/IPB/20_Software/SCDD Library/LVDC DSP_C

SCDD_Main0

Software Component Detailed Design

Version: 0.1 (C0_RfR)
Printed by: I-Ritesh.K
Printed on: Thursday, July 18, 2024

Generated from DOORS 9.5.1.2

ID	Software Component Detailed Design
SCDD_Main1	1 **Software Component Design Description**
SCDD_Main2	1.1 **Introduction**
SCDD_Main3	This document describes the needed requirements for a SWC or BSWM.
SCDD_Main4	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_ Main5	## 2 Attributes
SCDD_ Main6	Agreed attributes for SWE.3 (ENG.6)

ID	Software Component Detailed Design
SCDD_ Main10	3 **Views**
SCDD_ Main11	**SwConstructionView**: This view is used for the sw construction process.
SCDD_ Main12	**SCDD_EditView**: This view is used for creating the content of SCDD

ID Software Component Detailed Design

SCDD_ Main13

4 Main

SCDD_ Main14

4.1 External Interfaces

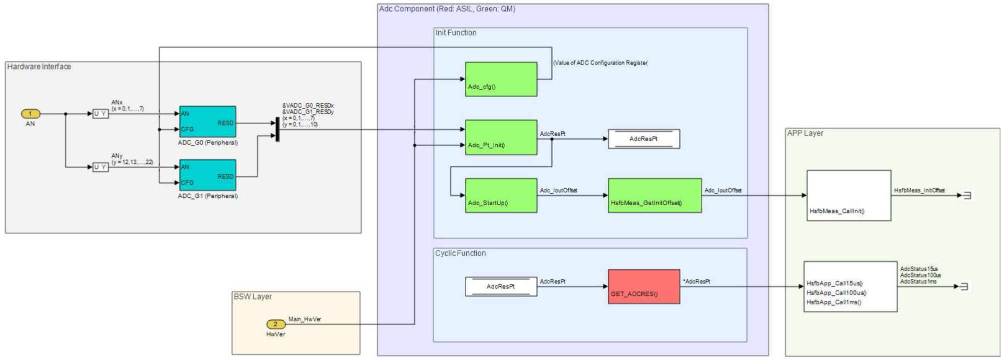
SCDD_ Main89

The function interface of this component are as following:

Function Name	Description	Data Type	Direction
core0_main	Main function for core 0	N/A	N/A
McalInit	Initialization of MCAL function	N/A	N/A
BswInit	Initialization of BSW function	N/A	N/A
delay_ms_main	Delay function for ms delay	N/A	N/A
Main_GetHwVerApp	Get HW version from Main to App component	N/A	N/A
Main_GetHwVer	Get HW version from Main	N/A	N/A
Main_UdsGetHwVer	Get HW version for Uds component	N/A	N/A
Trap_vInit		N/A	N/A
mg_vTrapTableRedirection		N/A	N/A
mg_vTrapTableRestore		N/A	N/A

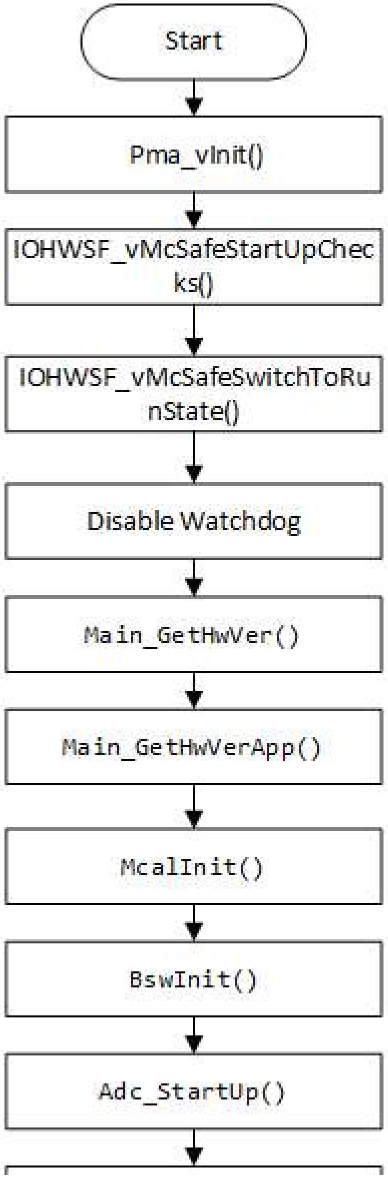
SCDD_ Main16

4.2 Internal design

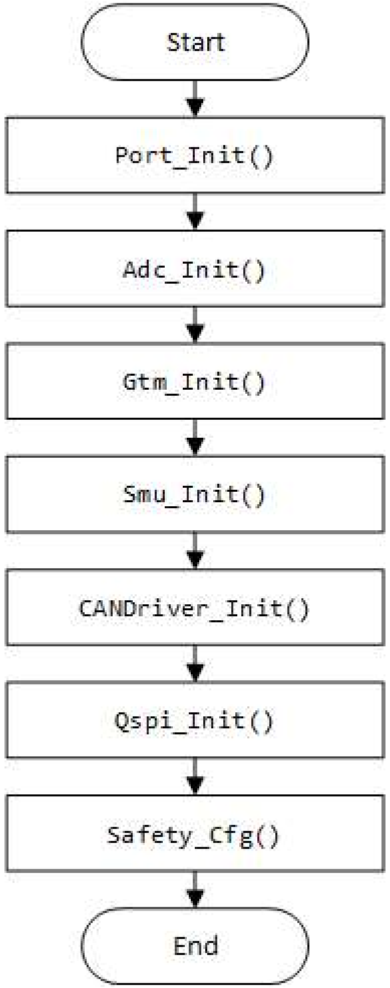
ID	Software Component Detailed Design
SCDD_ Main27	 Adc Component (Red: ASIL, Green: QM) Init Function Hardware Interface 1 AN U Y ANx (x = 0,1,...,7) AN CFG RESD ADC_G0 (Peripheral) U Y ANy (y = 12,13,...,22) AN CFG RESD ADC_G1 (Peripheral) &VADC_G0_RESDx &VADC_G1_RESDy (x = 0,1,...,7) (y = 0,1,...,10) Adc_cfg() (Value of ADC Configuration Register Adc_Pt_Init() AdcResPt AdcResPt Adc_StartUp() Adc_IoutOffset HsfbMeas_GetInitOffset() Adc_IoutOffset Cyclic Function AdcResPt AdcResPt GET_ADCRES() *AdcResPt BSW Layer 2 HwVer Main_HwVer APP Layer HsfbMeas_CallInit() HsfbMeas_InitOffset HsfbApp_Call15us() HsfbApp_Call100us() HsfbApp_Call1ms() AdcStatus15us AdcStatus100us AdcStatus1ms
SCDD_ Main18	4.3 **Requirements**
SCDD_ Main19	4.3.1 **QM**
SCDD_ Main42	The Main component is QM level.
SCDD_ Main20	4.3.2 **Data Define**
SCDD_ Main51	The C code of this component can be found in following link: https://desoeap16.delta.corp/svn/IPB_PPE_auto_porsche/branches/20_RelBranch/C01R0P0/LvDc/30_Bsw/Main
SCDD_ Main90	The HW variant code can be found in following link: https://desoeap16.delta.corp/svn/IPB_PPE_auto_porsche/trunk/20_Design/21_EE_Hardware_Design/2101_Concept/210134_Variant_Coding/HW_Variant_Code_Definition.xlsx
SCDD_ Main21	4.3.3 **Main Function**
SCDD_ Main76	4.3.3.1 **QM Function**

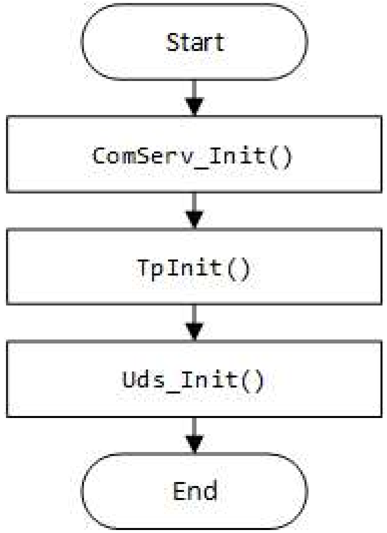
ID	Software Component Detailed Design
SCDD_ Main52	4.3.3.1.1 **core_main()**
SCDD_ Main55	The core0_main function is the main function for core 0, which will be executed after reset. The function sequence in core0_main is shown as following:
SCDD_ Main92	

ID	Software Component Detailed Design

SCDD_
Main92
	 Start
	Pma_vInit()
	IOHWSF_vMcSafeStartUpChecks()
	IOHWSF_vMcSafeSwitchToRunState()
	Disable Watchdog
	Main_GetHwVer()
	Main_GetHwVerApp()
	McalInit()
	BswInit()
	Adc_StartUp()

ID	Software Component Detailed Design
SCDD_ Main94	4.3.3.1.2 **McalInit()**
SCDD_ Main95	The McalInit function include all the initialization function in MCAL level, which will be executed after reset. The function sequence in McalInit is shown as following:

SCDD_ Main96	 Start
	Port_Init()
	Adc_Init()
	Gtm_Init()
	Smu_Init()
	CANDriver_Init()
	Qspi_Init()
	Safety_Cfg()
	End

ID	Software Component Detailed Design
SCDD_ Main97	4.3.3.1.3 **BswInit()**
SCDD_ Main98	The BswInit function include all the initialization function in BSW level, which will be executed after reset. The function sequence in BswInit is shown as following:
SCDD_ Main99	
SCDD_ Main100	4.3.3.1.4 **delay_us_main()**
SCDD_ Main102	The delay_us_main() function used to calculate time in us level,which use STM0_TIM1.
SCDD_ Main103	4.3.3.1.5 **delay_ms_main()**
SCDD_ Main104	The delay_ms_main() function used to calculate time in ms level,which use STM0_TIM4.
SCDD_ Main105	4.3.3.1.6 **Main_GetHwVerApp(void)**

ID	Software Component Detailed Design
SCDD_ Main107	The Main_GetHwVerApp function will read HW version for different requirement in same module in App level.
SCDD_ Main108	4.3.3.1.7 **Trap_vInit(void)**
SCDD_ Main109	4.3.3.1.8 **mg_vTrapTableRedirection(void)**
SCDD_ Main110	4.3.3.1.9 **mg_vTrapTableRestore(void)**
SCDD_ Main79	4.3.3.2 **QM Function**
SCDD_ Main54	4.3.3.2.1 **void Adc_Init(void)**
SCDD_ Main56	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.

SCDD_ Main81	This function will call following function:
	Adc_cfg()
	Adc_Pt_Init()

SCDD_ Main83	4.3.3.2.2 **void Adc_cfg(void)**
SCDD_ Main84	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.

SCDD_ Main85	This function will call following function:
	Adc_cfg_ch_HwVer1()
	Adc_cfg_ch_HwVer0()

<table>
<tr><th>ID</th><th>Software Component Detailed Design</th></tr>
<tr><td>SCDD_ Main57</td><td>4.3.3.2.3 <b>void Adc_cfg_ch_HwVer1(void)</b></td></tr>
<tr><td>SCDD_ Main58</td><td>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.</td></tr>
<tr><td>SCDD_ Main59</td><td>4.3.3.2.4 <b>void Adc_cfg_ch_HwVer0(void)</b></td></tr>
<tr><td>SCDD_ Main60</td><td>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.</td></tr>
<tr><td>SCDD_ Main63</td><td>4.3.3.2.5 <b>void Adc_Pt_Init(void)</b></td></tr>
<tr><td>SCDD_ Main64</td><td>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.</td></tr>
<tr><td>SCDD_ Main86</td><td>This function will call following function:
Adc_Pt_Init_HwVer1()
Adc_Pt_Init_HwVer0()</td></tr>
<tr><td>SCDD_ Main65</td><td>4.3.3.2.6 <b>void Adc_Pt_Init_HwVer1(void)</b></td></tr>
<tr><td>SCDD_ Main66</td><td>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.</td></tr>
<tr><td>SCDD_ Main88</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>AN14</td><td>VADC_G1_RESD2.U</td></tr>
<tr><td>1</td><td>Vin</td><td>AN17</td><td>VADC_G1_RESD5.U</td></tr>
<tr><td>2</td><td>Iout</td><td>AN2</td><td>VADC_G0_RESD2.U</td></tr>
</table></td></tr>
</table>

SCDD_Main88

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_Main67

4.3.3.2.7 void Adc_Pt_Init_HwVer0(void)

SCDD_Main68

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.

SCDD_Main87

The value of AdcResPt[x] are as following:

Number	**Signal Name**	**Channel**	***AdcResPt[x]**
0	Ipri	AN0	VADC_G0_RESD0.U
1	Vin	AN1	VADC_G0_RESD1.U
2	Iout	AN2	VADC_G0_RESD2.U
3	Vout	AN4	VADC_G0_RESD4.U
4	VoutDiag	AN12	VADC_G1_RESD0.U
5	IproFet	AN17	VADC_G1_RESD5.U
6	Vaux	AN3	VADC_G0_RESD3.U
7	Vkl30	AN18	VADC_G1_RESD6.U

<table>
<tr><th>ID</th><th>Software Component Detailed Design</th></tr>
<tr><td>SCDD_
Main87</td><td><table>
<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_
Main69</td><td>4.3.3.2.8 <b>void Adc_StartUp(void)</b></td></tr>
<tr><td>SCDD_
Main70</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_
Main71</td><td>4.3.3.2.9 <b>void HsfbMeas_GetInitOffset(HSFBMEAS_S_INITOFFSET *y)</b></td></tr>
<tr><td>SCDD_
Main72</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>