write the signal levels for microcontroller pin. The MCAL layer APIs should be used for configuration of this DIO drivers. If any error occurs, then it should be reported to Development Error Tracer (DET) module.

### 7.2 Error classification

#### 7.2.1 Development Errors

. SCM\_HW\_Abstraction\_Component for digital inputs is responsible for reporting the Development Error to the DET on the reception of INVALID DATA from the MCAL.

#### 7.2.2 Production Errors

Not Applicable since no Requirements

## 8 API specification

### 8.1 Function List

This is a list of API functions defined in this SWC SCM\_HW\_Abstraction\_Component

No.	Function Name	Function	Integrity Req id	Defined in source file
1	Dio_ReadChannel	Returns the value of the specified DIO channel		
2	CDD_Dio_ReadChannel	Returns the value of the specified CDD		

### 8.2 Function: Dio\_ReadChannel

#### 8.2.1 Function attributes:

Service name:	Dio_ReadChannel	
Syntax:	Dio_LevelType Dio_ReadChannel (Dio_ChannelType ChannelId)	
Service ID [hex]:	0x00	
Sync/Async:	Synchronous	
Reentrancy:	Reentrant	
Parameters (in):	ChannelId	ID of DIO channel
Parameters (inout):	None	
Parameters (out):	None	
Return value:	Dio_LevelType	STD_HIGH The physical level of the corresponding Pin is STD_HIGH STD_LOW The physical level of the corresponding Pin is
		STD_LOW
Description:	Returns the value of the specified DIO channel.	

### 8.2.2 Function Call Tree:

The diagrams below show the sequences when calling the Dio\_Read\_Channel service.

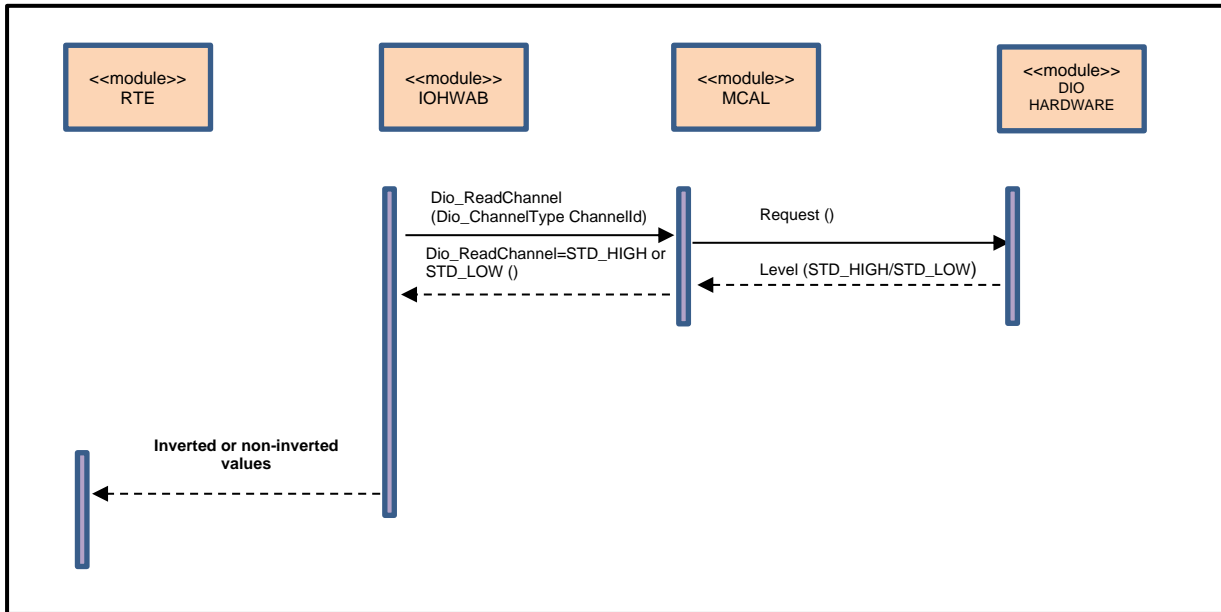


Figure 3: Read service sequence chart (Digital Input)

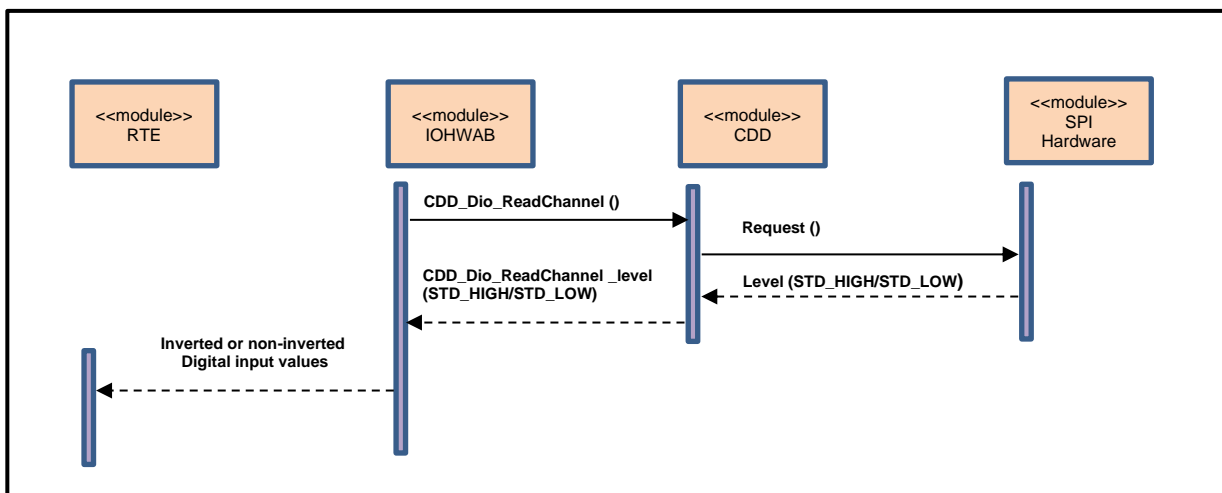
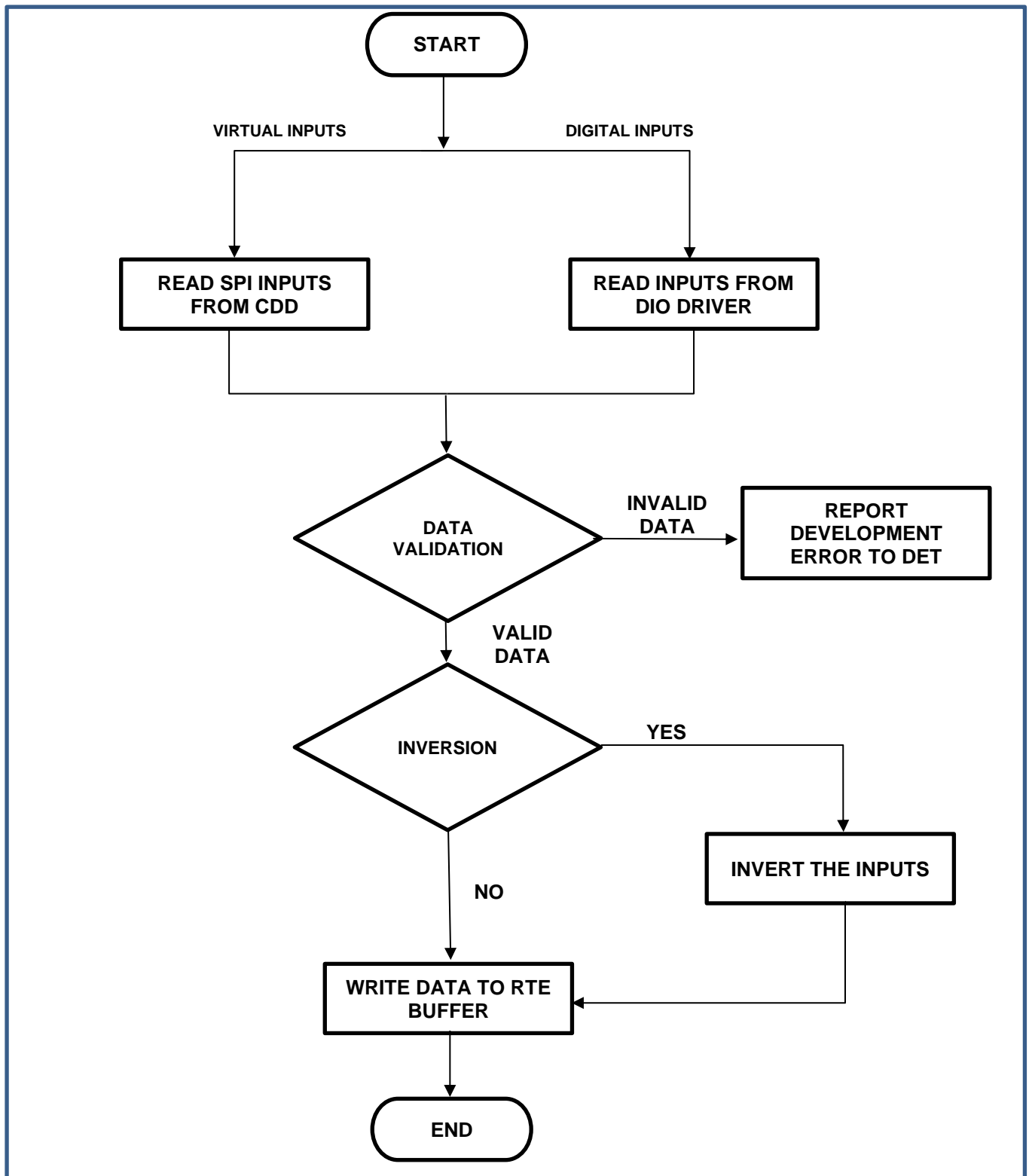


Figure 4: Read service sequence chart (Virtual Input)

### 8.3 Digital Input Flow Chart



## 9 Lessons Learnt

This section is to capture any lessons learnt during previous design cycle of this SWC. The intent is to not make the same mistakes in future.

## 10 Revision History

Revision history of this document is shown in below table

Revision Level	Date	Name	Description
1.0	10-08-2022	Samarth Kerudi	Initial creation of document