/IPB/20_Software/SCDD Library/LVDC DSP_C

SCDD_Adc

Software Component Detailed Design

Version: 0.1 (C0_RfR)
Printed by: I-Ritesh.K
Printed on: Tuesday, July 23, 2024

Generated from DOORS 9.5.1.2

1 Software Component Design Description 1

1.1 Introduction 1

2 Attributes 2

3 Views 3

4 Adc 4

4.1 External Interfaces 4

4.2 Internal design 4

4.3 Requirements 4

4.3.1 ASIL 4

4.3.2 Hardware Interface 4

4.3.3 Adc Function 5

4.3.3.1 ASIL Function 5

4.3.3.1.1 GET_ADCRES(n) 5

4.3.3.2 QM Function 6

4.3.3.2.1 void Adc_Init(void) 6

4.3.3.2.2 void Adc_cfg(void) 6

4.3.3.2.3 void Adc_cfg_ch_HwVer1(void) 6

4.3.3.2.4 void Adc_cfg_ch_HwVer0(void) 7

4.3.3.2.5 void Adc_Pt_Init(void) 7

4.3.3.2.6 void Adc_Pt_Init_HwVer1(void) 7

4.3.3.2.7 void Adc_Pt_Init_HwVer0(void) 7

4.3.3.2.8 void Adc_StartUp(void) 8

4.3.3.2.9 void HsfbMeas_GetInitOffset(HSFBMEAS_S_INITOFFSET *y) 8

ID	Software Component Detailed Design
SCDD_Adc1	1 **Software Component Design Description**
SCDD_Adc2	1.1 **Introduction**
SCDD_Adc3	This document describes the needed requirements for a SWC or BSWM.
SCDD_Adc4	This is module is the Software Component Detail Description. It contains each SW component of each SW architecture. It is always structured in: External Interface Internal Design Requirements

ID	Software Component Detailed Design
SCDD_Adc5	[2] **Attributes**
SCDD_Adc6	Agreed attributes for SWE.3 (ENG.6)

ID	Software Component Detailed Design
SCDD_Adc10	[3] **Views**
SCDD_Adc11	**SwConstructionView**: This view is used for the sw construction process.
SCDD_Adc12	**SCDD_EditView**: This view is used for creating the content of SCDD

SCDD_Adc13

4 Adc

SCDD_Adc14

4.1 External Interfaces

SCDD_Adc89

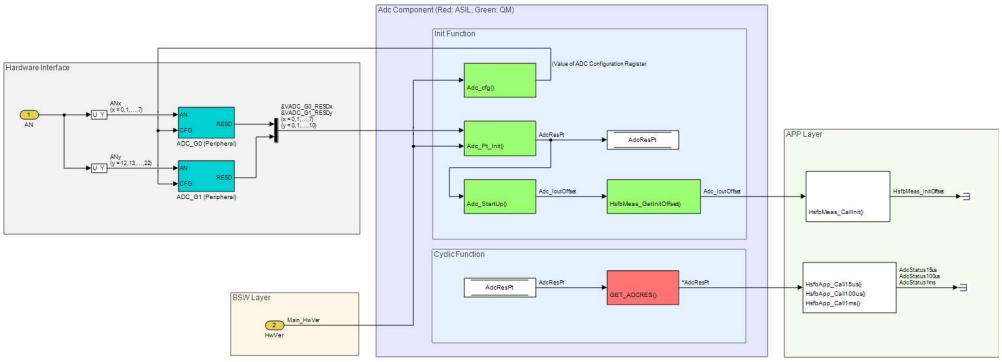
The function interface of this component are as following:

Function	Signal Name	Data Type	Direction
Adc_Init	N/A	N/A	N/A
Adc_cfg	N/A	N/A	N/A
Adc_cfg_ch_HwVer1	N/A	N/A	N/A
Adc_cfg_ch_HwVer0	N/A	N/A	N/A
Adc_Pt_Init	N/A	N/A	N/A
Adc_Pt_Init_HwVer1	N/A	N/A	N/A
Adc_Pt_Init_HwVer0	N/A	N/A	N/A
Adc_StartUp	N/A	N/A	N/A
HsfbMeas_GetInitOffset	y	HSFBMEAS_S_INITOFFSET	Output

SCDD_Adc16

4.2 Internal design

SCDD_Adc27



SCDD_Adc18

4.3 Requirements

SCDD_Adc19

4.3.1 ASIL

SCDD_Adc42

The Adc component is ASIL-B level.

SCDD_Adc20

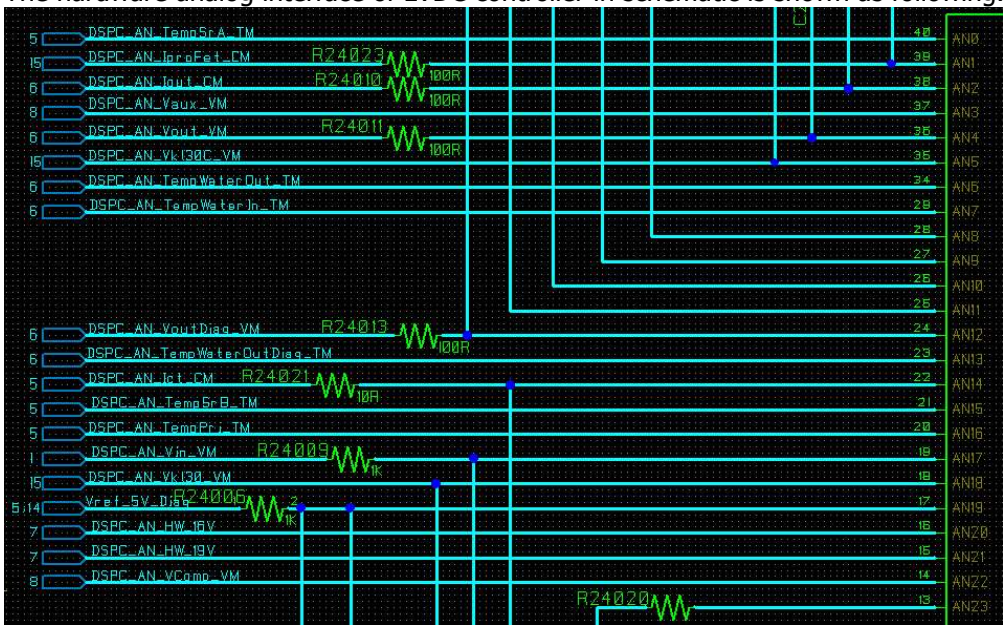
4.3.2 Hardware Interface

SCDD_Adc51

The C code of this component can be found in following link:
https://desoeap16.delta.corp/svn/IPB_PPE_auto_porsche/trunk/20_Design/23_Software/2304_Implementation/10_APPL/40_DcDcController/4010_HSFB_LVDC_B1_MBD/30_Bsw/Mcal/Adc

SCDD_Adc44

The HSI of LVDC controller can be found in following link:
https://desoeap16.delta.corp/svn/IPB_PPE_auto_porsche/trunk/20_Design/20_System_Design/2020_System_Archicture/50_HSI/HSI_all_C0.xlsx

ID	Software Component Detailed Design
SCDD_Adc45	The hardware analog interface of LVDC controller in schematic is shown as following. 
SCDD_Adc46	The hardware analog signals of LVDC controller are as following. **Signal Name** | **Description** DSPC_AN_TempSrA_TM | SrA temperature measurement DSPC_AN_IproFet_CM | Profet current measurement DSPC_AN_Iout_CM | LV current measurement DSPC_AN_Vaux_VM | 13V supply voltage measurement DSPC_AN_Vout_VM | LV voltage measurement DSPC_AN_Vkl30C_VM | KL30C voltage measurement DSPC_AN_TempWaterOut_TM | WaterOut temperature measurement DSPC_AN_TempWaterIn_TM | WaterIn temperature measurement DSPC_AN_VoutDiag_VM | LV voltage measurement diag DSPC_AN_TempWaterOutDiag_TM | WaterOut temperature measurement diag DSPC_AN_Ict_CM | CT current measurement DSPC_AN_TempSrB_TM | SrB temperature measurement DSPC_AN_TempPri_TM | Primary temperature measurement DSPC_AN_Vin_VM | HV voltage measurement DSPC_AN_Vkl30_VM | KL30 voltage measurement Vref_5V_Diag | Vref 5V measurement diag DSPC_AN_HW_16V | 16V OVP compare voltage measurement DSPC_AN_HW_19V | 19V OVP compare voltage measurement DSPC_AN_VComp_VM | HSK voltage measurement diag
SCDD_Adc21	4.3.3 **Adc Function**
SCDD_Adc76	4.3.3.1 **ASIL Function**
SCDD_Adc52	4.3.3.1.1 **GET_ADCRES(n)**

<table>
<tr><th>ID</th><th>Software Component Detailed Design</th></tr>
<tr><td>SCDD_Adc55</td><td>This function is a macro function, which will return the result of AdcResPt. The input of this function is the index of AdcResPt. The output of this function is the content of ADC result register, whose address is stored in AdcResPt[index].</td></tr>
<tr><td>SCDD_Adc73</td><td>The enumerate value of each signal are show as following:
<table>
<tr><th>Number</th><th>Signal Name</th></tr>
<tr><td>0</td><td>Ipri</td></tr>
<tr><td>1</td><td>Vin</td></tr>
<tr><td>2</td><td>Iout</td></tr>
<tr><td>3</td><td>Vout</td></tr>
<tr><td>4</td><td>VoutDiag</td></tr>
<tr><td>5</td><td>IproFet</td></tr>
<tr><td>6</td><td>Vaux</td></tr>
<tr><td>7</td><td>Vkl30</td></tr>
<tr><td>8</td><td>Vkl30C</td></tr>
<tr><td>9</td><td>Vcomp</td></tr>
<tr><td>10</td><td>VrefDiag</td></tr>
<tr><td>11</td><td>Vhw16V</td></tr>
<tr><td>12</td><td>Vhw19V</td></tr>
<tr><td>13</td><td>NtcPri</td></tr>
<tr><td>14</td><td>NtcSrA</td></tr>
<tr><td>15</td><td>NtcSrB</td></tr>
<tr><td>16</td><td>NtcWaterIn</td></tr>
<tr><td>17</td><td>NtcWaterOut</td></tr>
<tr><td>18</td><td>NtcWaterOutDiag</td></tr>
</table></td></tr>
<tr><td>SCDD_Adc79</td><td>4.3.3.2 QM Function</td></tr>
<tr><td>SCDD_Adc54</td><td>4.3.3.2.1 void Adc_Init(void)</td></tr>
<tr><td>SCDD_Adc56</td><td>This function is the initialization function of Adc component. This function will intialize the configuration of ADC peripheral and the value of AdcResPt.
This function has no input and no output.</td></tr>
<tr><td>SCDD_Adc81</td><td>This function will call following function:
Adc_cfg()
Adc_Pt_Init()</td></tr>
<tr><td>SCDD_Adc83</td><td>4.3.3.2.2 void Adc_cfg(void)</td></tr>
<tr><td>SCDD_Adc84</td><td>This function is used for configuration of ADC peripheral. The function will configure the CLC, ARBCFG, GLOBCFG, CHCTR, ICLASS, ARBPR register, and also configure the ADC channel.
This function has no input and no output.</td></tr>
<tr><td>SCDD_Adc85</td><td>This function will call following function:
Adc_cfg_ch_HwVer1()
Adc_cfg_ch_HwVer0()</td></tr>
<tr><td>SCDD_Adc57</td><td>4.3.3.2.3 void Adc_cfg_ch_HwVer1(void)</td></tr>
</table>

SCDD_Adc58: This function is used for configuration of ADC channel, based on C0 HW version. The function will configure the QINR0, QCTRL0, QMR0, ASCTRL, ASMR, ASSEL register.
This function has no input and no output.

SCDD_Adc59: 4.3.3.2.4 **void Adc_cfg_ch_HwVer0(void)**

SCDD_Adc60: This function is used for configuration of ADC channel, based on B02 HW version. The function will configure the QINR0, QCTRL0, QMR0, ASCTRL, ASMR, ASSEL register.
This function has no input and no output.

SCDD_Adc63: 4.3.3.2.5 **void Adc_Pt_Init(void)**

SCDD_Adc64: This function is used for initializing the value of AdcResPt. Due to the interface difference, the value of AdcResPt will be determined by HW version.
This function has no input and no output.

SCDD_Adc86: This function will call following function:
Adc_Pt_Init_HwVer1()
Adc_Pt_Init_HwVer0()

SCDD_Adc65: 4.3.3.2.6 **void Adc_Pt_Init_HwVer1(void)**

SCDD_Adc66: This function is used for initializing the value of AdcResPt, based on C0 HW version. The address of ADC result register will be stored in AdcResPt.
This function has no input and no output.

SCDD_Adc88: The value of AdcResPt[x] are as following:

Number	Signal Name	Channel	*AdcResPt[x]
0	Ipri	AN14	VADC_G1_RESD2.U
1	Vin	AN17	VADC_G1_RESD5.U
2	Iout	AN2	VADC_G0_RESD2.U
3	Vout	AN4	VADC_G0_RESD4.U
4	VoutDiag	AN12	VADC_G1_RESD0.U
5	IproFet	AN1	VADC_G0_RESD1.U
6	Vaux	AN3	VADC_G0_RESD3.U
7	Vkl30	AN18	VADC_G1_RESD6.U
8	Vkl30C	AN5	VADC_G0_RESD5.U
9	Vcomp	AN22	VADC_G1_RESD10.U
10	VrefDiag	AN19	VADC_G1_RESD7.U
11	Vhw16V	AN20	VADC_G1_RESD8.U
12	Vhw19V	AN21	VADC_G1_RESD9.U
13	NtcPri	AN16	VADC_G1_RESD4.U
14	NtcSrA	AN0	VADC_G0_RESD0.U
15	NtcSrB	AN15	VADC_G1_RESD3.U
16	NtcWaterIn	AN7	VADC_G0_RESD7.U
17	NtcWaterOut	AN6	VADC_G0_RESD6.U
18	NtcWaterOutDiag	AN13	VADC_G1_RESD1.U

SCDD_Adc67: 4.3.3.2.7 **void Adc_Pt_Init_HwVer0(void)**

<table>
<tr><th>ID</th><th>Software Component Detailed Design</th></tr>
<tr><td>SCDD_Adc68</td><td>This function is used for initializing the value of AdcResPt, based on B02 HW version. The address of ADC result register will be stored in AdcResPt.
This function has no input and no output.</td></tr>
<tr><td>SCDD_Adc87</td><td>The value of AdcResPt[x] are as following:
<table>
<tr><th>Number</th><th>Signal Name</th><th>Channel</th><th>*AdcResPt[x]</th></tr>
<tr><td>0</td><td>Ipri</td><td>AN0</td><td>VADC_G0_RESD0.U</td></tr>
<tr><td>1</td><td>Vin</td><td>AN1</td><td>VADC_G0_RESD1.U</td></tr>
<tr><td>2</td><td>Iout</td><td>AN2</td><td>VADC_G0_RESD2.U</td></tr>
<tr><td>3</td><td>Vout</td><td>AN4</td><td>VADC_G0_RESD4.U</td></tr>
<tr><td>4</td><td>VoutDiag</td><td>AN12</td><td>VADC_G1_RESD0.U</td></tr>
<tr><td>5</td><td>IproFet</td><td>AN17</td><td>VADC_G1_RESD5.U</td></tr>
<tr><td>6</td><td>Vaux</td><td>AN3</td><td>VADC_G0_RESD3.U</td></tr>
<tr><td>7</td><td>Vkl30</td><td>AN18</td><td>VADC_G1_RESD6.U</td></tr>
<tr><td>8</td><td>Vkl30C</td><td>AN5</td><td>VADC_G0_RESD5.U</td></tr>
<tr><td>9</td><td>Vcomp</td><td>N/A</td><td>N/A</td></tr>
<tr><td>10</td><td>VrefDiag</td><td>N/A</td><td>N/A</td></tr>
<tr><td>11</td><td>Vhw16V</td><td>N/A</td><td>N/A</td></tr>
<tr><td>12</td><td>Vhw19V</td><td>N/A</td><td>N/A</td></tr>
<tr><td>13</td><td>NtcPri</td><td>AN16</td><td>VADC_G1_RESD4.U</td></tr>
<tr><td>14</td><td>NtcSrA</td><td>AN14</td><td>VADC_G1_RESD2.U</td></tr>
<tr><td>15</td><td>NtcSrB</td><td>AN15</td><td>VADC_G1_RESD3.U</td></tr>
<tr><td>16</td><td>NtcWaterIn</td><td>AN7</td><td>VADC_G0_RESD7.U</td></tr>
<tr><td>17</td><td>NtcWaterOut</td><td>AN6</td><td>VADC_G0_RESD6.U</td></tr>
<tr><td>18</td><td>NtcWaterOutDiag</td><td>AN13</td><td>VADC_G1_RESD1.U</td></tr>
</table></td></tr>
<tr><td>SCDD_Adc69</td><td>4.3.3.2.8 void Adc_StartUp(void)</td></tr>
<tr><td>SCDD_Adc70</td><td>This function is used for calculating startup offset of Iout measurement. This function will read data for every 100us, and calculate the sum of 100 data. The average of these 100 data will be set to Adc_IoutOffset.
This function has no input and no output.</td></tr>
<tr><td>SCDD_Adc71</td><td>4.3.3.2.9 void HsfbMeas_GetInitOffset(HSFBMEAS_S_INITOFFSET *y)</td></tr>
<tr><td>SCDD_Adc72</td><td>This function is used for reading startup offset from HsfbMeas component. This function will copy the value of Adc_IoutOffset to HsfbMeas_InitOffset.Iout.
This function has no input.
The output of this function is the pointer of HSFBMEAS_S_INITOFFSET.</td></tr>
</table>