User Manual

for S32K14X MCU Driver

Document Number: UM2MCUASR4.3 Rev0001R1.0.1
Rev. 1.0

Contents

Section number **Title** **Page**

Chapter 1
Revision History

Chapter 2
Introduction

2.1 Supported Derivatives....19

2.2 Overview....19

2.3 About this Manual....20

2.4 Acronyms and Definitions....20

2.5 Reference List....21

Chapter 3
Driver

3.1 Requirements....23

3.2 Driver Design Summary....23

3.3 Hardware Resources....24

3.4 Deviation from Requirements....24

3.5 Driver Limitation....25

3.6 Driver usage and configuration tips....26

3.7 Runtime Errors....26

3.8 Software specification....26

3.8.1 Define Reference....27

3.8.1.1 Define MCU_INSTANCE_ID....27

3.8.1.2 Define MCU_MODULE_ID....27

3.8.1.3 Define MCU_INIT_ID....27

3.8.1.4 Define MCU_INITRAMSECTION_ID....27

3.8.1.5 Define MCU_INITCLOCK_ID....28

3.8.1.6 Define MCU_DISTRIBUTEPLLCLOCK_ID....28

3.8.1.7 Define MCU_GETPLLSTATUS_ID....28

3.8.1.8 Define MCU_GETRESETREASON_ID....28

3.8.1.9 Define MCU_GETRESETRAWVALUE_ID....29
3.8.1.10 Define MCU_PERFORMRESET_ID....29
3.8.1.11 Define MCU_SETMODE_ID....29
3.8.1.12 Define MCU_GETVERSIONINFO_ID....29
3.8.1.13 Define MCU_GETRAMSTATE_ID....30
3.8.1.14 Define MCU_GETPOWERDOMAIN_ID....30
3.8.1.15 Define MCU_GETPERIPHERALSTATE_ID....30
3.8.1.16 Define MCU_GETSYSTEMSTATE_ID....30
3.8.1.17 Define MCU_GETPOWERMODESTATE_ID....30
3.8.1.18 Define MCU_GETMEMCONFIG_ID....31
3.8.1.19 Define MCU_GETMIDRSTRUCTURE_ID....31
3.8.1.20 Define MCU_SRAMRETENCONFIG_ID....31
3.8.1.21 Define MCU_SRAM_RETEN_CONFIG_API....31
3.8.1.22 Define MCU_E_ISR_RESET_ALT_FAILURE....32
3.8.1.23 Define MCU_E_PARAM_CONFIG....32
3.8.1.24 Define MCU_E_PARAM_CLOCK....32
3.8.1.25 Define MCU_E_PARAM_MODE....33
3.8.1.26 Define MCU_E_PARAM_RAMSECTION....33
3.8.1.27 Define MCU_E_PLL_NOT_LOCKED....34
3.8.1.28 Define MCU_E_UNINIT....34
3.8.1.29 Define MCU_E_PARAM_POINTER....34
3.8.1.30 Define MCU_E_ALLREADY_INITIALIZED....35
3.8.1.31 Define MCU_E_ISR_CLOCK_FAILURE....35
3.8.1.32 Define MCU_E_ISR_PLL_LOCK_FAILURE....35
3.8.1.33 Define MCU_E_ISR_VOLTAGE_ERROR....36
3.8.1.34 Define MCU_E_ISR_TEMPERATURE_MONITOR_ERROR....36
3.8.1.35 Define MCU_DEV_ERROR_DETECT....36
3.8.1.36 Define MCU_VERSION_INFO_API....36
3.8.1.37 Define MCU_GET_RAM_STATE_API....37

3.8.1.38 Define MCU_INIT_CLOCK ... 37
3.8.1.39 Define MCU_NO_PLL ... 37
3.8.1.40 Define MCU_ENTER_LOW_POWER_MODE ... 38
3.8.1.41 Define MCU_PERFORM_RESET_API ... 38
3.8.1.42 Define MCU_TIMEOUT_LOOPS ... 39
3.8.1.43 Define MCU_RESET_CALLOUT_USED ... 39
3.8.1.44 Define MCU_ERROR_ISR_NOTIFICATION ... 39
3.8.1.45 Define MCU_VOLTAGE_ERROR_ISR_USED ... 39
3.8.1.46 Define MCU_TEMPERATURE_ERROR_ISR_USED ... 40
3.8.1.47 Define MCU_GET_PERIPH_STATE_API ... 40
3.8.1.48 Define MCU_GET_SYSTEM_STATE_API ... 40
3.8.1.49 Define MCU_POWERMODE_STATE_API ... 40
3.8.1.50 Define MCU_DISABLE_DEM_REPORT_ERROR_STATUS ... 41
3.8.1.51 Define MCU_FAST_MODE_CONFIG ... 41
3.8.1.52 Define MCU_MAX_CLKCONFIGS ... 41
3.8.1.53 Define MCU_MAX_MODECONFIGS ... 41
3.8.1.54 Define MCU_MAX_RAMCONFIGS ... 42
3.8.1.55 Define MCU_PRECOMPILE_SUPPORT ... 42
3.8.1.56 Define MCU_CONF_PB ... 42
3.8.1.57 Define MCU_NUMBER_OF_PCS_REGS ... 43
3.8.1.58 Define MCU_RAW_RESET_DEFAULT ... 43
3.8.1.59 Define MCU_VALIDATE_GLOBAL_CALL ... 43
3.8.2 Enum Reference ... 43
3.8.2.1 Enumeration Mcu_StatusType ... 44
3.8.2.2 Enumeration Mcu_PllStatusType ... 44
3.8.2.3 Enumeration Mcu_RamStateType ... 45
3.8.2.4 Enumeration Mcu_ResetType ... 45
3.8.2.5 Enumeration Mcu_SRAMRetenConfigType ... 46
3.8.3 Function Reference ... 47

3.8.3.1 Function Mcu_Init....47
3.8.3.2 Function Mcu_InitClock....48
3.8.3.3 Function Mcu_DistributePllClock....48
3.8.3.4 Function Mcu_InitRamSection....49
3.8.3.5 Function Mcu_SetMode....49
3.8.3.6 Function Mcu_GetPllStatus....50
3.8.3.7 Function Mcu_PerformReset....51
3.8.3.8 Function Mcu_GetPowerModeState....51
3.8.3.9 Function Mcu_GetRamState....51
3.8.3.10 Function Mcu_GetResetRawValue....52
3.8.3.11 Function Mcu_GetResetReason....52
3.8.3.12 Function Mcu_GetSystemState....53
3.8.3.13 Function Mcu_GetVersionInfo....53
3.8.3.14 Function Mcu_SRAMRetentionConfig....54
3.8.3.15 Function Mcu_DisableCmu....55
3.8.4 Structs Reference....55
3.8.4.1 Structure Mcu_ConfigType....55
3.8.4.2 Structure Mcu_HwIPsConfigType....56
3.8.4.3 Structure Mcu_RamConfigType....57
3.8.5 Types Reference....58
3.8.5.1 Typedef Mcu_ClockType....58
3.8.5.2 Typedef Mcu_ModeType....58
3.8.5.3 Typedef Mcu_RamSectionType....59
3.8.5.4 Typedef Mcu_RawResetType....59
3.9 Symbolic Names Disclaimer....59

Chapter 4
Tresos Configuration Plug-in

4.1 Configuration elements of Mcu....61
4.2 Form IMPLEMENTATION_CONFIG_VARIANT....61

4.3 Form McuGeneralConfiguration 62
4.3.1 McuDevErrorDetect (McuGeneralConfiguration) 62
4.3.2 McuVersionInfoApi (McuGeneralConfiguration) 63
4.3.3 McuGetRamStateApi (McuGeneralConfiguration) 63
4.3.4 McuInitClock (McuGeneralConfiguration) 63
4.3.5 McuNoPll (McuGeneralConfiguration) 64
4.3.6 McuEnterLowPowerMode (McuGeneralConfiguration) 64
4.3.7 McuTimeout (McuGeneralConfiguration) 65
4.3.8 McuEnableUserModeSupport (McuGeneralConfiguration) 65
4.3.9 McuPerformResetApi (McuGeneralConfiguration) 66
4.3.10 McuCalloutBeforePerformReset (McuGeneralConfiguration) 66
4.3.11 McuPerformResetCallout (McuGeneralConfiguration) 67
4.3.12 McuCmuNotification (McuGeneralConfiguration) 67
4.3.13 McuErrorIsrNotification (McuGeneralConfiguration) 67
4.3.14 McuDisableSimInit (McuGeneralConfiguration) 68
4.3.15 McuDisableRcmInit (McuGeneralConfiguration) 68
4.3.16 McuDisablePmcInit (McuGeneralConfiguration) 69
4.3.17 McuDisableSmcInit (McuGeneralConfiguration) 69
4.4 Form McuDebugConfiguration 70
4.4.1 McuDisableDemReportErrorStatus (McuDebugConfiguration) 70
4.4.2 McuGetPeriphStateApi (McuDebugConfiguration) 71
4.4.3 McuGetMidrStructureApi (McuDebugConfiguration) 71
4.4.4 McuDisableCmuApi (McuDebugConfiguration) 72
4.4.5 McuEnablePeripheralCMU (McuDebugConfiguration) 72
4.4.6 McuSRAMRetentionConfigApi (McuDebugConfiguration) 72
4.5 Form McuPublishedInformation 73
4.5.1 Form McuResetReasonConf 73
4.5.1.1 McuResetReason (McuResetReasonConf) 74
4.6 Form McuModuleConfiguration 74

4.6.1 McuNumberOfMcuModes (McuModuleConfiguration)....75
4.6.2 McuRamSectors (McuModuleConfiguration)....75
4.6.3 McuResetSetting (McuModuleConfiguration)....76
4.6.4 McuRTCCLKINFrequencyHz (McuModuleConfiguration)....76
4.6.5 McuClk32KSelect (McuModuleConfiguration)....77
4.6.6 McuLPOClockSelect (McuModuleConfiguration) McuClockSrcFailureNotification....77
4.6.7 Form McuAllowedModes....77
4.6.7.1 McuAllowHighSpeedRunMode (McuAllowedModes)....78
4.6.7.2 McuAllowVeryLowPowerModes (McuAllowedModes)....79
4.6.8 Form McuClockSettingConfig....79
4.6.8.1 McuClockSettingId (McuClockSettingConfig)....80
4.6.8.2 McuSysClockUnderMcuControl (McuClockSettingConfig)....80
4.6.8.3 McuScgClkOutSelect (McuClockSettingConfig)....81
4.6.8.4 Form McuRunClockConfig....82
4.6.8.4.1 McuPreDivSystemClockFrequency (McuRunClockConfig)....82
4.6.8.4.2 McuCoreClockFrequency (McuRunClockConfig)....83
4.6.8.4.3 McuSystemClockFrequency (McuRunClockConfig)....83
4.6.8.4.4 McuBusClockFrequency (McuRunClockConfig)....84
4.6.8.4.5 McuFlashClockFrequency (McuRunClockConfig)....84
4.6.8.4.6 McuSystemClockSwitch (McuRunClockConfig)....85
4.6.8.4.7 McuCoreClockDivider (McuRunClockConfig)....85
4.6.8.4.8 McuBusClockDivider (McuRunClockConfig)....86
4.6.8.4.9 McuSlowClockDivider (McuRunClockConfig)....86
4.6.8.4.10 McuScgClkOutFrequency (McuRunClockConfig)....87
4.6.8.5 Form McuVlprClockConfig....87
4.6.8.5.1 McuPreDivSystemClockFrequency (McuVlprClockConfig)....88
4.6.8.5.2 McuCoreClockFrequency (McuVlprClockConfig)....88
4.6.8.5.3 McuSystemClockFrequency (McuVlprClockConfig)....89
4.6.8.5.4 McuBusClockFrequency (McuVlprClockConfig)....89

4.6.8.5.5 McuFlashClockFrequency (McuVlprClockConfig) ... 90
4.6.8.5.6 McuSystemClockSwitch (McuVlprClockConfig) ... 90
4.6.8.5.7 McuCoreClockDivider (McuVlprClockConfig) ... 91
4.6.8.5.8 McuBusClockDivider (McuVlprClockConfig) ... 91
4.6.8.5.9 McuSlowClockDivider (McuVlprClockConfig) ... 92
4.6.8.5.10 McuScgClkOutFrequency (McuVlprClockConfig) ... 92
4.6.8.6 Form McuHsrunClockConfig ... 93
4.6.8.6.1 McuPreDivSystemClockFrequency (McuHsrunClockConfig) ... 93
4.6.8.6.2 McuCoreClockFrequency (McuHsrunClockConfig) ... 94
4.6.8.6.3 McuSystemClockFrequency (McuHsrunClockConfig) ... 94
4.6.8.6.4 McuBusClockFrequency (McuHsrunClockConfig) ... 95
4.6.8.6.5 McuFlashClockFrequency (McuHsrunClockConfig) ... 95
4.6.8.6.6 McuSystemClockSwitch (McuHsrunClockConfig) ... 96
4.6.8.6.7 McuCoreClockDivider (McuHsrunClockConfig) ... 96
4.6.8.6.8 McuBusClockDivider (McuHsrunClockConfig) ... 97
4.6.8.6.9 McuSlowClockDivider (McuHsrunClockConfig) ... 97
4.6.8.6.10 McuScgClkOutFrequency (McuHsrunClockConfig) ... 98
4.6.8.7 Form McuSystemOSCClockConfig ... 98
4.6.8.7.1 McuSOSCUnderMcuControl (McuSystemOSCClockConfig) ... 99
4.6.8.7.2 McuSOSCFrequency (McuSystemOSCClockConfig) ... 99
4.6.8.7.3 McuSOSCDiv2Frequency (McuSystemOSCClockConfig) ... 100
4.6.8.7.4 McuSOSCDiv1Frequency (McuSystemOSCClockConfig) ... 100
4.6.8.7.5 McuSOSCEnable (McuSystemOSCClockConfig) ... 101
4.6.8.7.6 McuSOSCClockMonitorResetEnable (McuSystemOSCClockConfig) ... 101
4.6.8.7.7 McuSOSCClockMonitorEnable (McuSystemOSCClockConfig) ... 102
4.6.8.7.8 McuSOSCDiv2 (McuSystemOSCClockConfig) ... 102
4.6.8.7.9 McuSOSCDiv1 (McuSystemOSCClockConfig) ... 103
4.6.8.7.10 McuSOSCRangeSelect (McuSystemOSCClockConfig) ... 104
4.6.8.7.11 McuSOSCHighGainOscillatorSelect (McuSystemOSCClockConfig) ... 104

4.6.8.7.12 McuSOSCExternalReferenceSelect (McuSystemOSCClockConfig)..... 105
4.6.8.8 Form McuSIRCClockConfig..... 106
4.6.8.8.1 McuSIRCUnderMcuControl (McuSIRCClockConfig)..... 106
4.6.8.8.2 McuSIRCFrequency (McuSIRCClockConfig)..... 107
4.6.8.8.3 McuSIRCDiv2Frequency (McuSIRCClockConfig)..... 107
4.6.8.8.4 McuSIRCDiv1Frequency (McuSIRCClockConfig)..... 107
4.6.8.8.5 McuSIRCEnable (McuSIRCClockConfig)..... 108
4.6.8.8.6 McuSIRCLowPowerEnable (McuSIRCClockConfig)..... 108
4.6.8.8.7 McuSIRCStopEnable (McuSIRCClockConfig)..... 109
4.6.8.8.8 McuSIRCDiv2 (McuSIRCClockConfig)..... 109
4.6.8.8.9 McuSIRCDiv1 (McuSIRCClockConfig)..... 110
4.6.8.8.10 McuSIRCRangeSelect (McuSIRCClockConfig)..... 110
4.6.8.9 Form McuFIRCClockConfig..... 111
4.6.8.9.1 McuFIRCUnderMcuControl (McuFIRCClockConfig)..... 112
4.6.8.9.2 McuFIRCFrequency (McuFIRCClockConfig)..... 112
4.6.8.9.3 McuFIRCDiv2Frequency (McuFIRCClockConfig)..... 113
4.6.8.9.4 McuFIRCDiv1Frequency (McuFIRCClockConfig)..... 113
4.6.8.9.5 McuFIRCEnable (McuFIRCClockConfig)..... 114
4.6.8.9.6 McuFIRCRegulatorEnable (McuFIRCClockConfig)..... 114
4.6.8.9.7 McuFIRCDiv2 (McuFIRCClockConfig)..... 115
4.6.8.9.8 McuFIRCDiv1 (McuFIRCClockConfig)..... 115
4.6.8.9.9 McuFIRCRangeSelect (McuFIRCClockConfig)..... 116
4.6.8.10 Form McuSystemPll..... 117
4.6.8.10.1 McuSystemPllUnderMcuControl (McuSystemPll)..... 117
4.6.8.10.2 McuSPLLFrequency (McuSystemPll)..... 117
4.6.8.10.3 McuSPLLDiv2Frequency (McuSystemPll)..... 118
4.6.8.10.4 McuSPLLDiv1Frequency (McuSystemPll)..... 118
4.6.8.10.5 McuSPLLEnable (McuSystemPll)..... 119
4.6.8.10.6 McuSPLLClockMonitorResetEnable (McuSystemPll)..... 119

4.6.8.10.7 McuSPLLClockMonitorEnable (McuSystemPll)....120
4.6.8.10.8 McuSPLLDiv2 (McuSystemPll)....120
4.6.8.10.9 McuSPLLDiv1 (McuSystemPll)....121
4.6.8.10.10 McuSPLLInputClkPreDivider (McuSystemPll)....122
4.6.8.10.11 McuSPLLInputFrequency (McuSystemPll)....122
4.6.8.10.12 McuSPLLMultiplier (McuSystemPll)....123
4.6.8.11 Form McuSIMClockConfig....123
4.6.8.11.1 McuDebugTraceDividerEnable (McuSIMClockConfig)....124
4.6.8.11.2 McuTraceClockDivider (McuSIMClockConfig)....124
4.6.8.11.3 McuTraceClockFraction (McuSIMClockConfig)....125
4.6.8.11.4 McuTraceClockSelect (McuSIMClockConfig)....125
4.6.8.11.5 McuClockOutEnable (McuSIMClockConfig)....126
4.6.8.11.6 McuClockOutDivider (McuSIMClockConfig)....126
4.6.8.11.7 McuClockOutSelect (McuSIMClockConfig)....127
4.6.8.11.8 McuEIMClockGatingEnable (McuSIMClockConfig)....128
4.6.8.11.9 McuERMClockGatingEnable (McuSIMClockConfig)....129
4.6.8.11.10 McuDMAClockGatingEnable (McuSIMClockConfig)....129
4.6.8.11.11 McuMPUClockGatingEnable (McuSIMClockConfig)....130
4.6.8.11.12 McuMSCMClockGatingEnable (McuSIMClockConfig)....130
4.6.8.11.13 McuGPIOClockGatingEnable (McuSIMClockConfig)....131
4.6.8.12 Form McuPeripheralClockConfig....131
4.6.8.12.1 McuPerName (McuPeripheralClockConfig)....132
4.6.8.12.2 McuPeripheralClockEnable (McuPeripheralClockConfig)....132
4.6.8.12.3 McuPeripheralClockSelect (McuPeripheralClockConfig)....133
4.6.8.12.4 McuPeripheralClockDivider (McuPeripheralClockConfig)....134
4.6.8.12.5 McuPeripheralFractionalDivider (McuPeripheralClockConfig)....135
4.6.8.12.6 McuPeripheralClockFrequency (McuPeripheralClockConfig)....135
4.6.8.13 Form McuClkMonitor_0....136
4.6.8.13.1 McuClkMonitorEn (McuClkMonitor_0)....136

4.6.8.13.2 McuFrequencyHigherInterrupt (McuClkMonitor_0) 137
4.6.8.13.3 McuFrequencyLowerInterrupt (McuClkMonitor_0) 137
4.6.8.13.4 McuReferenceCountConfiguration (McuClkMonitor_0) 138
4.6.8.13.5 McuHighFrequencyRef (McuClkMonitor_0) 138
4.6.8.13.6 McuLowFrequencyRef (McuClkMonitor_0) 139
4.6.8.14 Form McuClkMonitor_1 139
4.6.8.14.1 McuClkMonitorEn (McuClkMonitor_1) 140
4.6.8.14.2 McuFrequencyHigherInterrupt (McuClkMonitor_1) 140
4.6.8.14.3 McuFrequencyLowerInterrupt (McuClkMonitor_1) 140
4.6.8.14.4 McuReferenceCountConfiguration (McuClkMonitor_1) 141
4.6.8.14.5 McuHighFrequencyRef (McuClkMonitor_1) 141
4.6.8.14.6 McuLowFrequencyRef (McuClkMonitor_1) 142
4.6.8.15 Form McuClockReferencePoint 142
4.6.8.15.1 McuClockReferencePointFrequency (McuClockReferencePoint) 143
4.6.8.15.2 McuClockFrequencySelect (McuClockReferencePoint) 143
4.6.9 Form McuDemEventParameterRefs 145
4.6.9.1 MCU_E_TIMEOUT_FAILURE (McuDemEventParameterRefs) 145
4.6.9.2 MCU_E_CLOCK_FAILURE (McuDemEventParameterRefs) 145
4.6.10 Form McuInterruptEvents 146
4.6.10.1 McuVoltageErrorEvent (McuInterruptEvents) 146
4.6.10.2 McuAlternateResetEvent (McuInterruptEvents) 146
4.6.11 Form McuModeSettingConf 147
4.6.11.1 McuMode (McuModeSettingConf) 147
4.6.11.2 McuPowerMode (McuModeSettingConf) 148
4.6.12 Form McuPowerControl 148
4.6.12.1 McuLowVoltageDetectInterruptEnable (McuPowerControl) 149
4.6.12.2 McuLowVoltageDetectResetEnable (McuPowerControl) 149
4.6.12.3 McuLowVoltageWarningInterruptEnable (McuPowerControl) 150
4.6.12.4 McuLPODisable (McuPowerControl) 150

4.6.12.5 McuClockBiasDisable (McuPowerControl)..... 151
4.6.12.6 McuLowPowerBiasEnable (McuPowerControl)..... 151
4.6.12.7 McuLpoTrimming (McuPowerControl)..... 152
4.6.13 Form McuRamSectorSettingConf..... 153
4.6.13.1 McuRamSectorId (McuRamSectorSettingConf)..... 153
4.6.13.2 McuRamDefaultValue (McuRamSectorSettingConf)..... 154
4.6.13.3 McuRamSectionBaseAddress (McuRamSectorSettingConf)..... 154
4.6.13.4 McuRamSectionWriteSize (McuRamSectorSettingConf)..... 155
4.6.13.5 McuRamSectionSize (McuRamSectorSettingConf)..... 155
4.6.13.6 McuRamSectionBaseAddrLinkerSym (McuRamSectorSettingConf)..... 156
4.6.13.7 McuRamSectionSizeLinkerSym (McuRamSectorSettingConf)..... 156
4.6.14 Form McuResetConfig..... 157
4.6.14.1 McuResetPinFilterBusClockSelect (McuResetConfig)..... 157
4.6.14.2 McuResetPinFilterInStopMode (McuResetConfig)..... 158
4.6.14.3 McuResetPinFilterInRunAndWait (McuResetConfig)..... 158
4.6.14.4 Form McuSystemInterruptEnable..... 159
4.6.14.4.1 McuResetDelayTime (McuSystemInterruptEnable)..... 160
4.6.14.4.2 McuStopAcknowledgeErrorInterrupt (McuSystemInterruptEnable)..... 160
4.6.14.4.3 McuMDMAPSystemResetInterrupt (McuSystemInterruptEnable)..... 161
4.6.14.4.4 McuSoftwareInterrupt (McuSystemInterruptEnable)..... 161
4.6.14.4.5 McuCoreLockupInterrupt (McuSystemInterruptEnable)..... 162
4.6.14.4.6 McuJTAGResetInterrupt (McuSystemInterruptEnable)..... 162
4.6.14.4.7 McuGlobalInterrupt (McuSystemInterruptEnable)..... 163
4.6.14.4.8 McuExternalResetPinInterrupt (McuSystemInterruptEnable)..... 163
4.6.14.4.9 McuWatchdogInterrupt (McuSystemInterruptEnable)..... 164
4.6.14.4.10 McuCMULossOfClockResetInterrupt (McuSystemInterruptEnable)..... 164
4.6.14.4.11 McuLossOfLockInterrupt (McuSystemInterruptEnable)..... 165
4.6.14.4.12 McuLossOfClockInterrupt (McuSystemInterruptEnable)..... 165
4.6.15 Form McuSIMConfig..... 166

4.6.15.1 Form McuChipControlConfiguration ... 166
4.6.15.1.1 McuEnableAdcSupplyMonitoring (McuChipControlConfiguration) ... 167
4.6.15.1.2 McuAdcSupply (McuChipControlConfiguration) ... 167
4.6.15.1.3 McuPDBBackToBackSelect (McuChipControlConfiguration) ... 168
4.6.15.1.4 McuPTB14InterleaveChannelSelect (McuChipControlConfiguration) ... 169
4.6.15.1.5 McuPTB13InterleaveChannelSelect (McuChipControlConfiguration) ... 169
4.6.15.1.6 McuPTB1InterleaveChannelSelect (McuChipControlConfiguration) ... 170
4.6.15.1.7 McuPTB0InterleaveChannelSelect (McuChipControlConfiguration) ... 170
4.6.15.2 Form McuFlexTimerConfiguration ... 171
4.6.15.2.1 McuFTM3ExternalClockPinSelect (McuFlexTimerConfiguration) ... 172
4.6.15.2.2 McuFTM2ExternalClockPinSelect (McuFlexTimerConfiguration) ... 173
4.6.15.2.3 McuFTM1ExternalClockPinSelect (McuFlexTimerConfiguration) ... 174
4.6.15.2.4 McuFTM0ExternalClockPinSelect (McuFlexTimerConfiguration) ... 175
4.6.15.2.5 McuFTM7ExternalClockPinSelect (McuFlexTimerConfiguration) ... 176
4.6.15.2.6 McuFTM6ExternalClockPinSelect (McuFlexTimerConfiguration) ... 177
4.6.15.2.7 McuFTM5ExternalClockPinSelect (McuFlexTimerConfiguration) ... 177
4.6.15.2.8 McuFTM4ExternalClockPinSelect (McuFlexTimerConfiguration) ... 178
4.6.15.2.9 McuFTM3Fault0Select (McuFlexTimerConfiguration) ... 179
4.6.15.2.10 McuFTM3Fault1Select (McuFlexTimerConfiguration) ... 180
4.6.15.2.11 McuFTM3Fault2Select (McuFlexTimerConfiguration) ... 180
4.6.15.2.12 McuFTM2Fault0Select (McuFlexTimerConfiguration) ... 181
4.6.15.2.13 McuFTM2Fault1Select (McuFlexTimerConfiguration) ... 182
4.6.15.2.14 McuFTM2Fault2Select (McuFlexTimerConfiguration) ... 182
4.6.15.2.15 McuFTM1Fault0Select (McuFlexTimerConfiguration) ... 183
4.6.15.2.16 McuFTM1Fault1Select (McuFlexTimerConfiguration) ... 184
4.6.15.2.17 McuFTM1Fault2Select (McuFlexTimerConfiguration) ... 184
4.6.15.2.18 McuFTM0Fault0Select (McuFlexTimerConfiguration) ... 185
4.6.15.2.19 McuFTM0Fault1Select (McuFlexTimerConfiguration) ... 186
4.6.15.2.20 McuFTM0Fault2Select (McuFlexTimerConfiguration) ... 186

4.6.15.2.21 McuFTM3Ch0ModulationSelect (McuFlexTimerConfiguration).....187
4.6.15.2.22 McuFTM3Ch1ModulationSelect (McuFlexTimerConfiguration).....187
4.6.15.2.23 McuFTM3Ch2ModulationSelect (McuFlexTimerConfiguration).....188
4.6.15.2.24 McuFTM3Ch3ModulationSelect (McuFlexTimerConfiguration).....188
4.6.15.2.25 McuFTM3Ch4ModulationSelect (McuFlexTimerConfiguration).....189
4.6.15.2.26 McuFTM3Ch5ModulationSelect (McuFlexTimerConfiguration).....189
4.6.15.2.27 McuFTM3Ch6ModulationSelect (McuFlexTimerConfiguration).....190
4.6.15.2.28 McuFTM3Ch7ModulationSelect (McuFlexTimerConfiguration).....190
4.6.15.2.29 McuFTM0Ch0ModulationSelect (McuFlexTimerConfiguration).....191
4.6.15.2.30 McuFTM0Ch1ModulationSelect (McuFlexTimerConfiguration).....191
4.6.15.2.31 McuFTM0Ch2ModulationSelect (McuFlexTimerConfiguration).....192
4.6.15.2.32 McuFTM0Ch3ModulationSelect (McuFlexTimerConfiguration).....192
4.6.15.2.33 McuFTM0Ch4ModulationSelect (McuFlexTimerConfiguration).....193
4.6.15.2.34 McuFTM0Ch5ModulationSelect (McuFlexTimerConfiguration).....193
4.6.15.2.35 McuFTM0Ch6ModulationSelect (McuFlexTimerConfiguration).....194
4.6.15.2.36 McuFTM0Ch7ModulationSelect (McuFlexTimerConfiguration).....194
4.6.15.2.37 McuFTM2Ch1InputSelect (McuFlexTimerConfiguration).....195
4.6.15.2.38 McuFTM2Ch0InputSelect (McuFlexTimerConfiguration).....195
4.6.15.2.39 McuFTM1Ch0InputSelect (McuFlexTimerConfiguration).....196
4.6.15.2.40 McuFTMGlobalLoadEnable (McuFlexTimerConfiguration)..... 197
4.6.15.2.41 McuFTM7SyncBit (McuFlexTimerConfiguration).....197
4.6.15.2.42 McuFTM6SyncBit (McuFlexTimerConfiguration).....198
4.6.15.2.43 McuFTM5SyncBit (McuFlexTimerConfiguration).....198
4.6.15.2.44 McuFTM4SyncBit (McuFlexTimerConfiguration).....198
4.6.15.2.45 McuFTM3SyncBit (McuFlexTimerConfiguration).....199
4.6.15.2.46 McuFTM2SyncBit (McuFlexTimerConfiguration).....199
4.6.15.2.47 McuFTM1SyncBit (McuFlexTimerConfiguration).....200
4.6.15.2.48 McuFTM0SyncBit (McuFlexTimerConfiguration).....200
4.6.15.2.49 McuQspiClkSelect (McuFlexTimerConfiguration).....200

4.6.15.2.50 McuRMII_ClkSelect (McuFlexTimerConfiguration)..... 201
4.6.15.2.51 McuRMII_Clk_OBE (McuFlexTimerConfiguration)..... 201
4.6.15.2.52 McuFTM7OBEControl (McuFlexTimerConfiguration)..... 202
4.6.15.2.53 McuFTM6OBEControl (McuFlexTimerConfiguration)..... 203
4.6.15.2.54 McuFTM5OBEControl (McuFlexTimerConfiguration)..... 203
4.6.15.2.55 McuFTM4OBEControl (McuFlexTimerConfiguration)..... 204
4.6.15.2.56 McuFTM3OBEControl (McuFlexTimerConfiguration)..... 204
4.6.15.2.57 McuFTM2OBEControl (McuFlexTimerConfiguration)..... 205
4.6.15.2.58 McuFTM1OBEControl (McuFlexTimerConfiguration)..... 206
4.6.15.2.59 McuFTM0OBEControl (McuFlexTimerConfiguration)..... 206
4.6.15.2.60 McuFTM_GTBControl (McuFlexTimerConfiguration)..... 207
4.6.15.3 Form McuAdcOptionsConfiguration..... 207
4.6.15.3.1 McuADC1PreTrigeerSourceSelect (McuAdcOptionsConfiguration)..... 208
4.6.15.3.2 McuADC1SoftwarePreTrigeerSourceSelect (McuAdcOptionsConfiguration)..... 208
4.6.15.3.3 McuADC1TrigeerSourceSelect (McuAdcOptionsConfiguration)..... 209
4.6.15.3.4 McuADC0PreTrigeerSourceSelect (McuAdcOptionsConfiguration)..... 210
4.6.15.3.5 McuADC0SoftwarePreTrigeerSourceSelect (McuAdcOptionsConfiguration)..... 210
4.6.15.3.6 McuADC0TrigeerSourceSelect (McuAdcOptionsConfiguration)..... 211
4.6.15.3.7 McuSoftwareTriggerToTRGMUX (McuAdcOptionsConfiguration)..... 212
4.7 Form CommonPublishedInformation..... 212
4.7.1 ArReleaseMajorVersion (CommonPublishedInformation)..... 213
4.7.2 ArReleaseMinorVersion (CommonPublishedInformation)..... 213
4.7.3 ArReleaseRevisionVersion (CommonPublishedInformation)..... 214
4.7.4 ModuleId (CommonPublishedInformation)..... 214
4.7.5 SwMajorVersion (CommonPublishedInformation)..... 215
4.7.6 SwMinorVersion (CommonPublishedInformation)..... 215
4.7.7 SwPatchVersion (CommonPublishedInformation)..... 216
4.7.8 VendorApiInfix (CommonPublishedInformation)..... 216
4.7.9 VendorId (CommonPublishedInformation)..... 217

Chapter 1
Revision History

Table 1-1. Revision History

Revision	Date	Author	Description
1.0	21/06/2019	NXP MCAL Team	Updated version for ASR 4.3.1S32K14XR1.0.1

Chapter 2
Introduction

This User Manual describes NXP Semiconductors AUTOSAR Micro Control Unit (Mcu) for S32K14X .

AUTOSAR Mcu driver configuration parameters and deviations from the specification are described in Mcu Driver chapter of this document. AUTOSAR Mcu driver requirements and APIs are described in the AUTOSAR Mcu driver software specification document.

2.1 Supported Derivatives

The software described in this document is intented to be used with the following microcontroller devices of NXP Semiconductors .

Table 2-1. S32K14X Derivatives

NXP Semiconductors	s32k148_lqfp144, s32k148_lqfp176, s32k148_mapbga100, s32k146_lqfp144, s32k146_lqfp100, s32k146_lqfp64, s32k146_mapbga100, s32k144_lqfp100, s32k144_lqfp64, s32k144_mapbga100, s32k142_lqfp100, s32k142_lqfp64, s32k118_lqfp48, s32k118_lqfp64, s32k142_lqfp48, s32k144_lqfp48, s32k148_lqfp100

All of the above microcontroller devices are collectively named as S32K14X .

2.2 Overview

AUTOSAR (AUTomotive Open System ARchitecture) is an industry partnership working to establish standards for software interfaces and software modules for automobile electronic control systems.

AUTOSAR

- paves the way for innovative electronic systems that further improve performance, safety and environmental friendliness.
- is a strong global partnership that creates one common standard: "Cooperate on standards, compete on implementation".
- is a key enabling technology to manage the growing electrics/electronics complexity. It aims to be prepared for the upcoming technologies and to improve cost-efficiency without making any compromise with respect to quality.
- facilitates the exchange and update of software and hardware over the service life of the vehicle.

2.3 About this Manual

This Technical Reference employs the following typographical conventions:

Boldface type: Bold is used for important terms, notes and warnings.

Italic font: Italic typeface is used for code snippets in the text. Note that C language modifiers such "const" or "volatile" are sometimes omitted to improve readability of the presented code.

Notes and warnings are shown as below:

Note

This is a note.

2.4 Acronyms and Definitions

Table 2-2. Acronyms and Definitions

Term	Definition
API	Application Programming Interface
AUTOSAR	Automotive Open System Architecture
ASM	Assembler
BSMI	Basic Software Make file Interface
CAN	Controller Area Network
DEM	Diagnostic Event Manager
DET	Default Error Tracer
C/CPP	C and C++ Source Code
VLE	Variable Length Encoding

Table continues on the next page...

Table 2-2. Acronyms and Definitions (continued)

Term	Definition
N/A	Not Applicable
MCU	Micro Controller Unit

2.5 Reference List

Table 2-3. Reference List

#	Title	Version
1	Specification of Mcu Driver	AUTOSAR Release 4.3.1
2	S32K14X Reference Manual	Reference Manual, Rev. 9, 9/2018
3	S32K142 Mask Set Errata for Mask 0N33V (0N33V)	30/11/2017
4	S32K144 Mask Set Errata for Mask 0N57U (0N57U)	30/11/2017
5	S32K146 Mask Set Errata for Mask 0N73V (0N73V)	30/11/2017
6	S32K148 Mask Set Errata for Mask 0N20V (0N20V)	25/10/2018
7	S32K118 Mask Set Errata for Mask 0N97V (0N97V)	07/01/2019

Chapter 3
Driver

3.1 Requirements

Requirements for this driver are detailed in the AUTOSAR 4.3 Rev0001Mcu Driver Software Specification document (See Table Reference List).

3.2 Driver Design Summary

The S32K14X contains the following blocks:

- IPV_SCG
- IPV_SIM
- IPV_PCC
- IPV_PMC
- IPV_SMC
- IPV_CORTEXM4
- IPV_RCM
- IPV_CMU

IPV_SCG controls the System Clock Generation.

The **IPV_SCG** covers the following IP's:

- **SCG** System Clock Generation

IPV_SIM controls the System Integration Module.

The **IPV_SIM** covers the following IP's:

- **SIM** System Integration Module

IPV_PCC controls the Peripheral Clock.

The **IPV_PCC** covers the following IP's:

- **PCC** Peripheral Clock Control

IPV_PMC controls the power management controller.

The **IPV_PMC** covers the following IP's:

- **PMC** Power Management Controller

IPV_SMC controls the System Mode.

The **IPV_SMC** covers the following IP's:

- **SMC** System Mode Control

IPV_CORTEXM controls the General Operation.

The **IPV_CORTEXM** covers the following IP's:

- **CORTEXM4** CortexM4 Registers

IPV_RCM Reset Controller Module.

The **IPV_RCM** covers the following IP's:

- **RCM** Reset Controller Module

IPV_CMU controls the Clock Monitoring Unit.

The **IPV_CMU** covers the following IP's:

- **CMU** Clock Monitoring Unit

3.3 Hardware Resources

The hardware configured by the Mcu driver is the same between derivatives.

3.4 Deviation from Requirements

The driver deviates from the AUTOSAR Mcu Driver software specification in some places. The table identifies the AUTOSAR requirements that are not fully implemented, implemented differently, or out of scope for the Mcu Driver. Table Table 3-1 provides Status column description.

Table 3-1. Deviations Status Column Description

Term	Definition
N/A	Not available
N/T	Not testable
N/S	Out of scope
N/I	Not implemented
N/F	Not fully implemented

Below table identifies the AUTOSAR requirements that are not fully implemented, implemented differently, or out of scope for the driver.

Table 3-2. Driver Deviations Table

Requirement	Status	Description	Notes
SWS_Mcu_00215	N/S	The type definitions for Mcu_Lcfg.c and Mcu_PBcfg.c are located in the file Mcu.h.	Driver design doesn't include Link-time configuration support.
SWS_Mcu_00216	N/S	Mcu_Lcfg.c shall include Mcu_Cbk.h for a link time configuration if the call back function is linked to the module via the ROM structure.	Driver design doesn't include Link-time configuration support.
SWS_Mcu_00218	N/S	Mcu_PBcfg.c shall include Mcu_Cbk.h for post build time configuration if the call back function is linked to the module via the ROM structure.	Driver design doesn't include Link-time configuration support.
SWS_Mcu_00245	N/S	If the register can affect several hardware modules and if it is not an I/O register, it shall be initialised by this MCU driver. (BSW12125, BSW12461)	There is a separate plug-in that will cover shared ip's
SWS_Mcu_00056	N/S	The function Mcu_DistributePllClock shall return without affecting the MCU hardware if the PLL clock has been automatically activated by the MCU hardware. (BSW12336)	The function Mcu_DistributePllClock will change the Mcu hardware. The clock switching to PLL is not completed by Mcu_InitClock
SWS_Mcu_00053	N/S	If clock failure notification is enabled in the configuration set and a clock source failure error occurs, the error code MCU_E_CLOCK_FAILURE shall be reported. (See also SWS_Mcu_00051). (BSW12394)	no hardware support
SWS_Mcu_00257	N/S	Fail criteria for MCU_E_CLOCK_FAILURE: a clock source failure occurs	For the clock failure case the error MCU_E_ISR_CLOCK_FAILURE was defined and if the error notification is configured by the application the CMU interrupt will report it.
SWS_Mcu_00258	N/S	Pass criteria for MCU_E_CLOCK_FAILURE: no clock source failure occurs	For the clock failure case the error MCU_E_ISR_CLOCK_FAILURE was defined and if the error notification is configured by the application the CMU interrupt will report it.

3.5 Driver Limitation

None

3.6 Driver usage and configuration tips

For reconfiguring the PLLs using Mcu_InitClock and Mcu_DistributePllClock the peripherals that are clocked using the PLL that needs to be reconfigured should be turned OFF using Mcu_SetMode to transition in a mode where that peripheral is OFF.

For bypassing the configuration of a clock source, the system clock or of auxiliary clocks during Mcu_InitClock the check box "[source] under MCU control" should be unchecked. This will generate smaller configurations that will be updated faster and more efficiently.

3.7 Runtime Errors

The driver generates the following DEM errors at runtime.

Table 3-3. Runtime Errors

Function	Error Code	Condition triggering the error
Mcu_Init	MCU_E_PARAM_CONFIG	Invalid input pointer
Mcu_InitClock	MCU_E_PARAM_CLOCK	Invalid input parameter
Mcu_SetMode	MCU_E_PARAM_MODE	Invalid input parameter
Mcu_InitRamSection	MCU_E_PARAM_RAMSECTION	Invalid input parameter or invalid memory configuration
Mcu_DistributePllClock	MCU_E_PLL_NOT_LOCKED	One of the used PLL's is failed to achieve lock
All functions, except Mcu_Init and Mcu_GetVersionInfo	MCU_E_UNINIT	The driver is in an uninitialized state
Mcu_GetVersionInfo	MCU_E_PARAM_POINTER	Invalid input parameter
Mcu_Init	MCU_E_ALLREADY_INITIALIZED	The driver is already initialized

3.8 Software specification

The following sections contains driver software specifications.

3.8.1 Define Reference

Constants supported by the driver are as per AUTOSAR Mcu Driver software specification Version 4.3 Rev0001 .

3.8.1.1 Define MCU_INSTANCE_ID

Table 3-4. Define MCU_INSTANCE_ID Description

Name	MCU_INSTANCE_ID
Initializer	(uint8)0x0U

3.8.1.2 Define MCU_MODULE_ID

Table 3-5. Define MCU_MODULE_ID Description

Name	MCU_MODULE_ID
Initializer	101

3.8.1.3 Define MCU_INIT_ID

Service Ids for MCU APIs.

Table 3-6. Define MCU_INIT_ID Description

Name	MCU_INIT_ID
Initializer	(uint8)0x00U

3.8.1.4 Define MCU_INITRAMSECTION_ID

Service Ids for MCU APIs.

Table 3-7. Define MCU_INITRAMSECTION_ID Description

Name	MCU_INITRAMSECTION_ID
Initializer	(uint8)0x01U

3.8.1.5 Define MCU_INITCLOCK_ID

Service Ids for MCU APIs.

Table 3-8. Define MCU_INITCLOCK_ID Description

Name	MCU_INITCLOCK_ID
Initializer	(uint8)0x02U

3.8.1.6 Define MCU_DISTRIBUTEPLLCLOCK_ID

Service Ids for MCU APIs.

Table 3-9. Define MCU_DISTRIBUTEPLLCLOCK_ID Description

Name	MCU_DISTRIBUTEPLLCLOCK_ID
Initializer	(uint8)0x03U

3.8.1.7 Define MCU_GETPLLSTATUS_ID

Service Ids for MCU APIs.

Table 3-10. Define MCU_GETPLLSTATUS_ID Description

Name	MCU_GETPLLSTATUS_ID
Initializer	(uint8)0x04U

3.8.1.8 Define MCU_GETRESETREASON_ID

Service Ids for MCU APIs.

Table 3-11. Define MCU_GETRESETREASON_ID Description

Name	MCU_GETRESETREASON_ID
Initializer	(uint8)0x05U

3.8.1.9 Define MCU_GETRESETRAWVALUE_ID

Service Ids for MCU APIs.

Table 3-12. Define MCU_GETRESETRAWVALUE_ID Description

Name	MCU_GETRESETRAWVALUE_ID
Initializer	(uint8)0x06U

3.8.1.10 Define MCU_PERFORMRESET_ID

Service Ids for MCU APIs.

Table 3-13. Define MCU_PERFORMRESET_ID Description

Name	MCU_PERFORMRESET_ID
Initializer	(uint8)0x07U

3.8.1.11 Define MCU_SETMODE_ID

Service Ids for MCU APIs.

Table 3-14. Define MCU_SETMODE_ID Description

Name	MCU_SETMODE_ID
Initializer	(uint8)0x08U

3.8.1.12 Define MCU_GETVERSIONINFO_ID

Service Ids for MCU APIs.

Table 3-15. Define MCU_GETVERSIONINFO_ID Description

Name	MCU_GETVERSIONINFO_ID
Initializer	(uint8)0x09U

3.8.1.13 Define MCU_GETRAMSTATE_ID

Service Ids for MCU APIs.

Table 3-16. Define MCU_GETRAMSTATE_ID Description

Name	MCU_GETRAMSTATE_ID
Initializer	(uint8)0x0AU

3.8.1.14 Define MCU_GETPOWERDOMAIN_ID

Service Ids for MCU APIs.

Table 3-17. Define MCU_GETPOWERDOMAIN_ID Description

Name	MCU_GETPOWERDOMAIN_ID
Initializer	(uint8)0x0BU

3.8.1.15 Define MCU_GETPERIPHERALSTATE_ID

Service Ids for MCU APIs.

Table 3-18. Define MCU_GETPERIPHERALSTATE_ID Description

Name	MCU_GETPERIPHERALSTATE_ID
Initializer	(uint8)0x0CU

3.8.1.16 Define MCU_GETSYSTEMSTATE_ID

Service Ids for MCU APIs.

Table 3-19. Define MCU_GETSYSTEMSTATE_ID Description

Name	MCU_GETSYSTEMSTATE_ID
Initializer	(uint8)0x0DU

3.8.1.17 Define MCU_GETPOWERMODESTATE_ID

Service Ids for MCU APIs.

Table 3-20. Define MCU_GETPOWERMODESTATE_ID Description

Name	MCU_GETPOWERMODESTATE_ID
Initializer	(uint8)0x0EU

3.8.1.18 Define MCU_GETMEMCONFIG_ID

Service Ids for MCU APIs.

Table 3-21. Define MCU_GETMEMCONFIG_ID Description

Name	MCU_GETMEMCONFIG_ID
Initializer	(uint8)0x13U

3.8.1.19 Define MCU_GETMIDRSTRUCTURE_ID

Service Ids for MCU APIs.

Table 3-22. Define MCU_GETMIDRSTRUCTURE_ID Description

Name	MCU_GETMIDRSTRUCTURE_ID
Initializer	(uint8)0x14U

3.8.1.20 Define MCU_SRAMRETENCONFIG_ID

Service ID for MCU APIs.

Table 3-23. Define MCU_SRAMRETENCONFIG_ID Description

Name	MCU_SRAMRETENCONFIG_ID
Initializer	(uint8)0x18U

3.8.1.21 Define MCU_SRAM_RETEN_CONFIG_API

The function Mcu_SRAMRetentionConfig is only available if the parameter McuSRAMRetentionConfigApi is set to TRUE

Table 3-24. Define MCU_SRAM_RETEN_CONFIG_API Description

Name	MCU_SRAM_RETEN_CONFIG_API
Initializer	STD_OFF

3.8.1.22 Define MCU_E_ISR_RESET_ALT_FAILURE

Error ISR values are of type uint8. The following error codes are reported by the error ISR.

Table 3-25. Define MCU_E_ISR_RESET_ALT_FAILURE Description

Name	MCU_E_ISR_RESET_ALT_FAILURE
Initializer	(uint8)0x08U

3.8.1.23 Define MCU_E_PARAM_CONFIG

Development error values are of type uint8. The following errors and exceptions shall be detectable by the MCU module depending on its build version (development/production mode).

If development error detection is enabled, the parameter ConfigPtr shall be checked for being NULL. If the parmameter is NULL, the error code MCU_E_PARAM_CONFIG shall be reported to the DET.

Table 3-26. Define MCU_E_PARAM_CONFIG Description

Name	MCU_E_PARAM_CONFIG
Initializer	((uint8)0x0AU)

3.8.1.24 Define MCU_E_PARAM_CLOCK

Development error values are of type uint8. The following errors and exceptions shall be detectable by the MCU module depending on its build version (development/production mode).

The ClockSetting shall be within the settings defined in the configuration data structure. If not, the error code MCU_E_PARAM_CLOCK shall be reported to the DET.

Table 3-27. Define MCU_E_PARAM_CLOCK Description

Name	MCU_E_PARAM_CLOCK
Initializer	((uint8)0x0BU)

3.8.1.25 Define MCU_E_PARAM_MODE

Development error values are of type uint8. The following errors and exceptions shall be detectable by the MCU module depending on its build version (development/production mode).

McuMode shall be within the modes defined in the configuration data structure. If not, the error code MCU_E_PARAM_MODE shall be reported to the DET.

Table 3-28. Define MCU_E_PARAM_MODE Description

Name	MCU_E_PARAM_MODE
Initializer	((uint8)0x0CU)

3.8.1.26 Define MCU_E_PARAM_RAMSECTION

Development error values are of type uint8. The following errors and exceptions shall be detectable by the MCU module depending on its build version (development/production mode).

RamSection shall be within the sections defined in the configuration data structure. If not, the error code MCU_E_PARAM_RAMSECTION shall be reported to the DET.

Table 3-29. Define MCU_E_PARAM_RAMSECTION Description

Name	MCU_E_PARAM_RAMSECTION
Initializer	((uint8)0x0DU)

3.8.1.27 Define MCU_E_PLL_NOT_LOCKED

Development error values are of type uint8. The following errors and exceptions shall be detectable by the MCU module depending on its build version (development/production mode).

The error shall be reported if the status of the PLL is detected as not locked with the function Mcu_DistributePllClock() .

Table 3-30. Define MCU_E_PLL_NOT_LOCKED Description

Name	MCU_E_PLL_NOT_LOCKED
Initializer	((uint8)0x0EU)

3.8.1.28 Define MCU_E_UNINIT

Development error values are of type uint8. The following errors and exceptions shall be detectable by the MCU module depending on its build version (development/production mode).

If development error detection is enabled and if any other function (except Mcu_GetVersionInfo) of the MCU module is called before Mcu_Init function, the error code MCU_E_UNINIT shall be reported to the DET.

Table 3-31. Define MCU_E_UNINIT Description

Name	MCU_E_UNINIT
Initializer	((uint8)0x0FU)

3.8.1.29 Define MCU_E_PARAM_POINTER

Development error values are of type uint8. The following errors and exceptions shall be detectable by the MCU module depending on its build version (development/production mode).

if development error detection is enabled, the parameter versioninfo shall be checked for being NULL. The error MCU_E_PARAM_POINTER shall be reported in case the value is a NULL pointer.

Table 3-32. Define MCU_E_PARAM_POINTER Description

Name	MCU_E_PARAM_POINTER
Initializer	((uint8)0x10U)

3.8.1.30 Define MCU_E_ALLREADY_INITIALIZED

Development error values are of type uint8. The following errors and exceptions shall be detectable by the MCU module depending on its build version (development/production mode).

Table 3-33. Define MCU_E_ALLREADY_INITIALIZED Description

Name	MCU_E_ALLREADY_INITIALIZED
Initializer	(uint8)0x13U

3.8.1.31 Define MCU_E_ISR_CLOCK_FAILURE

Error ISR values are of type uint8. The following error codes are reported by the error ISR.

Table 3-34. Define MCU_E_ISR_CLOCK_FAILURE Description

Name	MCU_E_ISR_CLOCK_FAILURE
Initializer	(uint8)0x01U

3.8.1.32 Define MCU_E_ISR_PLL_LOCK_FAILURE

Error ISR values are of type uint8. The following error codes are reported by the error ISR.

Table 3-35. Define MCU_E_ISR_PLL_LOCK_FAILURE Description

Name	MCU_E_ISR_PLL_LOCK_FAILURE
Initializer	((uint8)0x02U)

3.8.1.33 Define MCU_E_ISR_VOLTAGE_ERROR

Error ISR values are of type uint8. The following error codes are reported by the error ISR.

Table 3-36. Define MCU_E_ISR_VOLTAGE_ERROR Description

Name	MCU_E_ISR_VOLTAGE_ERROR
Initializer	((uint8)0x06U)

3.8.1.34 Define MCU_E_ISR_TEMPERATURE_MONITOR_ERROR

Error ISR values are of type uint8. The following error codes are reported by the error ISR.

Table 3-37. Define MCU_E_ISR_TEMPERATURE_MONITOR_ERROR Description

Name	MCU_E_ISR_TEMPERATURE_MONITOR_ERROR
Initializer	((uint8)0x07U)

3.8.1.35 Define MCU_DEV_ERROR_DETECT

(MCU166)Pre-processor switch for enabling the development error detection and reporting to the DET.

Satisfied Requirements: MCU166

Table 3-38. Define MCU_DEV_ERROR_DETECT Description

Name	MCU_DEV_ERROR_DETECT
Initializer	(STD_ON)

3.8.1.36 Define MCU_VERSION_INFO_API

(MCU168)Pre-processor switch to enable/disable the API to read out the modules version information.

Satisfied Requirements: MCU168_conf

Table 3-39. Define MCU_VERSION_INFO_API Description

Name	MCU_VERSION_INFO_API
Initializer	(STD_ON)

3.8.1.37 Define MCU_GET_RAM_STATE_API

(MCU181)Pre-processor switch to enable/disable the API Mcu_GetRamState.

Satisfied Requirements: MCU181_conf

Table 3-40. Define MCU_GET_RAM_STATE_API Description

Name	MCU_GET_RAM_STATE_API
Initializer	(STD_ON)

3.8.1.38 Define MCU_INIT_CLOCK

(MCU182)If this parameter is set to FALSE, the clock initialization has to be disabled from the MCU driver.

Satisfied Requirements: MCU182_conf

Table 3-41. Define MCU_INIT_CLOCK Description

Name	MCU_INIT_CLOCK
Initializer	(STD_ON)

3.8.1.39 Define MCU_NO_PLL

(MCU180)This parameter shall be set True, if the H/W does not have a PLL or the PLL circuitry is enabled after the power on without S/W intervention.

Satisfied Requirements: MCU180_conf

Table 3-42. Define MCU_NO_PLL Description

Name	MCU_NO_PLL
Initializer	(STD_OFF)

3.8.1.40 Define MCU_ENTER_LOW_POWER_MODE

Support for Low Power mode. If this parameter has been configured to 'TRUE', the function 'Mcu_SetMode()' shall not be impacted and behave as specified.

Satisfied Requirements: PR-MCAL-3184

Table 3-43. Define MCU_ENTER_LOW_POWER_MODE Description

Name	MCU_ENTER_LOW_POWER_MODE
Initializer	(STD_ON)

3.8.1.41 Define MCU_PERFORM_RESET_API

(MCU146)The function Mcu_PerformReset is only available if the runtime parameter McuPerformResetApi is set to TRUE.

Satisfied Requirements: MCU146

Table 3-44. Define MCU_PERFORM_RESET_API Description

Name	MCU_PERFORM_RESET_API
Initializer	(STD_ON)

3.8.1.42 Define MCU_TIMEOUT_LOOPS

Timeout representing the number of loops for preventing to lock inside am infinite while/ for.

Table 3-45. Define MCU_TIMEOUT_LOOPS Description

Name	MCU_TIMEOUT_LOOPS
Initializer	(uint32)10000U

3.8.1.43 Define MCU_RESET_CALLOUT_USED

The user callout reset is/isn't available (STD_ON/STD_OFF) - called by MCU right before`Mcu_PerformReset()`.

Table 3-46. Define MCU_RESET_CALLOUT_USED Description

Name	MCU_RESET_CALLOUT_USED
Initializer	(STD_OFF)

3.8.1.44 Define MCU_ERROR_ISR_NOTIFICATION

The callout configured by the user for CMU notifications.

Table 3-47. Define MCU_ERROR_ISR_NOTIFICATION Description

Name	MCU_ERROR_ISR_NOTIFICATION
Initializer	FUNC(void, MCU_CODE) MCU_ERROR_ISR_NOTIFICATION(VAR (uint8, AUTOMATIC) u8ErrorCode)

3.8.1.45 Define MCU_VOLTAGE_ERROR_ISR_USED

ISR Mcu_VoltageError_ISR is/isn't available (STD_ON/STD_OFF).

Table 3-48. Define MCU_VOLTAGE_ERROR_ISR_USED Description

Name	MCU_VOLTAGE_ERROR_ISR_USED
Initializer	(STD_OFF)

3.8.1.46 Define MCU_TEMPERATURE_ERROR_ISR_USED

ISR Mcu_VoltageError_ISR is/isn't available (STD_ON/STD_OFF).

Table 3-49. Define MCU_TEMPERATURE_ERROR_ISR_USED Description

Name	MCU_TEMPERATURE_ERROR_ISR_USED
Initializer	(STD_OFF)

3.8.1.47 Define MCU_GET_PERIPH_STATE_API

Enable the usage of Non-Autosar API Mcu_GetPeripheral_State() for getting infos about peripheral state from MC_ME module.

Table 3-50. Define MCU_GET_PERIPH_STATE_API Description

Name	MCU_GET_PERIPH_STATE_API
Initializer	(STD_OFF)

3.8.1.48 Define MCU_GET_SYSTEM_STATE_API

Enable the usage of Non-Autosar API Mcu_GetSystem_State() for getting infos system platform configuration.

Table 3-51. Define MCU_GET_SYSTEM_STATE_API Description

Name	MCU_GET_SYSTEM_STATE_API
Initializer	(STD_OFF)

3.8.1.49 Define MCU_POWERMODE_STATE_API

Enable the usage of Non-Autosar API Mcu_GetPowerMode_State() for getting infos system platform configuration.

Table 3-52. Define MCU_POWERMODE_STATE_API Description

Name	MCU_POWERMODE_STATE_API
Initializer	(STD_OFF)

3.8.1.50 Define MCU_DISABLE_DEM_REPORT_ERROR_STATUS

Enable/Disable the API for reporting the Dem Error.

Table 3-53. Define MCU_DISABLE_DEM_REPORT_ERROR_STATUS Description

Name	MCU_DISABLE_DEM_REPORT_ERROR_STATUS
Initializer	(STD_OFF)

3.8.1.51 Define MCU_FAST_MODE_CONFIG

This define controls the availability of the Mcu_MC_ME_FastModeConfig function.

Table 3-54. Define MCU_FAST_MODE_CONFIG Description

Name	MCU_FAST_MODE_CONFIG
Initializer	(STD_OFF)

3.8.1.52 Define MCU_MAX_CLKCONFIGS

Table 3-55. Define MCU_MAX_CLKCONFIGS Description

Name	MCU_MAX_CLKCONFIGS
Initializer	((uint32)[!"num:i($MaxNoOfClkCfgs)"!]U)

3.8.1.53 Define MCU_MAX_MODECONFIGS

Table 3-56. Define MCU_MAX_MODECONFIGS Description

Name	MCU_MAX_MODECONFIGS
Initializer	((uint32)[!"num:i($MaxNoOfModeCfgs)"!]U)

3.8.1.54 Define MCU_MAX_RAMCONFIGS

Maximum number of MCU Clock configurations.

Table 3-57. Define MCU_MAX_RAMCONFIGS Description

Name	MCU_MAX_RAMCONFIGS
Initializer	((uint32)[!"num:i($MaxNoOfRamCfgs)"!]U)

3.8.1.55 Define MCU_PRECOMPILE_SUPPORT

Pre-compile Support.

Table 3-58. Define MCU_PRECOMPILE_SUPPORT Description

Name	MCU_PRECOMPILE_SUPPORT
Initializer	[!IF "(IMPLEMENTATION_CONFIG_VARIANT = 'VariantPreCompile') and (variant:size() <= 1)"!](STD_ON)[!ELSE!](STD_OFF)[!ENDIF!]

3.8.1.56 Define MCU_CONF_PB

Post-Build structures from`Mcu_PBcfg.c`file.

Violates: MISRA 2004 Required Rule 19.4, C macros ...

Table 3-59. Define MCU_CONF_PB Description

Name	MCU_CONF_PB
Initializer	extern CONST(`Mcu_ConfigType`, MCU_CONST)`McuModuleConfiguration_0`;

3.8.1.57 Define MCU_NUMBER_OF_PCS_REGS

max number of prog. clock switch regs.

Table 3-60. Define MCU_NUMBER_OF_PCS_REGS Description

Name	MCU_NUMBER_OF_PCS_REGS
Initializer	((uint8)9U)

3.8.1.58 Define MCU_RAW_RESET_DEFAULT

The MCU module's implementer shall avoid the integration of incompatible files.

Details:

The function Mcu_GetResetRawValue shall return an implementation specific value which does not correspond to a valid value of the reset status register and is not equal to 0 if this function is called prior to calling of the function Mcu_Init, and if supported by the hardware.

Table 3-61. Define MCU_RAW_RESET_DEFAULT Description

Name	MCU_RAW_RESET_DEFAULT
Initializer	((uint32)0xFFFFFFFFUL)

3.8.1.59 Define MCU_VALIDATE_GLOBAL_CALL

Table 3-62. Define MCU_VALIDATE_GLOBAL_CALL Description

Name	MCU_VALIDATE_GLOBAL_CALL
Initializer	(MCU_DEV_ERROR_DETECT)

3.8.2 Enum Reference

Enumeration of all constants supported by the driver are as per AUTOSAR Mcu Driver software specification Version 4.3 Rev0001 .

3.8.2.1 Enumeration Mcu_StatusType

The MCU module's implementer shall avoid the integration of incompatible files.

Details:

This enumerated type contains the Mcu driver's possible states.

Table 3-63. Enumeration Mcu_StatusType Values

Name	Initializer	Description
MCU_UNINIT	0x3U	The Mcu driver is not uninitialized.
MCU_IDLE	0xCU	= 0xE1 The Mcu driver is currently idle.
MCU_BUSY	0xAU	= 0xD2 The Mcu driver is currently busy.

3.8.2.2 Enumeration Mcu_PllStatusType

Type of the return value of the function Mcu_GetPllStatus.

Details:

The type of Mcu_PllStatusType is an enumeration with the following values: MCU_PLL_LOCKED, MCU_PLL_UNLOCKED, MCU_PLL_STATUS_UNDEFINED.

Implements: Mcu_PllStatusType_enumeration

Table 3-64. Enumeration Mcu_PllStatusType Values

Name	Initializer	Description
MCU_PLL_LOCKED	0x33U	PLL is locked.
MCU_PLL_UNLOCKED	0xCCU	PLL is unlocked.
MCU_PLL_STATUS_UNDEFINED	0x5AU	PLL Status is unknown.

3.8.2.3 Enumeration Mcu_RamStateType

Ram State of the microcontroller.

Details:

This is the Ram State data type returned by the function `Mcu_GetRamState()` of the Mcu module.

Table 3-65. Enumeration Mcu_RamStateType Values

Name	Initializer	Description
MCU_RAMSTATE_INVALID	0x0U	RAM content is not valid or unknown (default).
MCU_RAMSTATE_VALID	0x1U	RAM content is valid.

3.8.2.4 Enumeration Mcu_ResetType

The type Mcu_ResetType, represents the different reset that a specified MCU can have.

Details:

The MCU module shall provide at least the values MCU_POWER_ON_RESET and MCU_RESET_UNDEFINED for the enumeration Mcu_ResetType.

Implements: Mcu_ResetType_enumeration

Table 3-66. Enumeration Mcu_ResetType Values

Name	Initializer	Description
MCU_STOP_ACKNOWLEDGE_ERROR_RESET	McuConf_McuResetReasonConf_MCU_STOP_ACKNOWLEDGE_ERROR_RESET	Stop Acknowledge Error reset . RCM_SRS[SACKERR].
MCU_MDM_AP_SYSTEM_RESET	McuConf_McuResetReasonConf_MCU_MDM_AP_SYSTEM_RESET	MDM-AP System Reset Request . RCM_SRS[MDM_AP].
MCU_SW_RESET	McuConf_McuResetReasonConf_MCU_SW_RESET	Software reset . RCM_SRS[SW].
MCU_CORE_LOCKUP_RESET	McuConf_McuResetReasonConf_MCU_CORE_LOCKUP_RESET	Core Lockup reset . RCM_SRS[LOCKUP].

Table continues on the next page...

Table 3-66. Enumeration Mcu_ResetType Values (continued)

Name	Initializer	Description
MCU_JTAG_RESET	McuConf_McuResetReasonConf_MCU_JTAG_RESET	JTAG generated reset . RCM_SRS[JTAG].
MCU_POWER_ON_RESET	McuConf_McuResetReasonConf_MCU_POWER_ON_RESET	Power-on reset. RCM_SRS[POR].
MCU_EXTERNAL_PIN_RESET	McuConf_McuResetReasonConf_MCU_EXTERNAL_PIN_RESET	External Reset Pin. RCM_SRS[PIN].
MCU_WATCHDOG_RESET	McuConf_McuResetReasonConf_MCU_WATCHDOG_RESET	Watchdog reset. RCM_SRS[Watchdog].
MCU_CMU_LOSS_OF_CLOCK_RESET	McuConf_McuResetReasonConf_MCU_CMU_LOSS_OF_CLOCK_RESET	CMU Loss-of-Clock Reset. RCM_SRS[CMU_LOC].
MCU_LOSS_OF_LOCK_RESET	McuConf_McuResetReasonConf_MCU_LOSS_OF_LOCK_RESET	Loss-of-Lock Reset. RCM_SRS[LOL].
MCU_LOSS_OF_CLOCK_RESET	McuConf_McuResetReasonConf_MCU_LOSS_OF_CLOCK_RESET	Loss-of-Clock Reset. RCM_SRS[LOC].
MCU_LOW_OR_HIGH_VOLTAGE_DETECT_RESET	McuConf_McuResetReasonConf_MCU_LOW_OR_HIGH_VOLTAGE_DETECT_RESET	Low-Voltage Detect Reset or High-Voltage Detect Reset. RCM_SRS[LVD].
MCU_NO_RESET_REASON	McuConf_McuResetReasonConf_MCU_NO_RESET_REASON	No reset reason found.
MCU_MULTIPLE_RESET_REASON	McuConf_McuResetReasonConf_MCU_MULTIPLE_RESET_REASON	More than one reset events are logged except "Power on event".
MCU_RESET_UNDEFINED	McuConf_McuResetReasonConf_MCU_RESET_UNDEFINED	Undefined reset source.

3.8.2.5 Enumeration Mcu_SRAMRetenConfigType

Type of parameter value of the function Mcu_SRAMRetentionConfig.

Details:

The type of Mcu_SRAMRetenConfigType is an enumeration with the following values: MCU_SRAML_RETEN, MCU_SRAMU_RETEN, MCU_SRAMLU_RETEN, MCU_NO_SRAMLU_RETEN.

Implements: Mcu_SRAMRetenConfigType_enumeration

Table 3-67. Enumeration Mcu_SRAMRetenConfigType Values

Name	Initializer	Description
MCU_SRAML_RETEN	0x00100000U	Only SRAML will be retained.
MCU_SRAMU_RETEN	0x00200000U	Only SRAML will be retained.
MCU_SRAMLU_RETEN	0x00000000U	Both SRAML and SRAMU will be retained.
MCU_NO_SRAMLU_RETEN	0x00300000U	Both SRAML and SRAMU will not be retained.

3.8.3 Function Reference

Functions of all functions supported by the driver are as per AUTOSAR Mcu Driver software specification Version 4.3 Rev0001 .

3.8.3.1 Function Mcu_Init

MCU driver initialization function.

Details:

This routine initializes the MCU Driver. The intention of this function is to make the configuration setting for power down, clock and Ram sections visible within the MCU Driver.

Return: void.

Implements: Mcu_Init_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Violates: MISRA 2004 Required Rule 19.15, Repeated include file

Prototype: `void Mcu_Init(const Mcu_ConfigType *ConfigPtr);`

Table 3-68. Mcu_Init Arguments

Type	Name	Direction	Description
const`Mcu_ConfigType`*	ConfigPtr	**input**	Pointer to configuration structure.

3.8.3.2 Function Mcu_InitClock

MCU driver clock initialization function.

Details:

This function intializes the PLL and MCU specific clock options. The clock setting is provided from the configuration structure.

Return: Command has or has not been accepted.

Implements: Mcu_InitClock_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `Std_ReturnType Mcu_InitClock(Mcu_ClockType ClockSetting);`

Table 3-69. Mcu_InitClock Arguments

Type	Name	Direction	Description
`Mcu_ClockType`	ClockSetting	**input**	Clock setting ID from config structure to be used.

Table 3-70. Mcu_InitClock Return Values

Name	Description
E_OK	The driver state allowed the execution of the function and the provided parameter was in range.
E_NOT_OK	The driver state did not allowed execution or the parameter was invalid.

3.8.3.3 Function Mcu_DistributePllClock

This function activates the PLL clock to the MCU clock distribution.

Details:

Function completes the PLL configuration and then activates the PLL clock to MCU. If the MCU_NO_PLL is TRUE the Mcu_DistributePllClock has to be disabled. The function will not distribute the PLL clock if the driver state does not allow it, or the PLL is not stable.

Return: Std_ReturnType.

Implements: Mcu_DistributePllClock_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `Std_ReturnType Mcu_DistributePllClock( void);`

3.8.3.4 Function Mcu_InitRamSection

MCU driver initialization of Ram sections.

Details:

Function intializes the ram section selected by RamSection parameter. The section base address, size and value to be written are provided from the configuration structure. The function will write the value specified in the configuration structure indexed by RamSection. After the write it will read back the RAM to verify that the requested value was written.

Return: Command has or has not been accepted.

Implements: Mcu_InitRamSection_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `Std_ReturnType Mcu_InitRamSection(Mcu_RamSectionType RamSection);`

Table 3-71. Mcu_InitRamSection Arguments

Type	Name	Direction	Description
`Mcu_RamSectionType`	RamSection	**input**	Index of ram section from config structure to be initialized.

Table 3-72. Mcu_InitRamSection Return Values

Name	Description
E_OK	Valid parameter, the driver state allowed execution and the RAM check was successful.
E_NOT_OK	Invalid parameter, the driver state did not allowed execution or the RAM check was not successful.

3.8.3.5 Function Mcu_SetMode

This function sets the MCU power mode.

Details:

This function activates MCU power mode from config structure selected by McuMode parameter. If the driver state is invalid or McuMode is not in range the function will skip changing the mcu mode.

Return: void.

Implements: Mcu_SetMode_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `void Mcu_SetMode(Mcu_ModeType McuMode);`

Table 3-73. Mcu_SetMode Arguments

Type	Name	Direction	Description
`Mcu_ModeType`	McuMode	**input**	MCU mode setting ID from config structure to be set.

3.8.3.6 Function Mcu_GetPllStatus

This function returns the lock status of the PLL.

Details:

The user takes care that the PLL is locked by executing Mcu_GetPllStatus. If the MCU_NO_PLL is TRUE the MCU_GetPllStatus has to return MCU_PLL_STATUS_UNDEFINED. It will also return MCU_PLL_STATUS_UNDEFINED if the driver state was invalid

Return: Provides the lock status of the PLL.

Implements: Mcu_GetPllStatus_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `Mcu_PllStatusType Mcu_GetPllStatus(void);`

Table 3-74. Mcu_GetPllStatus Return Values

Name	Description
MCU_PLL_STATUS_UNDEFINED	PLL Status is unknown.
MCU_PLL_LOCKED	PLL is locked.
MCU_PLL_UNLOCKED	PLL is unlocked.

3.8.3.7 Function Mcu_PerformReset

This function performs a microcontroller reset.

Details:

This function performs a microcontroller reset by using the hardware feature of the microcontroller. In case the function returns, the user must reset the platform using an alternate reset mechanism

Return: void.

Implements: Mcu_PerformReset_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `void Mcu_PerformReset(void);`

3.8.3.8 Function Mcu_GetPowerModeState

Prototype: `Mcu_PowerModeStateType Mcu_GetPowerModeState(void);`

3.8.3.9 Function Mcu_GetRamState

This function returns the actual state of the RAM.

Details:

This function returns if the Ram Status is valid after a reset. The report is get from STCU as a result of MBIST (Memory Built-In Self Tests).

Return: Status of the Ram Content.

Implements: Mcu_GetRamState_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `Mcu_RamStateType Mcu_GetRamState(void);`

Table 3-75. Mcu_GetRamState Return Values

Name	Description
MCU_RAMSTATE_INVALID	Ram state is not valid or unknown (default), or the driver state does not allow this call.
MCU_RAMSTATE_VALID	Ram state is valid.

3.8.3.10 Function Mcu_GetResetRawValue

This function returns the Raw Reset value.

Details:

This routine returns the Raw Reset value that is read from the hardware.

Return: Description of the returned value.

Implements: Mcu_GetResetRawValue_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `Mcu_RawResetType Mcu_GetResetRawValue(void);`

Table 3-76. Mcu_GetResetRawValue Return Values

Name	Description
uint32	Code of the Raw reset value. The bits in the returned value are: • [0x2000] - MCU_RAW_RESET_SACKERR • [0x800] - MCU_RAW_RESET_MDM_AP • [0x400] - MCU_RAW_RESET_SW • [0x200] - MCU_RAW_RESET_LOCKUP • [0x100] - MCU_RAW_RESET_JTAG • [0x80] - MCU_RAW_RESET_POR • [0x40] - MCU_RAW_RESET_PIN • [0x20] - MCU_RAW_RESET_WDOG • [0x10] - MCU_RAW_RESET_CMU_LOC • [0x8] - MCU_RAW_RESET_LOL • [0x4] - MCU_RAW_RESET_LOC • [0x2] - MCU_RAW_RESET_LVD

3.8.3.11 Function Mcu_GetResetReason

This function returns the Reset reason.

Details:

This routine returns the Reset reason that is read from the hardware.

Return: Reason of the Reset event.

Implements: Mcu_GetResetReason_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `Mcu_ResetType Mcu_GetResetReason(void);`

Table 3-77. Mcu_GetResetReason Return Values

Name	Description
MCU_STOP_ACKNOWLEDGE_ERROR_RESET	Stop Acknowledge Error reset
MCU_MDM_AP_SYSTEM_RESET	MDM-AP System Reset Request
MCU_SW_RESET	Software reset
MCU_CORE_LOCKUP_RESET	Core Lockup reset
MCU_JTAG_RESET	JTAG reset
MCU_POWER_ON_RESET	Power-on reset
MCU_EXTERNAL_PIN_RESET	External Reset Pin
MCU_WATCHDOG_RESET	Watchdog reset
MCU_LOSS_OF_LOCK_RESET	Loss-of-Lock Reset
MCU_LOSS_OF_CLOCK_RESET	Loss-of-Clock Reset
MCU_LOW_OR_HIGH_VOLTAGE_DETECT_RESET	Low-Voltage Detect Reset or High-Voltage Detect Reset
MCU_NO_RESET_REASON	No reset.
MCU_MULTIPLE_RESET_REASON	Multiple reset reasons.
MCU_RESET_UNDEFINED	Undefined reset.

3.8.3.12 Function Mcu_GetSystemState

Prototype: `uint32 Mcu_GetSystemState(void);`

3.8.3.13 Function Mcu_GetVersionInfo

This function returns the Version Information for the MCU module.

Details:

This function returns the vendor id, module id, major, minor and patch version.

Return: void.

Implements: Mcu_GetVersionInfo_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `void Mcu_GetVersionInfo(Std_VersionInfoType *versioninfo);`

Table 3-78. Mcu_GetVersionInfo Arguments

Type	Name	Direction	Description
Std_VersionInfoType *	versioninfo	**input, output**	A pointer to a variable to store version info.

3.8.3.14 Function Mcu_SRAMRetentionConfig

This function configure SRAM retention.

Details:

This function configure SRAM retention base on both SRAML_RETEN and SRAMU_RETEN bits.

Return: void

Implements: Mcu_SRAMRetentionConfig_Activity

Violates: Violates MISRA 2004 Required Rule 8.10, global declaration of function

Prototype: `void Mcu_SRAMRetentionConfig(Mcu_SRAMRetenConfigType eSRAMRetenConfig);`

Table 3-79. Mcu_SRAMRetenConfigType Values

Name	Description
MCU_SRAML_RETEN	Only SRAML will be retained.
MCU_SRAMU_RETEN	Only SRAMU will be retained.
MCU_SRAMLU_RETEN	Both SRAML and SRAMU will be retained.
MCU_NO_SRAMLU_RETEN	Both SRAML and SRAMU will not be retained.

3.8.3.15 Function Mcu_DisableCmu

Prototype: `void Mcu_DisableCmu(uint8 u8IndexCmu);`

3.8.4 Structs Reference

Data structures supported by the driver are as per AUTOSAR Mcu Driver software specification Version 4.3 Rev0001 .

3.8.4.1 Structure Mcu_ConfigType

Initialization data for the MCU driver.

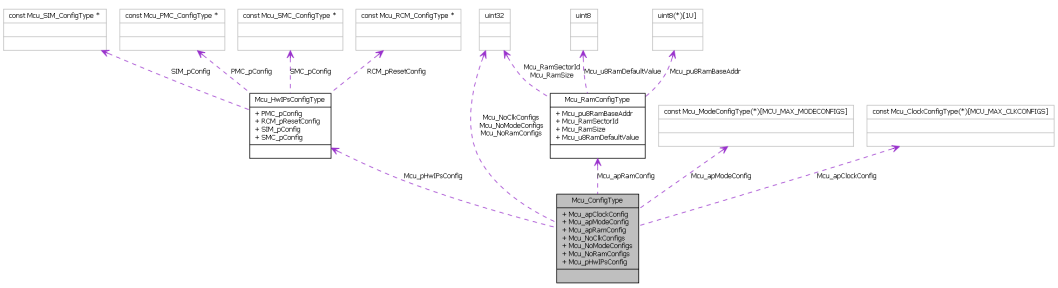


Figure 3-1. Struct Mcu_ConfigType

Details:

A pointer to such a structure is provided to the MCU initialization routines for configuration.

Implements: Mcu_ConfigType_struct

Declaration:

```
        typedef struct
        {
            const Mcu_DemConfigType* Mcu_pDemConfig,
            Mcu_RamSectionType Mcu_NoRamConfigs,
            Mcu_ModeType Mcu_NoModeConfigs,
            Mcu_ClockType Mcu_NoClkConfigs,
            const Mcu_RamConfigType (*Mcu_apRamConfig)[MCU_MAX_RAMCONFIGS],
            const Mcu_ModeConfigType (*Mcu_apModeConfig)[MCU_MAX_MODECONFIGS],
            const Mcu_LPU_LowPowerModeConfigType (*Mcu_apLowPowerModeConfig)
[MCU_MAX_MODECONFIGS],
            const Mcu_ClockConfigType (*Mcu_apClockConfig)[MCU_MAX_CLKCONFIGS],
            const Mcu_HwIPsConfigType* Mcu_pHwIPsConfig;
```

```
} Mcu_ConfigType;
```

Table 3-80. Structure Mcu_ConfigType member description

Member	Description
Mcu_pDemConfig	DEM error reporting configuration
Mcu_NoRamConfigs	Total number of RAM sections.
Mcu_NoModeConfigs	Total number of MCU modes.
Mcu_NoClkConfigs	Total number of MCU clock configurations.
Mcu_apRamConfig	RAM data configuration.
Mcu_apModeConfig	Power Modes data configuration.
Mcu_apLowPowerModeConfig	Low Power Modes data configuration.
Mcu_apClockConfig	Clock data configuration.
Mcu_pHwIPsConfig	IPs data generic configuration.

3.8.4.2 Structure Mcu_HwIPsConfigType

Mcu driver configuration structure.

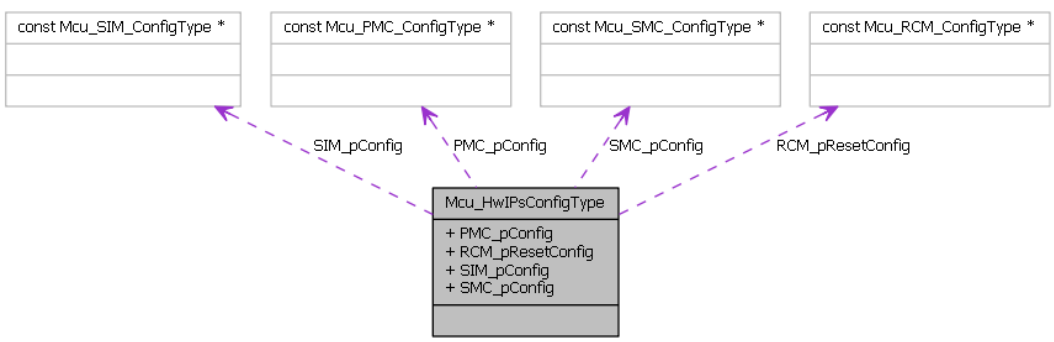


Figure 3-2. Struct Mcu_HwIPsConfigType

Details:

Configuration for RCM reset configuration module. Configuration for power management. Configuration for System integration module (SIM). Used by "Mcu_ConfigType" structure.

Declaration:

```
typedef struct
{
    const Mcu_PMC_ConfigType * PMC_pConfig,
    const Mcu_RCM_ConfigType * RCM_pResetConfig,
    const Mcu_SIM_ConfigType * SIM_pConfig,
    const Mcu_SMC_ConfigType * SMC_pConfig
```

```
} Mcu_HwIPsConfigType;
```

Table 3-81. Structure Mcu_HwIPsConfigType member description

Member	Description
PMC_pConfig	Configuration for MC_PCU/PMU (Power Management Unit) hardware IP, part of PCU.
RCM_pResetConfig	Configuration for RCM (Reset Control Module) hardware IP.
SIM_pConfig	Configuration for SIM (System Integration Module) hardware IP.
SMC_pConfig	Configuration for SMC hardware IP.

3.8.4.3 Structure Mcu_RamConfigType

Definition of a RAM section within the configuration structure. The definitions for each RAM section within the structure `Mcu_ConfigType` shall contain:

- RAM section base address.
- Section size.
- Data pre-setting to be initialized.

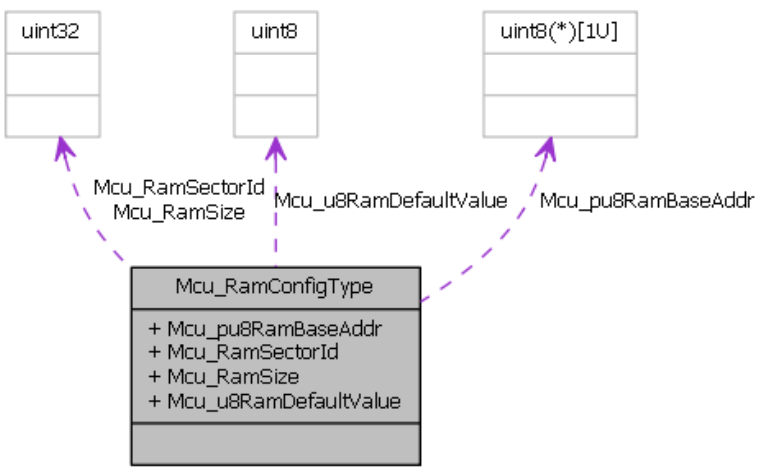


Figure 3-3. Struct Mcu_RamConfigType

Declaration:

```
typedef struct
{
    uint8 * Mcu_pu8RamBaseAddr,
    Mcu_RamSectionType Mcu_RamSectorId,
    Mcu_RamSizeType Mcu_RamSize,
    uint8 Mcu_u8RamDefaultValue
} Mcu_RamConfigType;
```

Table 3-82. Structure Mcu_RamConfigType member description

Member	Description
Mcu_pu8RamBaseAddr	RAM section base address.
Mcu_RamSectorId	The ID for Ram Sector configuration.
Mcu_RamSize	RAM section size.
Mcu_u8RamDefaultValue	RAM default value for initialization.

3.8.5 Types Reference

Types supported by the driver are as per AUTOSAR Mcu Driver software specification Version 4.3 Rev0001 .

3.8.5.1 Typedef Mcu_ClockType

Defines the identification (ID) for clock setting configured via the configuration structure.

Details:

The type shall be uint8, uint16 or uint32, depending on uC platform.

Implements: Mcu_ClockType_typedef

Type: uint32

3.8.5.2 Typedef Mcu_ModeType

The Mcu_ModeType specifies the identification (ID) for a MCU mode, configured via configuration structure.

Details:

The type shall be uint8, uint16 or uint32.

Implements: Mcu_ModeType_typedef

Type: uint32

3.8.5.3 Typedef Mcu_RamSectionType

The Mcu_RamSectionType specifies the identification (ID) for a RAM section, configured via the configuration structure. The type shall be uint8, uint16 or uint32, based on best performance.

Implements: Mcu_RamSectionType_typedef

Type: uint32

3.8.5.4 Typedef Mcu_RawResetType

The type Mcu_RawResetType specifies the reset reason in raw register format, read from a reset status register.

Details:

The type shall be uint8, uint16 or uint32 based on best performance.

Implements: Mcu_RawResetType_typedef Destructive and Functional Reset Events Log.

Type: uint32

3.9 Symbolic Names Disclaimer

All containers having the symbolic name tag set as true in the Autosar schema will generate defines like:

#define <Container_Short_Name> <Container_ID>

For this reason it is forbidden to duplicate the name of such containers across the MCAL configuration, or to use names that may trigger other compile issues (e.g. match existing #ifdefs arguments).

Chapter 4
Tresos Configuration Plug-in

This chapter describes the Tresos configuration plug-in for the Mcu Driver. The most of the parameters are described below.

4.1 Configuration elements of Mcu

Included forms :

- Form IMPLEMENTATION_CONFIG_VARIANT
- Form McuGeneralConfiguration
- Form McuDebugConfiguration
- Form McuPublishedInformation
- Form McuModuleConfiguration
- Form CommonPublishedInformation

4.2 Form IMPLEMENTATION_CONFIG_VARIANT

VariantPreCompile: Only precompile time configuration parameters. Only one set of parameters.

VariantPostBuild: Mix of precompile and postbuild time configuration parameters. Only one set of parameters.

VariantPostBuildSelectable: Mix of precompile and postbuild time configuration parameters. More sets of parameters.

Figure 4-1. Tresos Plugin snapshot for IMPLEMENTATION_CONFIG_VARIANT form.

Table 4-1. Attribute IMPLEMENTATION_CONFIG_VARIANT detailed description

Property	Value
Label	Config Variant
Type	ENUMERATION
Default	VariantPostBuild
Range	VariantPostBuild VariantPreCompile

4.3 Form McuGeneralConfiguration

This container contains the general configuration for the MCU driver.

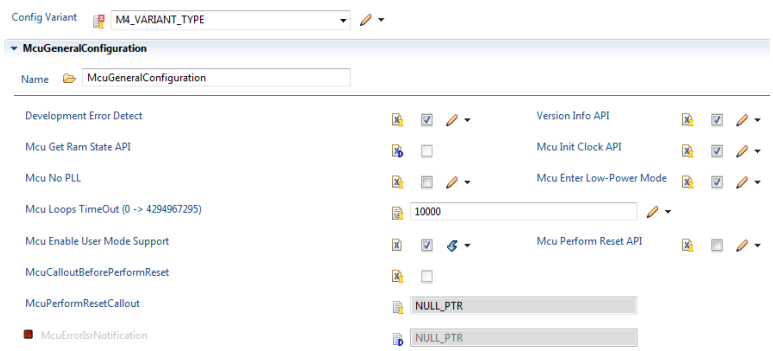


Figure 4-2. Tresos Plugin snapshot for McuGeneralConfiguration form.

4.3.1 McuDevErrorDetect (McuGeneralConfiguration)

Pre-processor switch for enabling the default error detection and reporting to the DET. The switch McuDevErrorDetect shall switch the Default Error Tracer (Det) detection and notification ON or OFF. The detection of default errors is configurable (ON/OFF) at precompile time. #define MCU_DEV_ERROR_DETECT (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-2. Attribute McuDevErrorDetect (McuGeneralConfiguration) detailed description

Property	Value
Label	Default Error Detect
Type	BOOLEAN
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	false

4.3.2 McuVersionInfoApi (McuGeneralConfiguration)

Pre-processor switch to enable/disable the API to read out the modules version information. #define MCU_VERSION_INFO_API (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-3. Attribute McuVersionInfoApi (McuGeneralConfiguration) detailed description

Property	Value
Label	Version Info API
Type	BOOLEAN
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	false

4.3.3 McuGetRamStateApi (McuGeneralConfiguration)

Pre-processor switch to enable/disable the API Mcu_GetRamState. #define MCU_GET_RAM_STATE_API (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-4. Attribute McuGetRamStateApi (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu Get Ram State API
Type	BOOLEAN
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	false

4.3.4 McuInitClock (McuGeneralConfiguration)

If this parameter is set to FALSE, the clock initialization has to be disabled from the MCU driver. This concept applies when there are some write once clock registers and a bootloader is present. If this parameter is set to TRUE, the MCU driver is responsible of the clock initialization #define MCU_INIT_CLOCK (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-5. Attribute McuInitClock (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu Init Clock API
Type	BOOLEAN
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	true

4.3.5 McuNoPll (McuGeneralConfiguration)

This parameter shall be set True, if the H/W does not have a PLL or the PLL circuitry is enabled after the power on without S/W intervention. In this case MCU_DistributePllClock has to be disabled and MCU_GetPllStatus has to return MCU_PLL_STATUS_UNDEFINED. Otherwise this parameters has to be set False. #define MCU_NO_PLL (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-6. Attribute McuNoPll (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu No PLL
Type	BOOLEAN
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	true

4.3.6 McuEnterLowPowerMode (McuGeneralConfiguration)

If this parameter has been configured to 'TRUE', the function 'Mcu_SetMode()' shall not be impacted and behave as specified. If this parameter has been configured to 'FALSE', the function 'Mcu_SetMode()' shall not perform the transition to any low power modes as are 'STOP' or 'HALT' or any other mode, where the core stops execution. #define MCU_ENTER_LOW_POWER_MODE (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-7. Attribute McuEnterLowPowerMode (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu Enter Low-Power Mode

Table continues on the next page...

Table 4-7. Attribute McuEnterLowPowerMode (McuGeneralConfiguration) detailed description (continued)

Property	Value
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.3.7 McuTimeout (McuGeneralConfiguration)

This parameter represents the maximum number of loops for blocking functionality. The maximum time needed for a MC_ME transition from DRUN to DRUN with keeping PLL running is 3 ms.

Please take this into consideration when choosing the value for this parameter.

Note: Implementation Specific Parameter.

Table 4-8. Attribute McuTimeout (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu Loops TimeOut
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	50000
Invalid	Range >=0 <=4294967295

4.3.8 McuEnableUserModeSupport (McuGeneralConfiguration)

When this parameter is enabled, the MDL module will adapt to run from User Mode, with the following measures:

a) using 'call trusted function' stubs for all internal function calls that access registers requiring supervisor mode.

b) other module specific measures

for more information, please see chapter 5.7 User Mode Support in IM

Note: Implementation Specific Parameter.

Table 4-9. Attribute McuEnableUserModeSupport (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu Enable User Mode Support
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.3.9 McuPerformResetApi (McuGeneralConfiguration)

Pre-processor switch to enable/disable the use the Mcu_PerformReset() API. OFF - Mcu_PerformReset() API is not used. ON - Mcu_PerformReset() API is used. #define MCU_PERFORM_RESET_API (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-10. Attribute McuPerformResetApi (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu Perform Reset API
Type	BOOLEAN
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	false

4.3.10 McuCalloutBeforePerformReset (McuGeneralConfiguration)

Check this if you want a callout function, called by MCU right before Mcu_PerformReset(). This parameter is available for configuration only if "McuPerformResetApi" is ON. #define MCU_RESET_CALLOUT_USED (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file. Note: Implementation Specific Parameter.

Table 4-11. Attribute McuCalloutBeforePerformReset (McuGeneralConfiguration) detailed description

Property	Value
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.3.11 McuPerformResetCallout (McuGeneralConfiguration)

Function name of callout. The field is editable only if "McuCalloutBeforePerformReset" is ON. Note: Implementation Specific Parameter.

Table 4-12. Attribute McuPerformResetCallout (McuGeneralConfiguration) detailed description

Property	Value
Type	FUNCTION-NAME
Origin	Custom
Symbolic Name	false
Default	NULL_PTR

4.3.12 McuCmuNotification (McuGeneralConfiguration)

Function pointer to callback function.

Note: Implementation Specific Parameter.

Table 4-13. Attribute McuCmuNotification (McuGeneralConfiguration) detailed description

Property	Value
Type	FUNCTION-NAME
Origin	Custom
Symbolic Name	false
Default	NULL_PTR

4.3.13 McuErrorIsrNotification (McuGeneralConfiguration)

Function name of callout. This function will be called by the error ISR.

Note: Implementation Specific Parameter.

Table 4-14. Attribute McuErrorIsrNotification (McuGeneralConfiguration) detailed description

Property	Value
Type	FUNCTION-NAME
Origin	Custom
Symbolic Name	false
Default	NULL_PTR

4.3.14 McuDisableSimInit (McuGeneralConfiguration)

If this parameter is set to TRUE, the System Integration Module (SIM) initialization has to be disabled in the MCU driver.

If this parameter is set to FALSE, the MCU driver is responsible for the System Integration Module (SIM) initialization.

#define MCU_DISABLE_SIM_INIT (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-15. Attribute McuDisableSimInit (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu Disable System Integration Module Initialization
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.3.15 McuDisableRcmInit (McuGeneralConfiguration)

If this parameter is set to TRUE, the Reset Control Module (RMC) initialization has to be disabled in the MCU driver.

If this parameter is set to FALSE, the MCU driver is responsible for the Reset Control Module (RMC) initialization.

#define MCU_DISABLE_RMC_INIT (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-16. Attribute McuDisableRcmInit (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu Disable Reset Control Module Initialization
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.3.16 McuDisablePmcInit (McuGeneralConfiguration)

If this parameter is set to TRUE, the Power Management Controller (PMC) initialization has to be disabled in the MCU driver.

If this parameter is set to FALSE, the MCU driver is responsible for the Power Management Controller (PMC) initialization.

#define MCU_DISABLE_PMC_INIT (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-17. Attribute McuDisablePmcInit (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu Disable Power Management Controller Initialization
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.3.17 McuDisableSmcInit (McuGeneralConfiguration)

If this parameter is set to TRUE, the System Mode Controller (SMC) initialization has to be disabled in the MCU driver.

If this parameter is set to FALSE, the MCU driver is responsible for the System Mode Controller (SMC) initialization.

#define MCU_DISABLE_SMC_INIT (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Table 4-18. Attribute McuDisableSmcInit (McuGeneralConfiguration) detailed description

Property	Value
Label	Mcu Disable System Mode Controller Initialization
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.4 Form McuDebugConfiguration

This container contains option for non-ASR APIs used for debug or extra-implementation. Note: Implementation Specific Parameter.

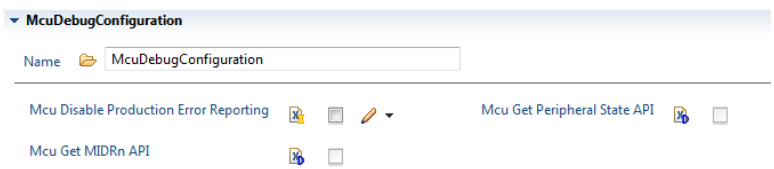


Figure 4-3. Tresos Plugin snapshot for McuDebugConfiguration form.

4.4.1 McuDisableDemReportErrorStatus (McuDebugConfiguration)

Enable/Disable the API for reporting the Dem Error.

Note: Implementation Specific Parameter.

Table 4-19. Attribute McuDisableDemReportErrorStatus (McuDebugConfiguration) detailed description

Property	Value
Label	Mcu Disable Production Error Reporting
Type	BOOLEAN

Table continues on the next page...

Table 4-19. Attribute McuDisableDemReportErrorStatus (McuDebugConfiguration) detailed description (continued)

Property	Value
Origin	Custom
Symbolic Name	false
Default	false

4.4.2 McuGetPeriphStateApi (McuDebugConfiguration)

Enable/Disable the API for checking peripheral states in the current mode from PCC configuration: Mcu_GetPeripheral_State(). E_OK means Peripheral with ID as parameter is clocked. E_NOT_OK means Peripheral with ID as parameter is not clocked. Note: Implementation Specific Parameter.

Table 4-20. Attribute McuGetPeriphStateApi (McuDebugConfiguration) detailed description

Property	Value
Label	Mcu Get Peripheral State API
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.4.3 McuGetMidrStructureApi (McuDebugConfiguration)

Enable/Disable the API for Mcu_GetMidrStructure().

Get information from SIUL2 MIDRn registers.

Note: Implementation Specific Parameter.

Table 4-21. Attribute McuGetMidrStructureApi (McuDebugConfiguration) detailed description

Property	Value
Label	Mcu Get MIDRn API
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.4.4 McuDisableCmuApi (McuDebugConfiguration)

Enable/Disable the API for disabling the clock monitoring unit.

Note: Implementation Specific Parameter.

Table 4-22. Attribute McuDisableCmuApi (McuDebugConfiguration) detailed description

Property	Value
Label	Mcu Disable CMU API
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.4.5 McuEnablePeripheralCMU (McuDebugConfiguration)

Enable/Disable Peripheral CMU for S32K11X

Note: Implementation Specific Parameter.

Table 4-23. Attribute McuEnablePeripheralCMU (McuDebugConfiguration) detailed description

Property	Value
Label	Mcu Enable Peripheral CMU
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.4.6 McuSRAMRetentionConfigApi (McuDebugConfiguration)

Enable/Disable the API for SRAM retention configuration.

Note: Implementation Specific Parameter.

Table 4-24. Attribute McuSRAMRetentionConfigApi (McuDebugConfiguration) detailed description

Property	Value
Label	Mcu SRAM Retention Config API
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.5 Form McuPublishedInformation

Container holding all MCU specific published information parameters.

Included forms :

- Form McuResetReasonConf

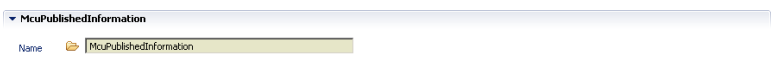
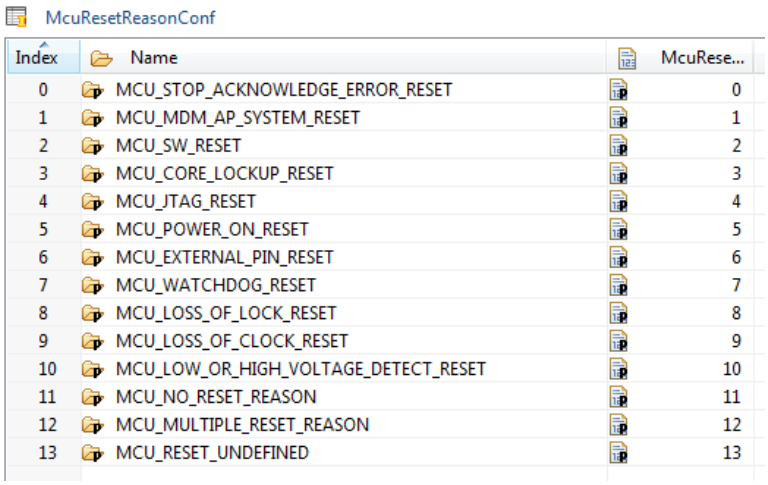


Figure 4-4. Tresos Plugin snapshot for McuPublishedInformation form.

4.5.1 Form McuResetReasonConf

This container contains the configuration for the different type of reset reason that can be retrieved from Mcu_GetResetReason Api.

Is included by form : Form McuPublishedInformation


McuResetReasonConf

Index	Name	McuRese...
0	MCU_STOP_ACKNOWLEDGE_ERROR_RESET	0
1	MCU_MDM_AP_SYSTEM_RESET	1
2	MCU_SW_RESET	2
3	MCU_CORE_LOCKUP_RESET	3
4	MCU_JTAG_RESET	4
5	MCU_POWER_ON_RESET	5
6	MCU_EXTERNAL_PIN_RESET	6
7	MCU_WATCHDOG_RESET	7
8	MCU_LOSS_OF_LOCK_RESET	8
9	MCU_LOSS_OF_CLOCK_RESET	9
10	MCU_LOW_OR_HIGH_VOLTAGE_DETECT_RESET	10
11	MCU_NO_RESET_REASON	11
12	MCU_MULTIPLE_RESET_REASON	12
13	MCU_RESET_UNDEFINED	13

Figure 4-5. Tresos Plugin snapshot for McuResetReasonConf form.

4.5.1.1 McuResetReason (McuResetReasonConf)

The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module.

Table 4-25. Attribute McuResetReason (McuResetReasonConf) detailed description

Property	Value
Type	INTEGER_LABEL
Origin	AUTOSAR_ECUC
Symbolic Name	true
Default	0
Invalid	Range <=255 >=0

4.6 Form McuModuleConfiguration

This container contains the configuration for the MCU driver.

Included forms :

- Form McuDemEventParameterRefs
- Form McuInterruptEvents
- Form McuResetConfig
- Form McuPowerControl

- Form McuClockSettingConfig
- Form McuModeSettingConf
- Form McuRamSectorSettingConf

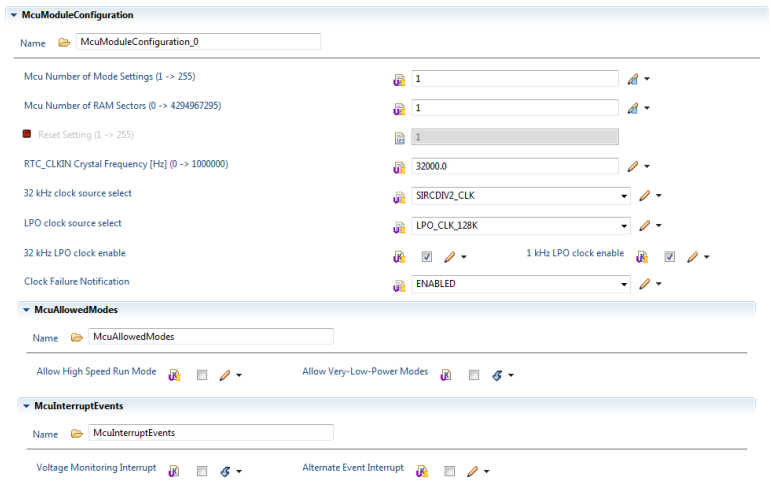


Figure 4-6. Tresos Plugin snapshot for McuModuleConfiguration form.

4.6.1 McuNumberOfMcuModes (McuModuleConfiguration)

This parameter shall represent the number of Modes available for the MCU (from "McuModeSettingConf" list). CalculationFormula = Number of configured "McuModeSettingConf". This parameter is not used.

Table 4-26. Attribute McuNumberOfMcuModes (McuModuleConfiguration) detailed description

Property	Value
Label	Mcu Number of Mode Settings
Type	INTEGER
Origin	AUTOSAR_ECUC
Symbolic Name	false

4.6.2 McuRamSectors (McuModuleConfiguration)

This parameter shall represent the number of RAM sectors available for the MCU (from "McuRamSectorSettingConf" list). CalculationFormula = Number of configured "McuRamSectorSettingConf". This parameter is not used.

Table 4-27. Attribute McuRamSectors (McuModuleConfiguration) detailed description

Property	Value
Label	Mcu Number of RAM Sectors
Type	INTEGER
Origin	AUTOSAR_ECUC
Symbolic Name	false

4.6.3 McuResetSetting (McuModuleConfiguration)

This parameters applies to the function Mcu_PerformReset(), which performs a microcontroller reset using the hardware feature of the microcontroller. Note: This parameter is not used by the current Implementation. Software Reset occurs when Mcu_PerformReset() function is called. This parameter is not used.

Table 4-28. Attribute McuResetSetting (McuModuleConfiguration) detailed description

Property	Value
Label	Reset Setting
Type	INTEGER
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	1
Enable	false

4.6.4 McuRTCCLKINFrequencyHz (McuModuleConfiguration)

Crystal Frequency or External Reference Frequency [Hz]. Note: Implementation Specific Parameter.

Table 4-29. Attribute McuRTCCLKINFrequencyHz (McuModuleConfiguration) detailed description

Property	Value
Label	McuRTCCLKINFrequencyHz
Type	FLOAT
Origin	Custom
Symbolic Name	false
Default	3200000
Invalid	Range

Table 4-29. Attribute McuRTCCLKINFrequencyHz (McuModuleConfiguration) detailed description

Property	Value
	<=1000000 >=0

4.6.5 McuClk32KSelect (McuModuleConfiguration)

This is a write-once parameter SIM_LPOCLKS[CLK32KSEL] - 32 kHz clock source select. Selects 32 kHz clock source for peripherals.

0 - SIRCDIV2_CLK

1 - 32 kHz LPO clock

2 - 32 kHz RTC_CLKIN clock

3 - FIRCDIV2_CLK

Note: Implementation Specific Parameter.

4.6.6 McuLPOClockSelect (McuModuleConfiguration)

This is a write-once parameter SIM_LPOCLKS[LPOCLKSEL] - LPO clock source select Selects LPO clock source for peripherals

0 - 128 kHz LPO clock

1 - No clock

2 - 32 kHz LPO clock which is divided by the 128 kHz LPO clock

3 - 1 kHz LPO clock which is divided by the 128 kHz LPO clock

Note: Implementation Specific Parameter.

4.6.6 McuClockSrcFailureNotification

Enables/Disables clock failure notification. In case this feature is not supported by HW the setting should be disabled. CMU0, CMU1, CMU2, CMU3, CMU4, CMU5, CMU6, CMU7, CMU8 settings are disabled if McuClockSrcFailureNotification=DISABLED.

4.6.7 Form McuAllowedModes

Configures SMC_PMPROT register. The PMPROT register can be written only once after any system reset.

If the MCU is configured for a disallowed or reserved power mode, the MCU remains in its current power mode. For example, if the MCU is in normal RUN mode and AVLP is 0, an attempt to enter VLPR mode using PMCTRL[RUNM] is blocked and PMCTRL[RUNM] remains 00b, indicating the MCU is still in Normal Run mode.

Note: Implementation specific Container.

Is included by form : Form McuModuleConfiguration

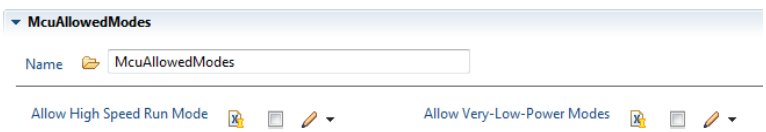


Figure 4-7. Tresos Plugin snapshot for McuAllowedModes form.

4.6.7.1 McuAllowHighSpeedRunMode (McuAllowedModes)

This is a write-once parameter

SMC_PMPROT[AHSRUN] - Allow High Speed Run mode

Provided the appropriate control bits are set up in PMCTRL, this write-once field allows the MCU to enter High Speed Run mode (HSRUN).

0 - HSRUN is not allowed

1 - HSRUN is allowed

Note: Implementation Specific Parameter.

Table 4-30. Attribute McuAllowHighSpeedRunMode (McuAllowedModes) detailed description

Property	Value
Label	Allow High Speed Run Mode
Type	BOOLEAN

Table continues on the next page...

Table 4-30. Attribute McuAllowHighSpeedRunMode (McuAllowedModes) detailed description (continued)

Property	Value
Origin	Custom
Symbolic Name	false
Default	false

4.6.7.2 McuAllowVeryLowPowerModes (McuAllowedModes)

This is a write-once parameter

Provided the appropriate control bits are set up in PMCTRL, this write-once field allows the MCU to enter any very-low-power mode (VLPR, and VLPS).

0 - VLPR and VLPS are not allowed.

1 - VLPR and VLPS are allowed.

Note: Implementation Specific Parameter.

Table 4-31. Attribute McuAllowVeryLowPowerModes (McuAllowedModes) detailed description

Property	Value
Label	Allow Very-Low-Power Modes
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8 Form McuClockSettingConfig

This container contains the configuration for the Clock settings of the MCU.

Is included by form : Form McuModuleConfiguration

Included forms :

- Form McuRunClockConfig
- Form McuVlprClockConfig

- Form McuHsrunClockConfig
- Form McuSystemOSCClockConfig
- Form McuSIRCClockConfig
- Form McuFIRCClockConfig
- Form McuSystemPll
- Form McuSIMClockConfig
- Form McuClkMonitor_0
- Form McuClkMonitor_1
- Form McuPeripheralClockConfig
- Form McuClockReferencePoint

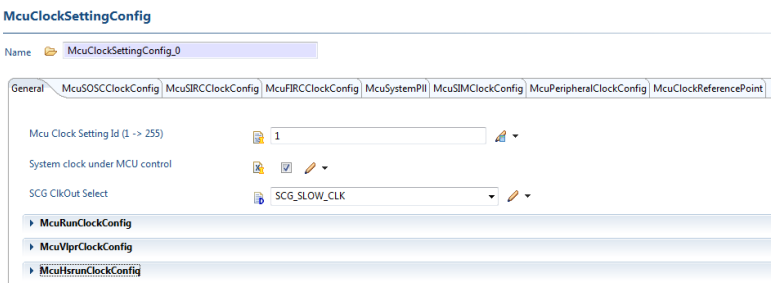


Figure 4-8. Tresos Plugin snapshot for McuClockSettingConfig form.

4.6.8.1 McuClockSettingId (McuClockSettingConfig)

The Id of this McuClockSettingConfig to be used as argument for the API call Mcu_InitClock().

Table 4-32. Attribute McuClockSettingId (McuClockSettingConfig) detailed description

Property	Value
Label	Mcu Clock Setting Id
Type	INTEGER
Origin	AUTOSAR_ECUC
Symbolic Name	true
Invalid	Range <=255 >=0

4.6.8.2 McuSysClockUnderMcuControl (McuClockSettingConfig)

0 - System clock tree is NOT under mcu control.

1 - System clock is under mcu control.

If this is set to false, the MCU code will not configure the SIU_SYSDIV register when Mcu_InitClock is called

Note: Implementation Specific Parameter.

Table 4-33. Attribute McuSysClockUnderMcuControl (McuClockSettingConfig) detailed description

Property	Value
Label	System clock under MCU control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.3 McuScgClkOutSelect (McuClockSettingConfig)

SCG_CLKOUTCNFG[CLKOUTSEL] - SCG Clkout Select. This register controls which SCG clock source is selected to be ported out to the CLKOUT pin.

0 - SCG SLOW Clock (FLASH_CLK Clock).

1 - System OSC

2 - Slow IRC

3 - Fast IRC

6 - System PLL

Note: Implementation Specific Parameter.

Table 4-34. Attribute McuScgClkOutSelect (McuClockSettingConfig) detailed description

Property	Value
Label	SCG ClkOut Select
Type	ENUMERATION
Origin	Custom

Table continues on the next page...

Table 4-34. Attribute McuScgClkOutSelect (McuClockSettingConfig) detailed description (continued)

Property	Value
Symbolic Name	false
Default	SCG_SLOW_CLK
Range	SCG_SLOW_CLK SOSC SIRC FIRC SPLL

4.6.8.4 Form McuRunClockConfig

This container configures the system clock source and the system clock dividers for the core, platform, external and bus clock domains when in Run mode only.

Note: Implementation specific Container.

Is included by form : Form McuClockSettingConfig

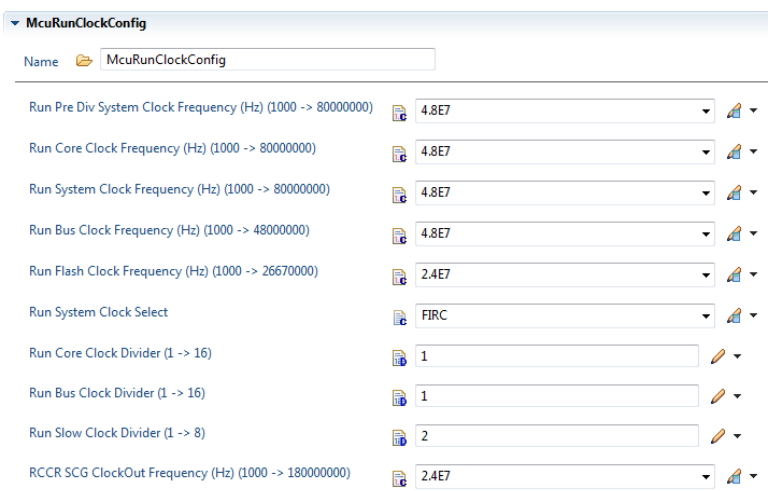


Figure 4-9. Tresos Plugin snapshot for McuRunClockConfig form.

4.6.8.4.1 McuPreDivSystemClockFrequency (McuRunClockConfig)

Run Core clock - Pre Divide System Clock Frequency.

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Note: PREDIV_SYS_CLK is only available in S32K148.

Note: Implementation Specific Parameter.

Table 4-35. Attribute McuPreDivSystemClockFrequency (McuRunClockConfig) detailed description

Property	Value
Label	Run Pre Div System Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.4.2 McuCoreClockFrequency (McuRunClockConfig)

Run Core clock - Clocks the ARM core, divided by DIVCORE bits inside SCG

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-36. Attribute McuCoreClockFrequency (McuRunClockConfig) detailed description

Property	Value
Label	Run Core Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.4.3 McuSystemClockFrequency (McuRunClockConfig)

Run System clock - Clocks the Crossbar, NVIC, Flash controller, FTM and PDB,etc.

RUN_SYS_CLK can run up to CORE_CLK.

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-37. Attribute McuSystemClockFrequency (McuRunClockConfig) detailed description

Property	Value
Label	Run System Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.4.4 McuBusClockFrequency (McuRunClockConfig)

Run Bus clock - BUS_CLK Clocks the Peripherals, divided by DIVBUS bits inside SCG

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-38. Attribute McuBusClockFrequency (McuRunClockConfig) detailed description

Property	Value
Label	Run Bus Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.4.5 McuFlashClockFrequency (McuRunClockConfig)

Run Flash clock - Clocks the flash module, divided by DIVSLOW bits inside SCG

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-39. Attribute McuFlashClockFrequency (McuRunClockConfig) detailed description

Property	Value
Label	Run Flash Clock Frequency (Hz)
Type	FLOAT

Table continues on the next page...

Table 4-39. Attribute McuFlashClockFrequency (McuRunClockConfig) detailed description (continued)

Property	Value
Origin	Custom
Symbolic Name	false

4.6.8.4.6 McuSystemClockSwitch (McuRunClockConfig)

Run System Clock Select. Configure the SCG_RCCR[SCS] register field. The system clock is either:

- System OSC (SCG_RCCR[SCS]=1)

- Slow IRC (SCG_RCCR[SCS]=2)

- Fast IRC (SCG_RCCR[SCS]=3)

- System PLL (SCG_RCCR[SCS]=6)

Value extracted from Resource: MCU.RunSystemClkSource.List

Note: Implementation Specific Parameter.

Table 4-40. Attribute McuSystemClockSwitch (McuRunClockConfig) detailed description

Property	Value
Label	Run System Clock Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false

4.6.8.4.7 McuCoreClockDivider (McuRunClockConfig)

Configures the SCG_RCCR[DIVCORE] bitfield This parameter represents the core clock divider. Note: implementation specific parameter.

Table 4-41. Attribute McuCoreClockDivider (McuRunClockConfig) detailed description

Property	Value
Label	Run Core Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	XPath Range <=16 >=1

4.6.8.4.8 McuBusClockDivider (McuRunClockConfig)

Configures the SCG_RCCR[DIVBUS] bitfield This parameter represents the bus clock divider. Note: implementation specific parameter.

Table 4-42. Attribute McuBusClockDivider (McuRunClockConfig) detailed description

Property	Value
Label	Run Bus Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range <=16 >=1

4.6.8.4.9 McuSlowClockDivider (McuRunClockConfig)

Configures the SCG_RCCR[DIVSLOW] bitfield.

This parameter represents the flash clock divider.

Note: Configuring DIVSLOW to lower frequencies than supported flash frequencies mentioned in datasheet (fFLASH) adds wait states and no power saving. It is recommended to configure DIVSLOW as close to fFLASH.

Table 4-43. Attribute McuSlowClockDivider (McuRunClockConfig) detailed description

Property	Value
Label	Run Slow Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	2
Invalid	Range <=8 >=1

4.6.8.4.10 McuScgClkOutFrequency (McuRunClockConfig)

This is frequency of SCG clockout. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-44. Attribute McuScgClkOutFrequency (McuRunClockConfig) detailed description

Property	Value
Label	RCCR SCG ClockOut Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.5 Form McuVlprClockConfig

Selects the clock source generating the system clock in VLPR mode.

The clock dividers cannot be changed while in VLPR mode. They must be programmed prior to entering VLPR mode to guarantee

- the core/system and bus clocks are less than or equal to 4 MHz

- the flash memory clock is less than or equal to 1 MHz.

Note: Implementation specific Container.

Is included by form : Form McuClockSettingConfig

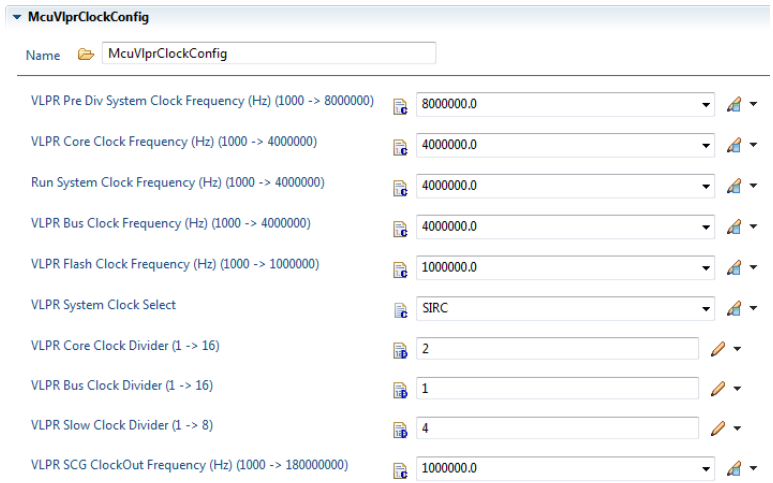


Figure 4-10. Tresos Plugin snapshot for McuVlprClockConfig form.

4.6.8.5.1 McuPreDivSystemClockFrequency (McuVlprClockConfig)

VLPR Core clock - Pre Divide System Clock Frequency.

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Note: PREDIV_SYS_CLK is only available in S32K148.

Note: Implementation Specific Parameter.

Table 4-45. Attribute McuPreDivSystemClockFrequency (McuVlprClockConfig) detailed description

Property	Value
Label	VLPR Pre Div System Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.5.2 McuCoreClockFrequency (McuVlprClockConfig)

VLPR Core clock - Clocks the ARM core, divided by DIVCORE bits inside SCG

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-46. Attribute McuCoreClockFrequency (McuVlprClockConfig) detailed description

Property	Value
Label	VLPR Core Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false
Invalid	Range <=4000000 >=1000

4.6.8.5.3 McuSystemClockFrequency (McuVlprClockConfig)

VLPR System clock - Clocks the Crossbar, NVIC, Flash controller, FTM and PDB,etc.

VLPR_SYS_CLK can run up to CORE_CLK.

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-47. Attribute McuSystemClockFrequency (McuVlprClockConfig) detailed description

Property	Value
Label	Run System Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false
Invalid	Range <=4000000 >=1000

4.6.8.5.4 McuBusClockFrequency (McuVlprClockConfig)

VLPR Bus clock - BUS_CLK Clocks the Peripherals, divided by DIVBUS bits inside SCG

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-48. Attribute McuBusClockFrequency (McuVlprClockConfig) detailed description

Property	Value
Label	VLPR Bus Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false
Invalid	Range <=4000000 >=1000

4.6.8.5.5 McuFlashClockFrequency (McuVlprClockConfig)

VLPR Flash clock - Clocks the flash module, divided by DIVSLOW bits inside SCG

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-49. Attribute McuFlashClockFrequency (McuVlprClockConfig) detailed description

Property	Value
Label	VLPR Flash Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false
Invalid	XPath Range <=1000000 >=1000

4.6.8.5.6 McuSystemClockSwitch (McuVlprClockConfig)

VLPR System Clock Select. Configure the SCG_VCCR[SCS] register field. The system clock is either:

- Slow IRC (SCG_VCCR[SCS]=2)

Value extracted from Resource: MCU.VlprSystemClkSource.List

Note: Implementation Specific Parameter.

Table 4-50. Attribute McuSystemClockSwitch (McuVlprClockConfig) detailed description

Property	Value
Label	VLPR System Clock Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false

4.6.8.5.7 McuCoreClockDivider (McuVlprClockConfig)

Configures the SCG_VCCR[DIVCORE] bitfield This parameter represents the core clock divider. Note: implementation specific parameter.

Table 4-51. Attribute McuCoreClockDivider (McuVlprClockConfig) detailed description

Property	Value
Label	VLPR Core Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	2
Invalid	Range <=16 >=1

4.6.8.5.8 McuBusClockDivider (McuVlprClockConfig)

Configures the SCG_VCCR[DIVBUS] bitfield This parameter represents the bus clock divider. Note: implementation specific parameter.

Table 4-52. Attribute McuBusClockDivider (McuVlprClockConfig) detailed description

Property	Value
Label	VLPR Bus Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range <=16 >=1

4.6.8.5.9 McuSlowClockDivider (McuVlprClockConfig)

Configures the SCG_RCCR[DIVSLOW] bitfield.

This parameter represents the flash clock divider.

Note: Configuring DIVSLOW to lower frequencies than supported flash frequencies mentioned in datasheet (fFLASH) adds wait states and no power saving. It is recommended to configure DIVSLOW as close to fFLASH.

Table 4-53. Attribute McuSlowClockDivider (McuVlprClockConfig) detailed description

Property	Value
Label	VLPR Slow Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	4
Invalid	Range <=8 >=1

4.6.8.5.10 McuScgClkOutFrequency (McuVlprClockConfig)

This is frequency of SCG clockout. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-54. Attribute McuScgClkOutFrequency (McuVlprClockConfig) detailed description

Property	Value
Label	VLPR SCG ClockOut Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.6 Form McuHsrunClockConfig

This container configures the system clock source and the system clock dividers for the core, platform, external and bus clock domains when in HSRUN mode only.

Note: Implementation specific Container.

Is included by form : Form McuClockSettingConfig

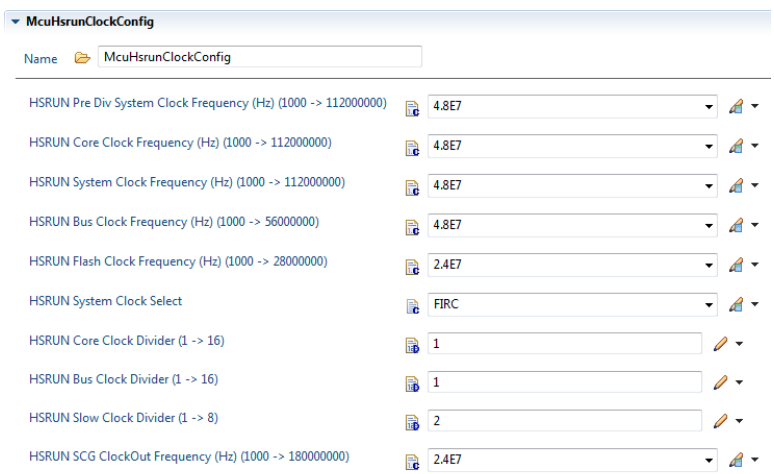


Figure 4-11. Tresos Plugin snapshot for McuHsrunClockConfig form.

4.6.8.6.1 McuPreDivSystemClockFrequency (McuHsrunClockConfig)

HSRUN System clock - Pre Divide System Clock Frequency.

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Note: PREDIV_SYS_CLK is only available in S32K148.

Note: Implementation Specific Parameter.

Table 4-55. Attribute McuPreDivSystemClockFrequency (McuHsrunClockConfig) detailed description

Property	Value
Label	HSRUN Pre Div System Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.6.2 McuCoreClockFrequency (McuHsrunClockConfig)

HSRUN Core clock - Clocks the ARM core, divided by DIVCORE bits inside SCG

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-56. Attribute McuCoreClockFrequency (McuHsrunClockConfig) detailed description

Property	Value
Label	HSRUN Core Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.6.3 McuSystemClockFrequency (McuHsrunClockConfig)

HSRUN System clock - Clocks the Crossbar, NVIC, Flash controller, FTM and PDB,etc.

HSRUN_SYS_CLK can run up to CORE_CLK.

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-57. Attribute McuSystemClockFrequency (McuHsrunClockConfig) detailed description

Property	Value
Label	HSRUN System Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.6.4 McuBusClockFrequency (McuHsrunClockConfig)

HSRUN Bus clock - BUS_CLK Clocks the Peripherals, divided by DIVBUS bits inside SCG

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-58. Attribute McuBusClockFrequency (McuHsrunClockConfig) detailed description

Property	Value
Label	HSRUN Bus Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.6.5 McuFlashClockFrequency (McuHsrunClockConfig)

HSRUN Flash clock - Clocks the flash module, divided by DIVSLOW bits inside SCG

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-59. Attribute McuFlashClockFrequency (McuHsrunClockConfig) detailed description

Property	Value
Label	HSRUN Flash Clock Frequency (Hz)
Type	FLOAT

Table continues on the next page...

Table 4-59. Attribute McuFlashClockFrequency (McuHsrunClockConfig) detailed description (continued)

Property	Value
Origin	Custom
Symbolic Name	false

4.6.8.6.6 McuSystemClockSwitch (McuHsrunClockConfig)

HSRUN System Clock Select. Configure the SCG_RCCR[SCS] register field. The system clock is either:

- System OSC (SCG_RCCR[SCS]=1)

- Slow IRC (SCG_RCCR[SCS]=2)

- Fast IRC (SCG_RCCR[SCS]=3)

- System PLL (SCG_RCCR[SCS]=6)

Value extracted from Resource: MCU.HsrunSystemClkSource.List

Note: Implementation Specific Parameter.

Table 4-60. Attribute McuSystemClockSwitch (McuHsrunClockConfig) detailed description

Property	Value
Label	HSRUN System Clock Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false

4.6.8.6.7 McuCoreClockDivider (McuHsrunClockConfig)

Configures the SCG_HCCR[DIVCORE] bitfield This parameter represents the core clock divider. Note: implementation specific parameter.

Table 4-61. Attribute McuCoreClockDivider (McuHsrunClockConfig) detailed description

Property	Value
Label	HSRUN Core Clock Divider

Table continues on the next page...

Table 4-61. Attribute McuCoreClockDivider (McuHsrunClockConfig) detailed description (continued)

Property	Value
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	XPath Range <=16 >=1

4.6.8.6.8 McuBusClockDivider (McuHsrunClockConfig)

Configures the SCG_HCCR[DIVBUS] bitfield This parameter represents the bus clock divider. Note: implementation specific parameter.

Table 4-62. Attribute McuBusClockDivider (McuHsrunClockConfig) detailed description

Property	Value
Label	HSRUN Bus Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range <=16 >=1

4.6.8.6.9 McuSlowClockDivider (McuHsrunClockConfig)

Configures the SCG_RCCR[DIVSLOW] bitfield.

This parameter represents the flash clock divider.

Note: Configuring DIVSLOW to lower frequencies than supported flash frequencies mentioned in datasheet (fFLASH) adds wait states and no power saving. It is recommended to configure DIVSLOW as close to fFLASH.

Table 4-63. Attribute McuSlowClockDivider (McuHsrunClockConfig) detailed description

Property	Value
Label	HSRUN Slow Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	2
Invalid	Range <=8 >=1

4.6.8.6.10 McuScgClkOutFrequency (McuHsrunClockConfig)

This is frequency of SCG clockout. It is given in Hz.

Note: Implementation Specific Parameter.

Table 4-64. Attribute McuScgClkOutFrequency (McuHsrunClockConfig) detailed description

Property	Value
Label	HSRUN SCG ClockOut Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.7 Form McuSystemOSCClockConfig

Configures System OSC registers.

Note: Implementation specific Container.

Is included by form : Form McuClockSettingConfig

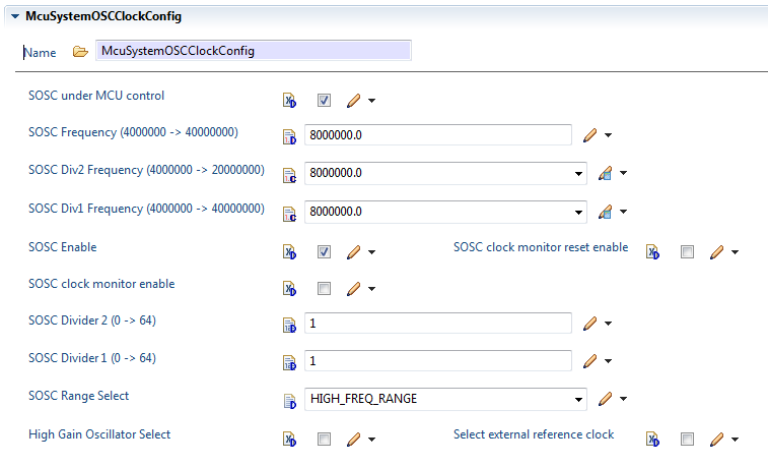


Figure 4-12. Tresos Plugin snapshot for McuSystemOSCClockConfig form.

4.6.8.7.1 McuSOSCUnderMcuControl (McuSystemOSCClockConfig)

0 - System OSC is NOT under mcu control.

1 - System OSC is under mcu control.

If this is set to false, the MCU code will not configure the SOSC registers when Mcu_InitClock is called

Note: Implementation Specific Parameter.

Table 4-65. Attribute McuSOSCUnderMcuControl (McuSystemOSCClockConfig) detailed description

Property	Value
Label	SOSC under MCU control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.7.2 McuSOSCFrequency (McuSystemOSCClockConfig)

This is the SOSC frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

If PLL is used, then oscillator needs to be in high range only, SCG_SOSCCFG[RANGE] on 11 as used in reference clock.

Note: Implementation Specific Parameter.

Table 4-66. Attribute McuSOSCFrequency (McuSystemOSCClockConfig) detailed description

Property	Value
Label	SOSC Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false
Default	8000000

4.6.8.7.3 McuSOSCDiv2Frequency (McuSystemOSCClockConfig)

This is the SOSC Divider 2 frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 40 MHz or less in RUN/HSRUN mode.

Note: Implementation Specific Parameter.

Table 4-67. Attribute McuSOSCDiv2Frequency (McuSystemOSCClockConfig) detailed description

Property	Value
Label	SOSC Div2 Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.7.4 McuSOSCDiv1Frequency (McuSystemOSCClockConfig)

This is the SOSC Divider 1 frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 40 MHz or less in RUN/HSRUN mode.

Note: Implementation Specific Parameter.

Table 4-68. Attribute McuSOSCDiv1Frequency (McuSystemOSCClockConfig) detailed description

Property	Value
Label	SOSC Div1 Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.7.5 McuSOSCEnable (McuSystemOSCClockConfig)

SCG_SOSCCSR[SOSCEN] - System OSC Enable

0 - System OSC is disabled.

1 - System OSC is enabled.

Note: Implementation Specific Parameter.

Table 4-69. Attribute McuSOSCEnable (McuSystemOSCClockConfig) detailed description

Property	Value
Label	SOSC Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.7.6 McuSOSCClockMonitorResetEnable (McuSystemOSCClockConfig)

SCG_SOSCCSR[SOSCCMRE] - System OSC Clock Monitor Reset Enable

0 - Clock Monitor generates interrupt when error detected.

1 - Clock Monitor generates reset when error detected.

Note: Implementation Specific Parameter.

Table 4-70. Attribute McuSOSCClockMonitorResetEnable (McuSystemOSCClockConfig) detailed description

Property	Value
Label	SOSC clock monitor reset enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.7.7 McuSOSCClockMonitorEnable (McuSystemOSCClockConfig)

SCG_SOSCCSR[SOSCCM] - System OSC Clock Monitor

Enables the clock monitor, if the clock source is disabled in a low power mode then the clock monitor is also disabled in the low power mode. When the clock monitor is disabled in a low power mode, it remains disabled until the clock valid flag is set following exit from the low power mode.

0 - System OSC Clock Monitor is disabled.

1 - System OSC Clock Monitor is enabled.

Note: Implementation Specific Parameter.

Table 4-71. Attribute McuSOSCClockMonitorEnable (McuSystemOSCClockConfig) detailed description

Property	Value
Label	SOSC clock monitor enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.7.8 McuSOSCDiv2 (McuSystemOSCClockConfig)

Configures SCG_SOSCDIV[SOSCDIV2].

System OSC Clock Divide 2.

Clock divider 2 for System OSC. Used by bus clock modules that need an asynchronous clock source.

0 - Output disabled.

Note: Implementation specific Parameter.

Table 4-72. Attribute McuSOSCDiv2 (McuSystemOSCClockConfig) detailed description

Property	Value
Label	SOSC Divider 2
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range XPath <=64 >=0

4.6.8.7.9 McuSOSCDiv1 (McuSystemOSCClockConfig)

Configures SCG_SOSCDIV[SOSCDIV1]

System OSC Clock Divide 1.

Clock divider 1 for System OSC. Used to generate the system clock source and by platform clock modules that need an asynchronous clock source.

0 - Output disabled.

Note: Implementation specific Parameter.

Table 4-73. Attribute McuSOSCDiv1 (McuSystemOSCClockConfig) detailed description

Property	Value
Label	SOSC Divider 1
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range XPath

Table 4-73. Attribute McuSOSCDiv1 (McuSystemOSCClockConfig) detailed description

Property	Value
	<=64 >=0

4.6.8.7.10 McuSOSCRangeSelect (McuSystemOSCClockConfig)

SCG_SOSCCFG[RANGE] - System OSC Range Select

Selects the frequency range for the system crystal oscillator (OSC)

Note

The following constraints are not checked by the xdm schema:

If PLL is used, then oscillator needs to be in high range only, SCG_SOSCCFG[RANGE] on 11 as used in reference clock.

- Medium frequency range selected for the crytstal oscillator of 4 MHz to 8 MHz.

- High frequency range selected for the crystal oscillator of 8 MHz to 32 MHz.

Note: Implementation Specific Parameter.

Table 4-74. Attribute McuSOSCRangeSelect (McuSystemOSCClockConfig) detailed description

Property	Value
Label	SOSC Range Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	HIGH_FREQ_RANGE
Range	MEDIUM_FREQ_RANGE HIGH_FREQ_RANGE

4.6.8.7.11 McuSOSCHighGainOscillatorSelect (McuSystemOSCClockConfig)

SCG_SOSCCFG[HGO] - High Gain Oscillator Select

Controls the crystal oscillator power mode of operations.

unchecked - Configure crystal oscillator for low-power operation

checked - Configure crystal oscillator for high-gain operation

Note: Implementation Specific Parameter.

Table 4-75. Attribute McuSOSCHighGainOscillatorSelect (McuSystemOSCClockConfig) detailed description

Property	Value
Label	High Gain Oscillator Select
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.7.12 McuSOSCExternalReferenceSelect (McuSystemOSCClockConfig)

SCG_SOSCCFG[EREFS] - External Reference Select

Selects the source for the external reference clock.

unchecked - Internal oscillator of OSC requested.

checked - External reference clock from PAD pin selected

Note: Implementation Specific Parameter.

Table 4-76. Attribute McuSOSCExternalReferenceSelect (McuSystemOSCClockConfig) detailed description

Property	Value
Label	Select external reference clock
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.8 Form McuSIRCClockConfig

Configures Slow IRC (SIRC) registers.

Note: Implementation specific Container.

Is included by form : Form McuClockSettingConfig

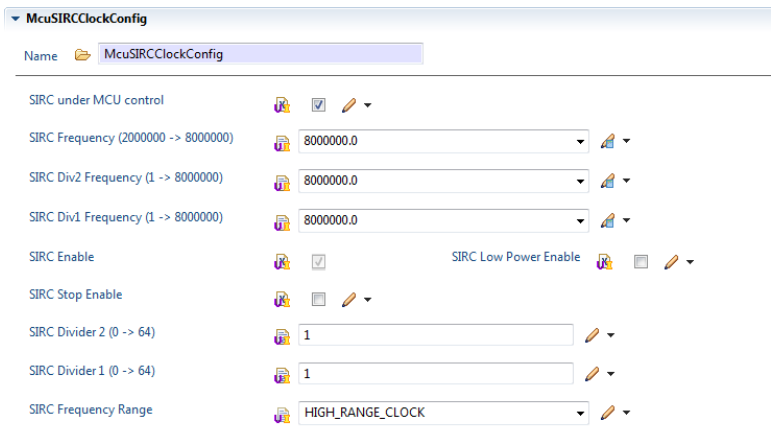


Figure 4-13. Tresos Plugin snapshot for McuSIRCClockConfig form.

4.6.8.8.1 McuSIRCUnderMcuControl (McuSIRCClockConfig)

0 - Slow IRC is NOT under mcu control.

1 - Slow IRC is under mcu control.

If this is set to false, the MCU code will not configure the SIRC registers when Mcu_InitClock is called

Note: Implementation Specific Parameter.

Table 4-77. Attribute McuSIRCUnderMcuControl (McuSIRCClockConfig) detailed description

Property	Value
Label	SIRC under MCU control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.8.2 McuSIRCFrequency (McuSIRCClockConfig)

This is the SIRC frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

Note: Implementation Specific Parameter.

Table 4-78. Attribute McuSIRCFrequency (McuSIRCClockConfig) detailed description

Property	Value
Label	SIRC Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false
Invalid	Range <=8000000 >=2000000

4.6.8.8.3 McuSIRCDiv2Frequency (McuSIRCClockConfig)

This is the SIRC Divider 2 frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 8 MHz or less.

Note: Implementation Specific Parameter.

Table 4-79. Attribute McuSIRCDiv2Frequency (McuSIRCClockConfig) detailed description

Property	Value
Label	SIRC Div2 Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.8.4 McuSIRCDiv1Frequency (McuSIRCClockConfig)

This is the SIRC Divider 1 frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 8 MHz or less in RUN/HSRUN mode and to 4 MHz or less in VLPR mode.

Table 4-80. Attribute McuSIRCDiv1Frequency (McuSIRCClockConfig) detailed description

Property	Value
Label	SIRC Div1 Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false
Invalid	Range <=8000000 >=1

4.6.8.8.5 McuSIRCEnable (McuSIRCClockConfig)

SCG_SIRCCSR[SIRCEN] - Slow IRC Enable

0 - Slow IRC is disabled.

1 - Slow IRC is enabled.

Note: Implementation Specific Parameter.

The Node is not editable. This is needed for switching system clock.

Table 4-81. Attribute McuSIRCEnable (McuSIRCClockConfig) detailed description

Property	Value
Label	SIRC Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.8.6 McuSIRCLowPowerEnable (McuSIRCClockConfig)

SCG_SIRCCSR[SIRCLPEN] - Slow IRC Low Power Enable

0 - Slow IRC is disabled in VLP modes.

1 - Slow IRC is enabled in VLP modes.

Note: Implementation Specific Parameter.

Table 4-82. Attribute McuSIRCLowPowerEnable (McuSIRCClockConfig) detailed description

Property	Value
Label	SIRC Low Power Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.8.7 McuSIRCStopEnable (McuSIRCClockConfig)

SCG_SIRCCSR[SIRCSTEN] - Slow IRC Enable

0 - Slow IRC is disabled.

1 - Slow IRC is enabled.

Note: Implementation Specific Parameter.

Table 4-83. Attribute McuSIRCStopEnable (McuSIRCClockConfig) detailed description

Property	Value
Label	SIRC Stop Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.8.8 McuSIRCDiv2 (McuSIRCClockConfig)

Configures SCG_SIRCDIV[SIRCDIV2].

Slow IRC Clock Divider 2.

Clock divider 2 for Slow IRC. Used by bus clock modules that need an asynchronous clock source.

0 - Output disabled.

Note: Implementation specific Parameter.

Table 4-84. Attribute McuSIRCDiv2 (McuSIRCClockConfig) detailed description

Property	Value
Label	SIRC Divider 2
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	2
Invalid	Range XPath <=64 >=0

4.6.8.8.9 McuSIRCDiv1 (McuSIRCClockConfig)

Configures SCG_SIRCDIV[SIRCDIV1].

Slow IRC Clock Divider 1.

Clock divider 1 for Slow IRC. Used to generate the system clock source and by platform clock modules that need an asynchronous clock source.

0 - Output disabled.

Note: Implementation specific Parameter.

Table 4-85. Attribute McuSIRCDiv1 (McuSIRCClockConfig) detailed description

Property	Value
Label	SIRC Divider 1
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range XPath <=64 >=0

4.6.8.8.10 McuSIRCRangeSelect (McuSIRCClockConfig)

SCG_SIRCCFG[RANGE] - Selects the Frequency Range

Slow IRC low range clock (2 MHz)

Slow IRC high range clock (8 MHz)

Note: The SIRC clock is chosen as source clock that must be sacrificed to be ON at all times.

Add addition, Software should not configure the SCG_SIRCCFG[RANGE] to any value other than HIGH_RANGE_CLOCK.

Note: Implementation Specific Parameter.

Table 4-86. Attribute McuSIRCRangeSelect (McuSIRCClockConfig) detailed description

Property	Value
Label	SIRC Frequency Range
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	HIGH_RANGE_CLOCK
Range	LOW_RANGE_CLOCK HIGH_RANGE_CLOCK

4.6.8.9 Form McuFIRCClockConfig

Configures Fast IRC (FIRC) registers.

Note: Implementation specific Container.

Is included by form : Form McuClockSettingConfig

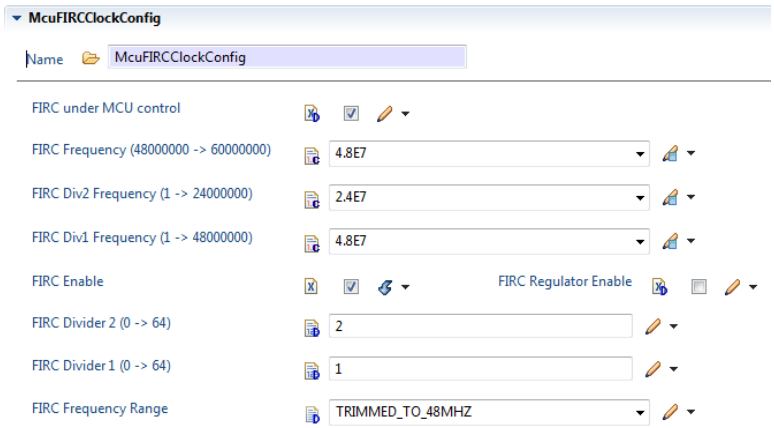


Figure 4-14. Tresos Plugin snapshot for McuFIRCClockConfig form.

4.6.8.9.1 McuFIRCUnderMcuControl (McuFIRCClockConfig)

0 - Fast IRC is NOT under mcu control.

1 - Fast IRC is under mcu control.

If this is set to false, the MCU code will not configure the FIRC registers when Mcu_InitClock is called

Note: Implementation Specific Parameter.

Table 4-87. Attribute McuFIRCUnderMcuControl (McuFIRCClockConfig) detailed description

Property	Value
Label	FIRC under MCU control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.9.2 McuFIRCFrequency (McuFIRCClockConfig)

This is the FIRC frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

Table 4-88. Attribute McuFIRCFrequency (McuFIRCClockConfig) detailed description

Property	Value
Label	FIRC Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false
Invalid	Range <=60000000 >=48000000

4.6.8.9.3 McuFIRCDiv2Frequency (McuFIRCClockConfig)

This is the FIRC Divider 2 frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 48 MHz or less in RUN/HSRUN mode.

Note: Implementation specific Container.

Table 4-89. Attribute McuFIRCDiv2Frequency (McuFIRCClockConfig) detailed description

Property	Value
Label	FIRC Div2 Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.9.4 McuFIRCDiv1Frequency (McuFIRCClockConfig)

This is the FIRC Divider 1 frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 48 MHz or less in RUN/HSRUN mode.

Note: Implementation specific Container.

Table 4-90. Attribute McuFIRCDiv1Frequency (McuFIRCClockConfig) detailed description

Property	Value
Label	FIRC Div1 Frequency
Type	FLOAT

Table continues on the next page...

Table 4-90. Attribute McuFIRCDiv1Frequency (McuFIRCClockConfig) detailed description (continued)

Property	Value
Origin	Custom
Symbolic Name	false

4.6.8.9.5 McuFIRCEnable (McuFIRCClockConfig)

SCG_FIRCCSR[FIRCEN] - Fast IRC Enable

0 - Fast IRC is disabled.

1 - Fast IRC is enabled.

Note: Implementation specific Container.

Table 4-91. Attribute McuFIRCEnable (McuFIRCClockConfig) detailed description

Property	Value
Label	FIRC Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.9.6 McuFIRCRegulatorEnable (McuFIRCClockConfig)

Fast IRC Regulator Enable

0 - Fast IRC Regulator is disabled. (SCG_FIRCCSR[FIRCREGOFF] = 1)

1 - Fast IRC Regulator is enabled. (SCG_FIRCCSR[FIRCREGOFF] = 0)

Note: Implementation Specific Parameter.

Table 4-92. Attribute McuFIRCRegulatorEnable (McuFIRCClockConfig) detailed description

Property	Value
Label	FIRC Regulator Enable
Type	BOOLEAN

Table continues on the next page...

Table 4-92. Attribute McuFIRCRegulatorEnable (McuFIRCClockConfig) detailed description (continued)

Property	Value
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.9.7 McuFIRCDiv2 (McuFIRCClockConfig)

Configures SCG_FIRCDIV[FIRCDIV2]

Fast IRC Clock Divider 2.

Clock divider 2 for the Fast IRC. Used by bus clock modules that need an asynchronous clock source.

0 - Output disabled.

Note: Implementation specific Parameter.

Table 4-93. Attribute McuFIRCDiv2 (McuFIRCClockConfig) detailed description

Property	Value
Label	FIRC Divider 2
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	2
Invalid	Range XPath <=64 >=0

4.6.8.9.8 McuFIRCDiv1 (McuFIRCClockConfig)

Configures SCG_FIRCDIV[FIRCDIV1]

Fast IRC Clock Divider 1.

Clock divider 1 for Fast IRC. Used to generate the system clock source and by platform clock modules that need an asynchronous clock source.

0 - Output disabled.

Note: Implementation specific Parameter.

Table 4-94. Attribute McuFIRCDiv1 (McuFIRCClockConfig) detailed description

Property	Value
Label	FIRC Divider 1
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range XPath <=64 >=0

4.6.8.9.9 McuFIRCRangeSelect (McuFIRCClockConfig)

SCG_FIRCCFG[RANGE] - Selects the Frequency Range.

00b - Fast IRC is trimmed to 48 MHz.

01b - Reserved.

10b - Reserved.

11b - Reserved.

Note: Software should not configure the SCG_FIRCCFG[RANGE] to any value other than Fast IRC is trimmed to 48 MHz.

Note: Implementation Specific Parameter.

Table 4-95. Attribute McuFIRCRangeSelect (McuFIRCClockConfig) detailed description

Property	Value
Label	FIRC Frequency Range
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	TRIMMED_TO_48MHZ
Range	TRIMMED_TO_48MHZ

4.6.8.10 Form McuSystemPll

This container provides the specific configuration for the System PLL. Note: Implementation Specific Container.

Is included by form : Form McuClockSettingConfig

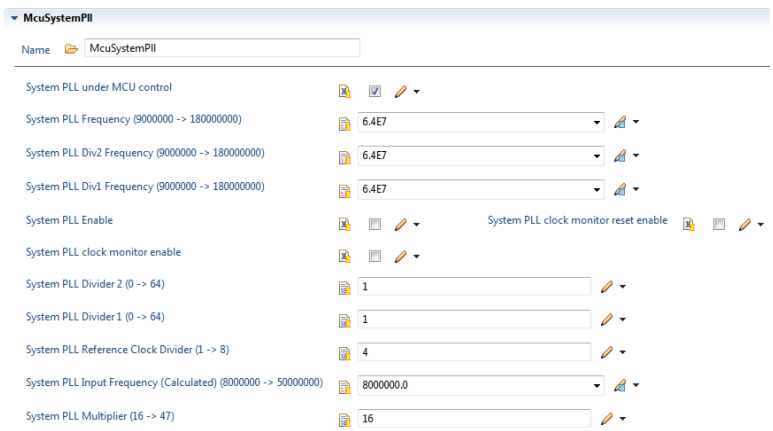


Figure 4-15. Tresos Plugin snapshot for McuSystemPll form.

4.6.8.10.1 McuSystemPllUnderMcuControl (McuSystemPll)

Set this to TRUE if System PLL is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock refference points

Note: Implementation Specific Parameter.

Table 4-96. Attribute McuSystemPllUnderMcuControl (McuSystemPll) detailed description

Property	Value
Label	System PLL under MCU control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.10.2 McuSPLLFrequency (McuSystemPll)

This is the System PLL frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

Note: Pll_freq = (McuSPLLInputFrequency * McuSPLLMultiplier)/2

Note: Implementation Specific Parameter.

Table 4-97. Attribute McuSPLLFrequency (McuSystemPll) detailed description

Property	Value
Label	System PLL Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false
Invalid	Range <=160000000 >=90000000

4.6.8.10.3 McuSPLLDiv2Frequency (McuSystemPll)

This is the System PLL Divider 2 frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 40 MHz or less in RUN mode and 56 MHz or less in HSRUN mode.

Note: Implementation Specific Parameter.

Table 4-98. Attribute McuSPLLDiv2Frequency (McuSystemPll) detailed description

Property	Value
Label	System PLL Div2 Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.10.4 McuSPLLDiv1Frequency (McuSystemPll)

This is the System PLL Divider 1 frequency for the specific instance of the McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 80MHz or less in RUN mode and to 112 MHz or less in HSRUN mode.

Note: Implementation Specific Parameter.

Table 4-99. Attribute McuSPLLDiv1Frequency (McuSystemPll) detailed description

Property	Value
Label	System PLL Div1 Frequency
Type	FLOAT
Origin	Custom
Symbolic Name	false

4.6.8.10.5 McuSPLLEnable (McuSystemPll)

SCG_SPLLCSR[SPLLEN] - System PLL Enable

0 - System PLL is disabled.

1 - System PLL is enabled.

Note: Implementation Specific Parameter.

Table 4-100. Attribute McuSPLLEnable (McuSystemPll) detailed description

Property	Value
Label	System PLL Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.10.6 McuSPLLClockMonitorResetEnable (McuSystemPll)

SCG_SPLLCSR[SPLLCMRE] - System PLL Clock Monitor Reset Enable

0 - Clock Monitor generates interrupt when error detected.

1 - Clock Monitor generates reset when error detected.

Note: Implementation Specific Parameter.

Table 4-101. Attribute McuSPLLClockMonitorResetEnable (McuSystemPll) detailed description

Property	Value
Label	System PLL clock monitor reset enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.10.7 McuSPLLClockMonitorEnable (McuSystemPll)

SCG_SPLLCSR[SPLLCM] - System PLL Clock Monitor

Enables the clock monitor, if the clock source is disabled in a low power mode then the clock monitor is also disabled in the low power mode. When the clock monitor is disabled in a low power mode, it remains disabled until the clock valid flag is set following exit from the low power mode.

0 - RTC OSC Clock Monitor is disabled.

1 - RTC OSC Clock Monitor is enabled.

Note: Implementation Specific Parameter.

Table 4-102. Attribute McuSPLLClockMonitorEnable (McuSystemPll) detailed description

Property	Value
Label	System PLL clock monitor enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.10.8 McuSPLLDiv2 (McuSystemPll)

Configures SCG_SPLLDIV[SPLLDIV2].

System PLL Clock Divider 2.

Clock divider 2 for System PLL. Used by bus clock modules that need an asynchronous clock source.

0 - Output disabled.

Note: Implementation specific Parameter.

Table 4-103. Attribute McuSPLLDiv2 (McuSystemPll) detailed description

Property	Value
Label	System PLL Divider 2
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	2
Invalid	Range XPath <=64 >=0

4.6.8.10.9 McuSPLLDiv1 (McuSystemPll)

Configures SCG_SPLLDIV[SPLLDIV1]

System PLL Clock Divider 1.

Clock divider 1 for System PLL. Used to generate the system clock source and by platform clock modules that need an asynchronous clock source.

0 - Output disabled.

Note: Implementation specific Parameter.

Table 4-104. Attribute McuSPLLDiv1 (McuSystemPll) detailed description

Property	Value
Label	System PLL Divider 1
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1

Table continues on the next page...

Table 4-104. Attribute McuSPLLDiv1 (McuSystemPll) detailed description (continued)

Property	Value
Invalid	Range XPath <=64 >=0

4.6.8.10.10 McuSPLLInputClkPreDivider (McuSystemPll)

PLL Reference Clock Divider.

Set the SPLL: SCG_SPLLCFG[PREDIV] field register.

Selects the amount to divide down the reference clock for the System PLL. The resulting frequency must be in the range of 8 MHz to 50 MHz.

Note: Implementation Specific Parameter.

Table 4-105. Attribute McuSPLLInputClkPreDivider (McuSystemPll) detailed description

Property	Value
Label	System PLL Reference Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range <=8 >=1

4.6.8.10.11 McuSPLLInputFrequency (McuSystemPll)

Resulted frequency after prediving.

F = SPLL_ReferenceClock / McuSPLLInputClkPreDivider.

Note: Implementation Specific Parameter.

Table 4-106. Attribute McuSPLLInputFrequency (McuSystemPll) detailed description

Property	Value
Label	System PLL Input Frequency (Calculated)
Type	FLOAT
Origin	Custom
Symbolic Name	false
Invalid	Range <=40000000 >=8000000

4.6.8.10.12 McuSPLLMultiplier (McuSystemPll)

System PLL Multiplier.

Set the SCG_SPLLCFG[MULT] field register.

Valid range is in [16..47]. Note: Implementation Specific Parameter.

Table 4-107. Attribute McuSPLLMultiplier (McuSystemPll) detailed description

Property	Value
Label	System PLL Multiplier
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	24
Invalid	Range <=47 >=16

4.6.8.11 Form McuSIMClockConfig

Configures SIM_CHIPCTL[TRACECLK_SEL], SIM_CHIPCTL[CLKOUTSEL] bits and SIM_PLATGC and SIM_CLKDIV4 registers.

Note: Implementation specific Container.

Is included by form : Form McuClockSettingConfig

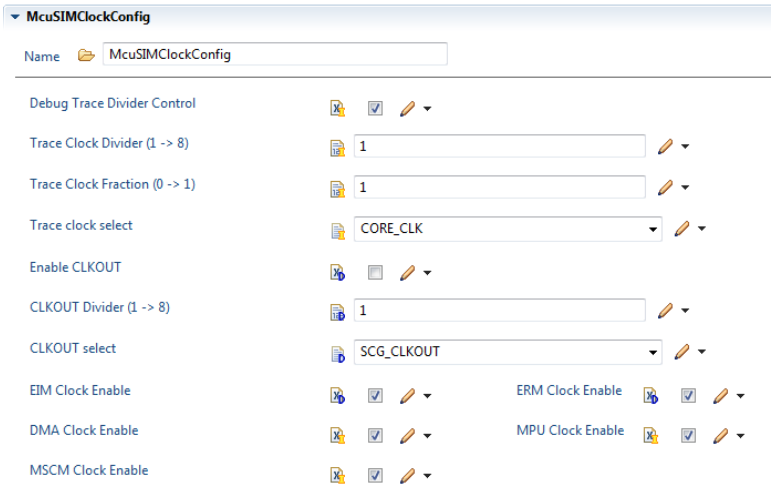


Figure 4-16. Tresos Plugin snapshot for McuSIMClockConfig form.

4.6.8.11.1 McuDebugTraceDividerEnable (McuSIMClockConfig)

SIM_CLKDIV4[TRACEDIVEN] - Debug Trace Divider Control

0 - Debug trace divider disabled.

1 - Debug trace divider enabled.

Note: Implementation Specific Parameter.

Table 4-108. Attribute McuDebugTraceDividerEnable (McuSIMClockConfig) detailed description

Property	Value
Label	Debug Trace Divider Control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.11.2 McuTraceClockDivider (McuSIMClockConfig)

Configures the SIM_CLKDIV4[TRACEDIV] bitfield Trace clock divider divisor - This field sets the divide value for the fractional clock divider used as a source for trace clock. The source clock for the trace clock is set by the SIM_CHIPCTRL[TRACECLK_SEL]. Divider output clock = Divider input clock * [(TRACEFRAC+1)/(TRACEDIV+1)]. Note: implementation specific parameter.

Table 4-109. Attribute McuTraceClockDivider (McuSIMClockConfig) detailed description

Property	Value
Label	Trace Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range <=8 >=1

4.6.8.11.3 McuTraceClockFraction (McuSIMClockConfig)

Configures the SIM_CLKDIV4[TRACEFRAC] bitfield Trace clock divider fraction - This field sets the divide value for the fractional clock divider used as a source for trace clock. The source clock for the trace clock is set by the SIM_CHIPCTRL[TRACECLK_SEL]. Divider output clock = Divider input clock * [(TRACEFRAC+1)/(TRACEDIV+1)]. Note: implementation specific parameter.

Table 4-110. Attribute McuTraceClockFraction (McuSIMClockConfig) detailed description

Property	Value
Label	Trace Clock Fraction
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	0
Invalid	Range <=1 >=0

4.6.8.11.4 McuTraceClockSelect (McuSIMClockConfig)

SIM_CHIPCTL[TRACECLK_SEL] - Debug trace clock select

Selects core clock or platform clock as the trace clock source.

Note: Implementation Specific Parameter.

Table 4-111. Attribute McuTraceClockSelect (McuSIMClockConfig) detailed description

Property	Value
Label	Trace clock select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	CORE_CLK
Range	CORE_CLK PLATFORM_CLK

4.6.8.11.5 McuClockOutEnable (McuSIMClockConfig)

SIM_CHIPCTL[CLKOUTEN] - CLKOUT enable

unchecked - Clockout disabled.

checked - Clockout enabled.

Note: Implementation Specific Parameter.

Table 4-112. Attribute McuClockOutEnable (McuSIMClockConfig) detailed description

Property	Value
Label	Enable CLKOUT
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.11.6 McuClockOutDivider (McuSIMClockConfig)

Configures the SIM_CHIPCTL[CLKOUTDIV] bitfield CLKOUT Divide Ratio. Note: implementation specific parameter.

Table 4-113. Attribute McuClockOutDivider (McuSIMClockConfig) detailed description

Property	Value
Label	CLKOUT Divider
Type	INTEGER
Origin	Custom

Table continues on the next page...

Table 4-113. Attribute McuClockOutDivider (McuSIMClockConfig) detailed description (continued)

Property	Value
Symbolic Name	false
Default	1
Invalid	Range <=8 >=1

4.6.8.11.7 McuClockOutSelect (McuSIMClockConfig)

SIM_CHIPCTL[CLKOUTSEL] - CLKOUT select

Selects the clock to output on the CLKOUT pin.

0 SCG_CLKOUT

2 SOSC_DIV2

4 SIRC_DIV2

5 QSPI_SFIF_CLK_HYP Divide by 2 clock (configured through SCLKCONFIG[5]) for HyperRAM going to sfif clock to QSPI(Specific for S32K148)

6 FIRC_DIV2

7 HCLK

8 SPLL_DIV2

9 BUS_CLK

A LPO_CLK_128K

B QSPI_IPG_CLK (Specific for S32K148)

C LPO_CLK as selected by SIM_LPOCLKS[LPOCLKSEL]

D QSPI_IPG_CLK_SFIF (Specific for S32K148)

E RTC_CLK as selected by SIM CLK 32 KHz Select

F QSPI_IPG_CLK_2XSFIF (Specific for S32K148)

Note: Implementation Specific Parameter.

Table 4-114. Attribute McuClockOutSelect (McuSIMClockConfig) detailed description

Property	Value
Label	SIM CLKOUT select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	SCG_CLKOUT
Range	SCG_CLKOUT SOSC_DIV2 SIRC_DIV2 QSPI_SFIF_CLK_HYP FIRC_DIV2 HCLK SPLL_DIV2 BUS_CLK LPO_CLK_128K QSPI_IPG_CLK LPO_CLK QSPI_IPG_CLK_SFIF RTC_CLK QSPI_IPG_CLK_2XSFIF

4.6.8.11.8 McuEIMClockGatingEnable (McuSIMClockConfig)

SIM_PLATCGC[CGCEIM] - EIM Clock Gating Control

Controls the clock gating to the EIM.

0 - Clock disabled.

1 - Clock enabled.

Note: Implementation Specific Parameter.

Table 4-115. Attribute McuEIMClockGatingEnable (McuSIMClockConfig) detailed description

Property	Value
Label	EIM Clock Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.11.9 McuERMClockGatingEnable (McuSIMClockConfig)

SIM_PLATCGC[CGCERM] - ERM Clock Gating Control

Controls the clock gating to the ERM.

0 - Clock disabled.

1 - Clock enabled.

Note: Implementation Specific Parameter.

Table 4-116. Attribute McuERMClockGatingEnable (McuSIMClockConfig) detailed description

Property	Value
Label	ERM Clock Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.11.10 McuDMAClockGatingEnable (McuSIMClockConfig)

SIM_PLATCGC[CGCDMA] - DMA Clock Gating Control

Controls the clock gating to the DMA module.

0 - Clock disabled.

1 - Clock enabled.

Note: Implementation Specific Parameter.

Table 4-117. Attribute McuDMAClockGatingEnable (McuSIMClockConfig) detailed description

Property	Value
Label	DMA Clock Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.11.11 McuMPUClockGatingEnable (McuSIMClockConfig)

SIM_PLATCGC[CGCMPU] - MPU Clock Gating Control

Controls the clock gating to the MPU module.

0 - Clock disabled.

1 - Clock enabled.

Note: Implementation Specific Parameter.

Table 4-118. Attribute McuMPUClockGatingEnable (McuSIMClockConfig) detailed description

Property	Value
Label	MPU Clock Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.11.12 McuMSCMClockGatingEnable (McuSIMClockConfig)

SIM_PLATCGC[CGCMSCM] - MSCM Clock Gating Control

Controls the clock gating to the MSCM module.

0 - Clock disabled.

1 - Clock enabled.

Note: Implementation Specific Parameter.

Table 4-119. Attribute McuMSCMClockGatingEnable (McuSIMClockConfig) detailed description

Property	Value
Label	MSCM Clock Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.11.13 McuGPIOClockGatingEnable (McuSIMClockConfig)

SIM_PLATCGC[CGCGPIO] - GPIO Clock Gating Control

Controls the clock gating to the GPIO module.

0 - Clock disabled.

1 - Clock enabled.

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-120. Attribute McuGPIOClockGatingEnable (McuSIMClockConfig) detailed description

Property	Value
Label	GPIO Clock Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.12 Form McuPeripheralClockConfig

This contains the combination for current peripheral in Run and LowPower Mode.

Note: Implementation Specific Container.

Is included by form : Form McuClockSettingConfig

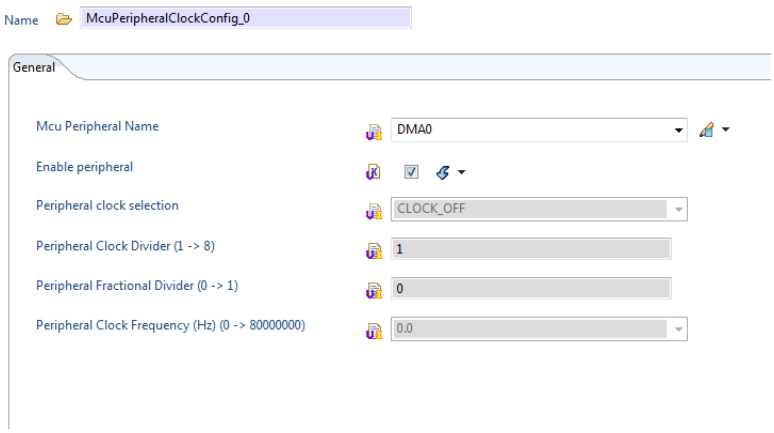


Figure 4-17. Tresos Plugin snapshot for McuPeripheralClockConfig form.

4.6.8.12.1 McuPerName (McuPeripheralClockConfig)

This is the name of the peripheral.

Note: Implementation Specific Parameter.

Table 4-121. Attribute McuPerName (McuPeripheralClockConfig) detailed description

Property	Value
Label	Mcu Peripheral Name
Type	ENUMERATION
Origin	Custom
Symbolic Name	false

4.6.8.12.2 McuPeripheralClockEnable (McuPeripheralClockConfig)

Sets PCC_[peripheral][CGC] bit. This read/write bit enables the clock for the peripheral.

0 - Clock disabled.

1 - Clock enabled.

Note: Implementation Specific Parameter.

Table 4-122. Attribute McuPeripheralClockEnable (McuPeripheralClockConfig) detailed description

Property	Value
Label	Enable peripheral
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	true

4.6.8.12.3 McuPeripheralClockSelect (McuPeripheralClockConfig)

Configures PCC_[peripheral][PCS].

This is used for peripherals that support various clock selections.

If the peripheral does not support various clock selections the field won't be editable.

This field can only be written when the CGC bit is 0 (clock disabled). Likewise, if the INUSE flag is set, this field is locked.

0 - Clock is off (or external clock as selected by FTMnCLKSEL for FTM modules).

1 - SOSCDIV2_CLK (SOCDIV1_CLK for FTM modules)

2 - SIRCDIV2_CLK (SIRCDIV1_CLK for FTM modules)

3 - FIRCDIV2_CLK (FIRCDIV1_CLK for FTM modules)

6 - SPLLDIV2_CLK (SPLLDIV1_CLK for FTM modules)

Note: Please make sure that the divider of clock source is not 0 .

Note: Implementation Specific Parameter.

If the field is not editable, it means the bit field is read only and the value will not be used.

Table 4-123. Attribute McuPeripheralClockSelect (McuPeripheralClockConfig) detailed description

Property	Value
Label	Peripheral clock selection
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	EXT_CLK_OR_CLK_OFF
Range	EXT_CLK_OR_CLK_OFF SOSC SIRC FIRC SPLL

4.6.8.12.4 McuPeripheralClockDivider (McuPeripheralClockConfig)

Configures PCC_[peripheral][PCD].

This is used for peripherals that require a clock divider. At SOC integration, each peripheral is assigned either a divider or not.

This field can only be written when the CGC bit is 0 (clock disabled). Likewise, if the INUSE flag is set, this field is locked.

Allowed values are from 1 to 8.

Note: Implementation Specific Parameter.

If the field is not editable, it means the bit field is read only and the value will not be used.

Table 4-124. Attribute McuPeripheralClockDivider (McuPeripheralClockConfig) detailed description

Property	Value
Label	Peripheral Clock Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range <=8 >=1

4.6.8.12.5 McuPeripheralFractionalDivider (McuPeripheralClockConfig)

Configures PCC_[peripheral][FRAC].

This sets the fraction multiply value for the fractional clock divider used as a clock source. Divider output clock = Divider input clock x [(FRAC+1)/(DIV+1)].

This field can only be written when the CGC bit is 0 (clock disabled). Likewise, if the INUSE flag is set, this field is locked.

0 - Fractional value is 0.

1 - Fractional value is 1.

Note: Implementation Specific Parameter.

If the field is not editable, it means the bit field is read only and the value will not be used.

Table 4-125. Attribute McuPeripheralFractionalDivider (McuPeripheralClockConfig) detailed description

Property	Value
Label	Peripheral Fractional Divider
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	0
Invalid	Range <=1 >=0

4.6.8.12.6 McuPeripheralClockFrequency (McuPeripheralClockConfig)

Divider output clock = Divider input clock x [(FRAC+1)/(DIV+1)].

This is only calculated if the clock source is selectable and if the peripheral is enabled.

This is the frequency for the specific instance of the "McuClockReferencePoint" container. Value calculated for user info. It is given in Hz.

Note: The maximum frequency of LPUARTx, LPSPIx, LPI2Cx, FLEXIO, LPTMR0 and LPIT are governed by BUS_CLK, FTMx are governed by SYS_CLK, ADCx are 50MHz but always less than BUS_CLK. So please check configuration value is fit for SYS_CLK and BUS_CLK values correspond to MCU mode.

Note: Implementation Specific Parameter.

Table 4-126. Attribute McuPeripheralClockFrequency (McuPeripheralClockConfig) detailed description

Property	Value
Label	Peripheral Clock Frequency (Hz)
Type	FLOAT
Origin	Custom
Symbolic Name	false
Invalid	Range <=112000000 >=0

4.6.8.13 Form McuClkMonitor_0

This container contains the specific configuration (parameters) of the Monitor_0. Clock Monitor Unit for Motor Clock. This parameter is enabled only if "McuClockSrcFailureNotification" is enabled.

Note: Implementation Specific Parameter.

Is included by form : Form McuClockSettingConfig

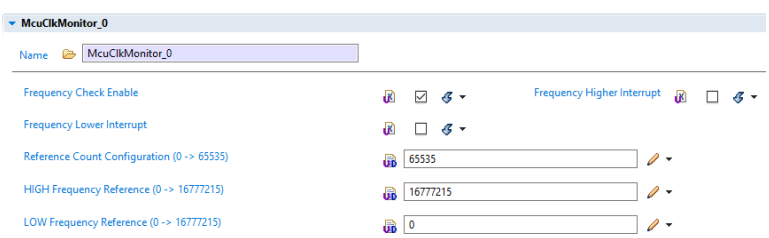


Figure 4-18. Tresos Plugin snapshot for McuClkMonitor_0 form.

4.6.8.13.1 McuClkMonitorEn (McuClkMonitor_0)

Enables/Disables the clock monitor (CMU_GCR[FCE]).

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-127. Attribute McuClkMonitorEn (McuClkMonitor_0) detailed description

Property	Value
Label	Frequency Check Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.13.2 McuFrequencyHigherInterrupt (McuClkMonitor_0)

Enables/Disables FHH asynchronous interrupt at the module boundary (CMU0_IER[FHHAIE]).

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-128. Attribute McuFrequencyHigherInterrupt (McuClkMonitor_0) detailed description

Property	Value
Label	Frequency Higher Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.13.3 McuFrequencyLowerInterrupt (McuClkMonitor_0)

Enables/Disables FLL asynchronous interrupt at the module boundary (CMU0_IER[FLLAIE]).

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-129. Attribute McuFrequencyLowerInterrupt (McuClkMonitor_0) detailed description

Property	Value
Label	Frequency Lower Interrupt
Type	BOOLEAN

Table continues on the next page...

Table 4-129. Attribute McuFrequencyLowerInterrupt (McuClkMonitor_0) detailed description (continued)

Property	Value
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.13.4 McuReferenceCountConfiguration (McuClkMonitor_0)

Set the CMU0: CMU_RCCR[REF_CNT] field register. CMU_RCCR[REF_CNT] is the number of reference clock counts that the frequency check runs. This field defines the duration of one frequency check window.

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-130. Attribute McuReferenceCountConfiguration (McuClkMonitor_0) detailed description

Property	Value
Label	Reference Count Configuration
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	65535

4.6.8.13.5 McuHighFrequencyRef (McuClkMonitor_0)

The value of the Max Frequency.

CMU0 detects if the monitored clock is greater than McuHighFrequencyRef. The HCTR[HFREF] value is determined by the equation:

((fMonitored_clock / fReference_clock) * RCCR[REF_CNT]) + high threshold margin

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-131. Attribute McuHighFrequencyRef (McuClkMonitor_0) detailed description

Property	Value
Label	HIGH Frequency Reference

Table continues on the next page...

Table 4-131. Attribute McuHighFrequencyRef (McuClkMonitor_0) detailed description (continued)

Property	Value
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	16777215

4.6.8.13.6 McuLowFrequencyRef (McuClkMonitor_0)

The value of the Min Frequency. CMU0 detects if the monitored clock is smaller than McuLowFrequencyRef. LCTR[LFREF] determines the low reference value for the monitored clock frequency. The LCTR[LFREF] value is determined by the equation: fMonitored_clock / fReference_clock * RCCR[REF_CNT] - low threshold margin. Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-132. Attribute McuLowFrequencyRef (McuClkMonitor_0) detailed description

Property	Value
Label	LOW Frequency Reference
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	0

4.6.8.14 Form McuClkMonitor_1

This container contains the specific configuration (parameters) of the Monitor_1. Clock Monitor Unit for Motor Clock. This parameter is enabled only if "McuClockSrcFailureNotification" is enabled.

Note: Implementation Specific Parameter.

Is included by form : Form McuClockSettingConfig

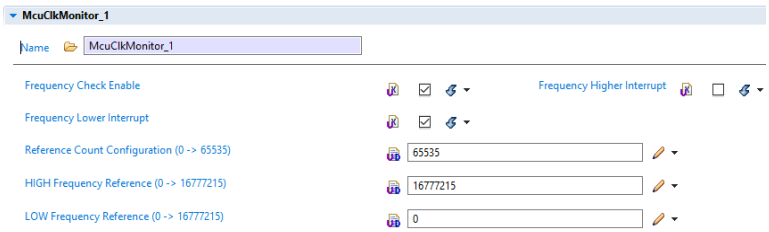


Figure 4-19. Tresos Plugin snapshot for McuClkMonitor_1 form.

4.6.8.14.1 McuClkMonitorEn (McuClkMonitor_1)

Enables/Disables the clock monitor (CMU_GCR[FCE]).

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-133. Attribute McuClkMonitorEn (McuClkMonitor_1) detailed description

Property	Value
Label	Frequency Check Enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.14.2 McuFrequencyHigherInterrupt (McuClkMonitor_1)

Enables/Disables FHH interrupt at the module boundary (CMU1_IER[FHHIE]).

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-134. Attribute McuFrequencyHigherInterrupt (McuClkMonitor_1) detailed description

Property	Value
Label	Frequency Higher Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.14.3 McuFrequencyLowerInterrupt (McuClkMonitor_1)

Enables/Disables FLL interrupt at the module boundary.(CMU1_IER[FLLIE]).

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-135. Attribute McuFrequencyLowerInterrupt (McuClkMonitor_1) detailed description

Property	Value
Label	Frequency Lower Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.8.14.4 McuReferenceCountConfiguration (McuClkMonitor_1)

Set the CMU0: CMU_RCCR[REF_CNT] field register. CMU_RCCR[REF_CNT] is the number of reference clock counts that the frequency check runs. This field defines the duration of one frequency check window.

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-136. Attribute McuReferenceCountConfiguration (McuClkMonitor_1) detailed description

Property	Value
Label	Reference Count Configuration
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	65535

4.6.8.14.5 McuHighFrequencyRef (McuClkMonitor_1)

The value of the Max Frequency.

CMU0 detects if the monitored clock is greater than McuHighFrequencyRef. The HCTR[HFREF] value is determined by the equation:

((fMonitored_clock / fReference_clock) * RCCR[REF_CNT]) + high threshold margin

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-137. Attribute McuHighFrequencyRef (McuClkMonitor_1) detailed description

Property	Value
Label	HIGH Frequency Reference
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	16777215

4.6.8.14.6 McuLowFrequencyRef (McuClkMonitor_1)

The value of the Min Frequency. CMU0 detects if the monitored clock is smaller than McuLowFrequencyRef. LCTR[LFREF] determines the low reference value for the monitored clock frequency. The LCTR[LFREF] value is determined by the equation: fMonitored_clock / fReference_clock * RCCR[REF_CNT] - low threshold margin. Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

Table 4-138. Attribute McuLowFrequencyRef (McuClkMonitor_1) detailed description

Property	Value
Label	LOW Frequency Reference
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	0

4.6.8.15 Form McuClockReferencePoint

This container defines a reference point in the Mcu Clock tree. It defines the frequency which then can be used by other modules as an input value. Lower multiplictiy is 1, as even in the simpliest case (only one frequency is used), there is one frequency to be defined.

Is included by form : Form McuClockSettingConfig

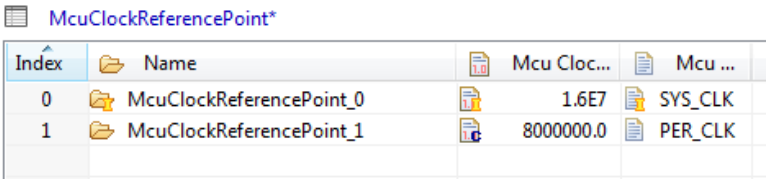


Figure 4-20. Tresos Plugin snapshot for McuClockReferencePoint form.

4.6.8.15.1 McuClockReferencePointFrequency (McuClockReferencePoint)

This is the frequency for the specific instance of the McuClockReferencePoint container. It shall be given in Hz. Calculated value.

Table 4-139. Attribute McuClockReferencePointFrequency (McuClockReferencePoint) detailed description

Property	Value
Label	Mcu Clock Reference Point Frequency
Type	FLOAT
Origin	AUTOSAR_ECUC
Symbolic Name	false
Invalid	Range <=320000000 >=0

4.6.8.15.2 McuClockFrequencySelect (McuClockReferencePoint)

Select clock source for the specific instance of the McuClockReferencePoint container.

Note: The clock frequency configured in McuPeripheralClockConfig should be used to export the clock frequency through McuClockReferencePoint. This reference point should be used in the configuration of the module that uses it (SPI, I2C, GPT, etc.). If the configured module has also an internal clock selection (like FlexTimer for example), the clock reference point should be configured taking the internal clock selection into account and the reference used should reflect the clock that finally enters the used peripheral.

Table 4-140. Attribute McuClockFrequencySelect (McuClockReferencePoint) detailed description

Property	Value
Label	Mcu Clock Frequency Select
Type	ENUMERATION
Origin	Custom

Table continues on the next page...

Table 4-140. Attribute McuClockFrequencySelect (McuClockReferencePoint) detailed description (continued)

Property	Value
Symbolic Name	false
Default	RUN_SYS_CLK
Range	CUSTOM FIRC_CLK FIRCDIV2_CLK FIRCDIV1_CLK SIRC_CLK SIRCDIV2_CLK SIRCDIV1_CLK SOSC_CLK SOSCDIV2_CLK SOSCDIV1_CLK LPO_CLK_128K LPO_CLK_32K LPO_CLK_1K SPLL_CLK SPLLDIV2_CLK SPLLDIV1_CLK RUN_PRI_DIV_SYS_CLK RUN_CORE_CLK RUN_SYS_CLK RUN_FLASH_CLK RUN_BUS_CLK HSRUN_PRI_DIV_SYS_CLK HSRUN_CORE_CLK HSRUN_SYS_CLK HSRUN_FLASH_CLK HSRUN_BUS_CLK VLPR_PRI_DIV_SYS_CLK VLPR_CORE_CLK VLPR_SYS_CLK VLPR_FLASH_CLK VLPR_BUS_CLK FTM3_CLK ADC1_CLK LPSPI0_CLK LPSPI1_CLK LPSPI2_CLK LPIT_CLK FTM0_CLK FTM1_CLK FTM2_CLK ADC0_CLK LPTMR0_CLK FLEXIO_CLK LPI2C0_CLK LPI2C1_CLK LPUART0_CLK LPUART1_CLK LPUART2_CLK FTM4_CLK FTM5_CLK FTM6_CLK

Table 4-140. Attribute McuClockFrequencySelect (McuClockReferencePoint) detailed description

Property	Value
	FTM7_CLK ENET_CLK

4.6.9 Form McuDemEventParameterRefs

Container for the references to DemEventParameter elements which shall be invoked using the API Dem_ReportErrorStatus API in case the corresponding error occurs.

The EventId is taken from the referenced DemEventParameter's DemEventId value. The standardized errors are provided in the container and can be extended by vendor specific error references.

Is included by form : Form McuModuleConfiguration

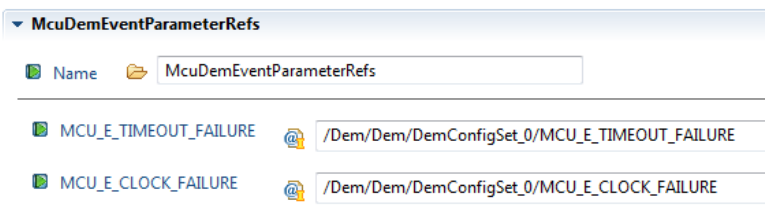


Figure 4-21. Tresos Plugin snapshot for McuDemEventParameterRefs form.

4.6.9.1 MCU_E_TIMEOUT_FAILURE (McuDemEventParameterRefs)

Reference to configured DEM event to report Timeout failure.

Table 4-141. Attribute MCU_E_TIMEOUT_FAILURE (McuDemEventParameterRefs) detailed description

Property	Value
Type	SYMBOLIC-NAME-REFERENCE
Origin	Custom

4.6.9.2 MCU_E_CLOCK_FAILURE (McuDemEventParameterRefs)

Reference to configured DEM event to report Clock source failure.

Table 4-142. Attribute MCU_E_CLOCK_FAILURE (McuDemEventParameterRefs) detailed description

Property	Value
Type	SYMBOLIC-NAME-REFERENCE
Origin	AUTOSAR_ECUC

4.6.10 Form McuInterruptEvents

Configuration for different interrupts handled by MCU. Note: Implementation specific Container.

Is included by form : Form McuModuleConfiguration

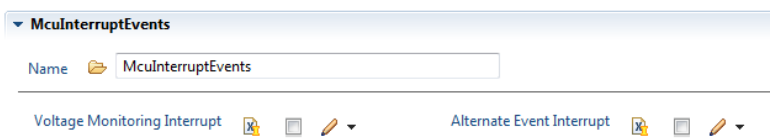


Figure 4-22. Tresos Plugin snapshot for McuInterruptEvents form.

4.6.10.1 McuVoltageErrorEvent (McuInterruptEvents)

Power Management Unit Fault Monitoring Interrupts. Note: Implementation Specific Parameter.

Table 4-143. Attribute McuVoltageErrorEvent (McuInterruptEvents) detailed description

Property	Value
Label	Voltage Monitoring Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.10.2 McuAlternateResetEvent (McuInterruptEvents)

Some events can generate an interrupt from RCM.

Note: Implementation Specific Parameter.

Table 4-144. Attribute McuAlternateResetEvent (McuInterruptEvents) detailed description

Property	Value
Label	Alternate Event Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.11 Form McuModeSettingConf

This container contains the configuration for the Mode setting of the MCU. Note: Implementation Specific Parameter.

Is included by form : Form McuModuleConfiguration

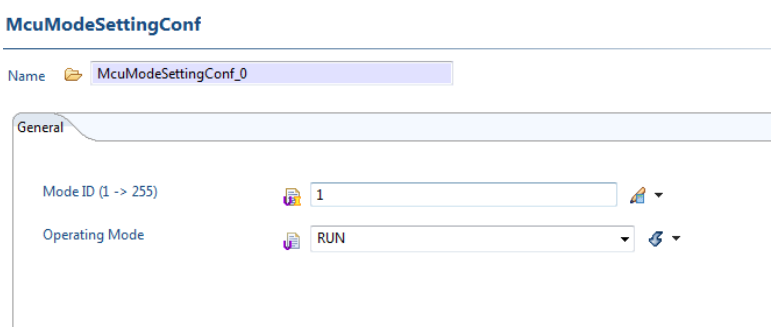


Figure 4-23. Tresos Plugin snapshot for McuModeSettingConf form.

4.6.11.1 McuMode (McuModeSettingConf)

This parameter shall represent the ID of the MCU mode.

Table 4-145. Attribute McuMode (McuModeSettingConf) detailed description

Property	Value
Label	Mode ID
Type	INTEGER

Table continues on the next page...

Table 4-145. Attribute McuMode (McuModeSettingConf) detailed description (continued)

Property	Value
Origin	AUTOSAR_ECUC
Symbolic Name	true
Invalid	Range <=255 >=0

4.6.11.2 McuPowerMode (McuModeSettingConf)

This parameter selects the Power Mode to be used. For valid Mode transitions refers to Power mode state diagram from Reference Manual. Note: Implementation Specific Parameter.

Table 4-146. Attribute McuPowerMode (McuModeSettingConf) detailed description

Property	Value
Label	Operating Mode
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	RUN
Range	RUN HSRUN VLPR VLPS STOP1 STOP2

4.6.12 Form McuPowerControl

Note: Implementation Specific Parameter.

Is included by form : Form McuModuleConfiguration

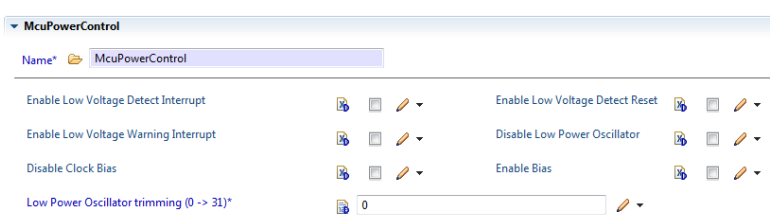


Figure 4-24. Tresos Plugin snapshot for McuPowerControl form.

4.6.12.1 McuLowVoltageDetectInterruptEnable (McuPowerControl)

PMC_LVDSC1[LVDIE] - Low Voltage Detect Interrupt Enable.

This bit enables hardware interrupt requests for LVDF.

0 - Hardware interrupt disabled (use polling).

1 - Request a hardware interrupt when LVDF = 1.

Note: Implementation Specific Parameter.

Table 4-147. Attribute McuLowVoltageDetectInterruptEnable (McuPowerControl) detailed description

Property	Value
Label	Enable Low Voltage Detect Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.12.2 McuLowVoltageDetectResetEnable (McuPowerControl)

PMC_LVDSC1[LVDRE] - Low Voltage Detect Reset Enable.

This bit enables the low voltage detect events to generate a system reset.

0 - No system resets on low voltage detect events.

1 - If the supply voltage falls below VLVD, a system reset will be generated.

Note: Implementation Specific Parameter.

Table 4-148. Attribute McuLowVoltageDetectResetEnable (McuPowerControl) detailed description

Property	Value
Label	Enable Low Voltage Detect Reset
Type	BOOLEAN
Origin	Custom

Table continues on the next page...

Table 4-148. Attribute McuLowVoltageDetectResetEnable (McuPowerControl) detailed description (continued)

Property	Value
Symbolic Name	false

4.6.12.3 McuLowVoltageWarningInterruptEnable (McuPowerControl)

PMC_LVDSC2[LVWIE] - Low-Voltage Warning Interrupt Enable.

This bit enables hardware interrupt requests for LVWF.

0 - Hardware interrupt disabled (use polling).

1 - Request a hardware interrupt when LVWF = 1.

Note: Implementation Specific Parameter.

Table 4-149. Attribute McuLowVoltageWarningInterruptEnable (McuPowerControl) detailed description

Property	Value
Label	Enable Low Voltage Warning Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.12.4 McuLPODisable (McuPowerControl)

PMC_REGSC[LPODIS] - This bit enables or disable the low power oscillator.

After disabling the LPO a time of 2 LPO clock cycles is required before it is allowed to enable it

again. Violating this waiting time of 2 cycles can result in malfunction of the LPO.

unchecked - Low power oscillator enabled.

checked - Low power oscillator disabled.

Note: Implementation Specific Parameter.

Table 4-150. Attribute McuLPODisable (McuPowerControl) detailed description

Property	Value
Label	Disable Low Power Oscillator
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.12.5 McuClockBiasDisable (McuPowerControl)

PMC_REGSC[CLKBIASDIS] - Clock Bias Disable Bit.

This bit disables the bias currents and reference voltages for some clock modules in order to further reduce power consumption in VLPS mode.

Note: While using this bit, it must be ensured that respective clock modules are disabled in VLPS mode. Else, severe malfunction of clock modules will happen.

unchecked - No effect.

checked - In VLPS mode, the bias currents and reference voltages for the following clock modules are disabled: SIRC, FIRC, PLL.

Note: Implementation Specific Parameter.

Table 4-151. Attribute McuClockBiasDisable (McuPowerControl) detailed description

Property	Value
Label	Disable Clock Bias
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.12.6 McuLowPowerBiasEnable (McuPowerControl)

PMC_REGSC[BIASEN] - Bias Enable Bit.

This bit enables source and well biasing for the core logic in low power mode. In full performance mode this bit has no effect. This is useful to further reduce MCU power consumption in low power mode.

unchecked - Biasing disabled, core logic can run in full performance.

checked - Biasing enabled, core logic is slower and there are restrictions in allowed system clock speed.

Note: Implementation Specific Parameter.

Table 4-152. Attribute McuLowPowerBiasEnable (McuPowerControl) detailed description

Property	Value
Label	Enable Bias
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.12.7 McuLpoTrimming (McuPowerControl)

Configures the PMC_LPOTRIM[LPOTRIM] bitfield.

These bits are used for trimming the frequency of the low power oscillator:

10000 - Period of LPO clock is lowest

10001 to 11111 - Period of LPO clock is increasing

00000 - Period of LPO clock is typical 128 kHz

00001 to 01110 - Period of LPO clock is increasing

01111 - Period of LPO clock is highest

Note: Implementation Specific Parameter.

Table 4-153. Attribute McuLpoTrimming (McuPowerControl) detailed description

Property	Value
Label	Low Power Oscillator trimming
Type	INTEGER
Origin	Custom
Symbolic Name	false
Default	0
Invalid	Range <=31 >=0

4.6.13 Form McuRamSectorSettingConf

This container contains the configuration for the RAM Sector setting.

Is included by form : Form McuModuleConfiguration

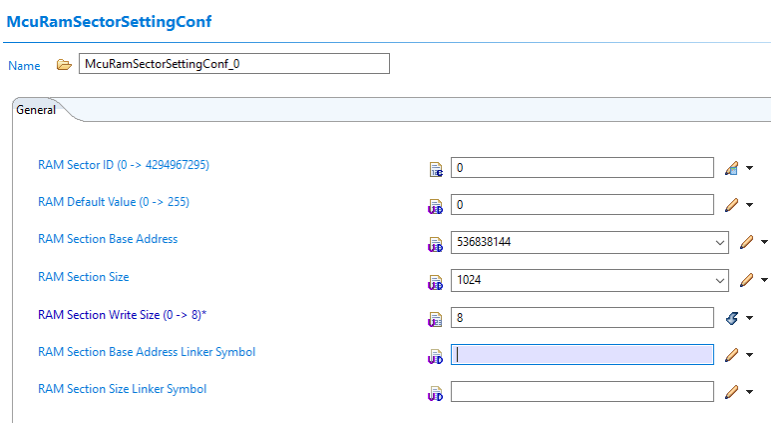


Figure 4-25. Tresos Plugin snapshot for McuRamSectorSettingConf form.

4.6.13.1 McuRamSectorId (McuRamSectorSettingConf)

This parameter shall represent the ID of the MCU RAM Sector configuration.

Table 4-154. Attribute McuRamSectorId (McuRamSectorSettingConf) detailed description

Property	Value
Label	RAM Sector ID
Type	INTEGER
Origin	Custom
Symbolic Name	true

Table continues on the next page...

Table 4-154. Attribute McuRamSectorId (McuRamSectorSettingConf) detailed description (continued)

Property	Value
Invalid	XPath Range <=4294967295 >=0

4.6.13.2 McuRamDefaultValue (McuRamSectorSettingConf)

This parameter shall represent the Data pre-setting to be initialized. Default value is 0xbabababa. Note: Implementation Specific Parameter.

Table 4-155. Attribute McuRamDefaultValue (McuRamSectorSettingConf) detailed description

Property	Value
Label	RAM Default Value
Type	INTEGER
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	0
Invalid	Range <=255 >=0

4.6.13.3 McuRamSectionBaseAddress (McuRamSectorSettingConf)

This parameter represents the RAM section base address. The address must be aligned to 4 bytes. Note: Implementation Specific Parameter.

Table 4-156. Attribute McuRamSectionBaseAddress (McuRamSectorSettingConf) detailed description

Property	Value
Label	RAM Section Base Address
Type	INTEGER
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	536869888
Invalid	Range

Table 4-156. Attribute McuRamSectionBaseAddress (McuRamSectorSettingConf) detailed description

Property	Value
	<ecu:get('MCU.McuModuleConfiguration.McuRamSectorSettingConf.McuRamSectionBaseAddress.high') >=ecu:get('MCU.McuModuleConfiguration.McuRamSectorSettingConf.McuRamSectionBaseAddress.low')

4.6.13.4 McuRamSectionWriteSize (McuRamSectorSettingConf)

This parameter shall define the size in bytes of data which can be written into RAM at once.The ram write size is currently restricted to {1, 2, 4, 8} bytes. Note: Implementation Specific Parameter.

Table 4-157. Attribute McuRamSectionWriteSize (McuRamSectorSettingConf) detailed description

Property	Value
Label	RAM Section Write Size
Type	INTEGER
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	8
Invalid	Range <=8 >=0

4.6.13.5 McuRamSectionSize (McuRamSectorSettingConf)

This parameter represents the RAM section size in bytes. The size must be multiple of 4. Note: Implementation Specific Parameter.

Table 4-158. Attribute McuRamSectionSize (McuRamSectorSettingConf) detailed description

Property	Value
Label	RAM Section Size
Type	INTEGER
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	1024

Table continues on the next page...

Table 4-158. Attribute McuRamSectionSize (McuRamSectorSettingConf) detailed description (continued)

Property	Value
Invalid	Range XPath <=ecu:get('MCU.McuModuleConfiguration.McuRamSectorSettingConf.McuRamSectionSize') >=0

4.6.13.6 McuRamSectionBaseAddrLinkerSym (McuRamSectorSettingConf)

This parameter represents the RAM section base address. The address must be aligned to 4 bytes. If this parameter is empty, then the integer values from "McuRamSectionBaseAddress" will be used. Note: Implementation Specific Parameter.

Table 4-159. Attribute McuRamSectionBaseAddrLinkerSym (McuRamSectorSettingConf) detailed description

Property	Value
Label	RAM Section Base Address Linker Symbol
Type	STRING
Origin	Custom
Symbolic Name	false
Default	

4.6.13.7 McuRamSectionSizeLinkerSym (McuRamSectorSettingConf)

This parameter represents the RAM section size in bytes. The size must be multiple of 4. If this parameter is empty, then the integer values from "McuRamSectionSize" will be used. Note: Implementation Specific Parameter.

Table 4-160. Attribute McuRamSectionSizeLinkerSym (McuRamSectorSettingConf) detailed description

Property	Value
Label	RAM Section Size Linker Symbol
Type	STRING
Origin	Custom
Symbolic Name	false
Default	

4.6.14 Form McuResetConfig

The Reset Control Module (MC_RCM) centralizes the different reset sources and manages the reset sequence of the device. Note: Implementation Specific Parameter.

Is included by form : Form McuModuleConfiguration

Included forms :

- Form McuSystemInterruptEnable

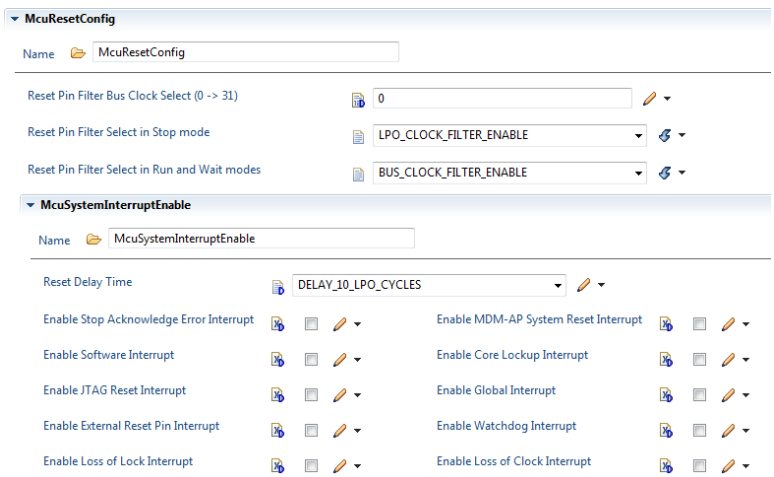


Figure 4-26. Tresos Plugin snapshot for McuResetConfig form.

4.6.14.1 McuResetPinFilterBusClockSelect (McuResetConfig)

RCM_RPC[RSTFLTSEL] - Reset Pin Filter Bus Clock Select.

Selects the reset pin bus clock filter width. Transitions for less than (RSTFLTSEL+1) bus clock cycles are always filtered, transitions equal to (RSTFLTSEL+1) bus clock cycles may be filtered.

Note: Implementation Specific Parameter.

Table 4-161. Attribute McuResetPinFilterBusClockSelect (McuResetConfig) detailed description

Property	Value
Label	Reset Pin Filter Bus Clock Select
Type	INTEGER
Origin	Custom

Table continues on the next page...

Table 4-161. Attribute McuResetPinFilterBusClockSelect (McuResetConfig) detailed description (continued)

Property	Value
Symbolic Name	false
Default	0

4.6.14.2 McuResetPinFilterInStopMode (McuResetConfig)

RCM_RPC[RSTFLTSS] - Reset Pin Filter Select in Stop Mode.

Selects how the reset pin filter is enabled in any stop mode.

0 - All filtering disabled.

1 - LPO clock filter enabled.

Note: Implementation Specific Parameter.

Table 4-162. Attribute McuResetPinFilterInStopMode (McuResetConfig) detailed description

Property	Value
Label	Reset Pin Filter Select in Stop mode
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	ALL_FILTERING_DISABLE
Range	ALL_FILTERING_DISABLE LPO_CLOCK_FILTER_ENABLE

4.6.14.3 McuResetPinFilterInRunAndWait (McuResetConfig)

RCM_RPC[RSTFLTSRW] - Reset Pin Filter Select in Run and Wait Modes.

Selects how the reset pin filter is enabled in run and wait modes.

0 - All filtering disabled.

1 - Bus clock filter enabled for normal operation.

2 - LPO clock filter enabled for normal operation.

Note: Implementation Specific Parameter.

Table 4-163. Attribute McuResetPinFilterInRunAndWait (McuResetConfig) detailed description

Property	Value
Label	Reset Pin Filter Select in Run and Wait modes
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	ALL_FILTERING_DISABLE
Range	ALL_FILTERING_DISABLE BUS_CLOCK_FILTER_ENABLE LPO_CLOCK_FILTER_ENABLE

4.6.14.4 Form McuSystemInterruptEnable

Configures RCM_SRIE

This registers delays the assertion of a system reset for a period of time (DELAY field) while an interrupt is generated.

This allows software to perform a graceful shutdown.

A Chip POR source cannot be delayed by this feature, and entering Stop mode will terminate the delay.

The SRS will only update after the system reset occurs.

Note: Implementation specific Container.

Is included by form : Form McuResetConfig

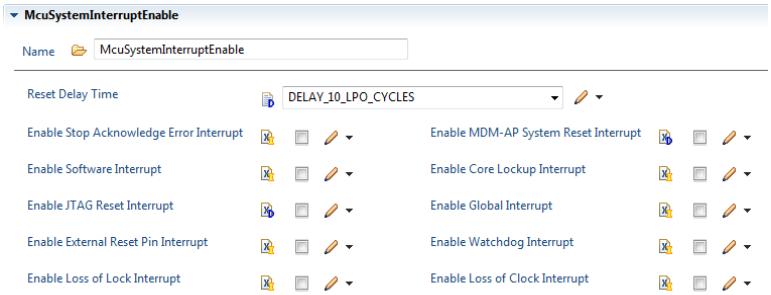


Figure 4-27. Tresos Plugin snapshot for McuSystemInterruptEnable form.

4.6.14.4.1 McuResetDelayTime (McuSystemInterruptEnable)

RCM_SRIE[DELAY] - Reset Delay Time.

Configures the maximum reset delay time from when the interrupt is asserted and the system reset occurs.

0 - 10 LPO cycles.

1 - 34 LPO cycles.

2 - 130 LPO cycles.

2 - 514 LPO cycles.

Note: Implementation Specific Parameter.

Table 4-164. Attribute McuResetDelayTime (McuSystemInterruptEnable) detailed description

Property	Value
Label	Reset Delay Time
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	DELAY_10_LPO_CYCLES
Range	DELAY_10_LPO_CYCLES DELAY_34_LPO_CYCLES DELAY_130_LPO_CYCLES DELAY_514_LPO_CYCLES

4.6.14.4.2 McuStopAcknowledgeErrorInterrupt (McuSystemInterruptEnable)

RCM_SRIE[SACKERR] - Stop Acknowledge Error Interrupt.

0 - Interrupt disabled.

1 - Interrupt enabled.

Note: Implementation Specific Parameter.

Table 4-165. Attribute McuStopAcknowledgeErrorInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable Stop Acknowledge Error Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.14.4.3 McuMDMAPSystemResetInterrupt (McuSystemInterruptEnable)

RCM_SRIE[MDM_AP] - MDM-AP System Reset Request.

0 - Interrupt disabled.

1 - Interrupt enabled.

Note: Implementation Specific Parameter.

Table 4-166. Attribute McuMDMAPSystemResetInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable MDM-AP System Reset Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.14.4.4 McuSoftwareInterrupt (McuSystemInterruptEnable)

RCM_SRIE[SW] - Software Interrupt.

0 - Interrupt disabled.

1 - Interrupt enabled.

Note: Implementation Specific Parameter.

Table 4-167. Attribute McuSoftwareInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable Software Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.14.4.5 McuCoreLockupInterrupt (McuSystemInterruptEnable)

RCM_SRIE[LOCKUP] - Core Lockup Interrupt.

0 - Interrupt disabled.

1 - Interrupt enabled.

Note: Implementation Specific Parameter.

Table 4-168. Attribute McuCoreLockupInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable Core Lockup Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.14.4.6 McuJTAGResetInterrupt (McuSystemInterruptEnable)

RCM_SRIE[JTAG] - JTAG generated reset.

0 - Interrupt disabled.

1 - Interrupt enabled.

Note: Implementation Specific Parameter.

Table 4-169. Attribute McuJTAGResetInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable JTAG Reset Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.14.4.7 McuGlobalInterrupt (McuSystemInterruptEnable)

RCM_SRIE[GIE] - Global Interrupt Enable.

0 - All interrupt sources disabled.

1 - All interrupt sources enabled.

Note: Implementation Specific Parameter.

Table 4-170. Attribute McuGlobalInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable Global Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.14.4.8 McuExternalResetPinInterrupt (McuSystemInterruptEnable)

RCM_SRIE[PIN] - External Reset Pin Interrupt.

0 - Interrupt disabled.

1 - Interrupt enabled.

Note: Implementation Specific Parameter.

Table 4-171. Attribute McuExternalResetPinInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable External Reset Pin Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.14.4.9 McuWatchdogInterrupt (McuSystemInterruptEnable)

RCM_SRIE[WDOG] - Watchdog Interrupt.

0 - Interrupt disabled.

1 - Interrupt enabled.

Note: Implementation Specific Parameter.

Table 4-172. Attribute McuWatchdogInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable Watchdog Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.14.4.10 McuCMULossOfClockResetInterrupt (McuSystemInterruptEnable)

RCM_SRIE[CMU_LOC] - CMU Loss-of-Clock Reset Interrupt.

0 - Interrupt disabled.

1 - Interrupt enabled.

Note: Implementation Specific Parameter.

Table 4-173. Attribute McuCMULossOfClockResetInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable CMU Loss of Clock Reset Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.14.4.11 McuLossOfLockInterrupt (McuSystemInterruptEnable)

RCM_SRIE[LOL] - Loss of Lock Interrupt.

0 - Interrupt disabled.

1 - Interrupt enabled.

Note: Implementation Specific Parameter.

Table 4-174. Attribute McuLossOfLockInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable Loss of Lock Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.14.4.12 McuLossOfClockInterrupt (McuSystemInterruptEnable)

RCM_SRIE[LOC] - Loss of Clock Interrupt.

0 - Interrupt disabled.

1 - Interrupt enabled.

Note: Implementation Specific Parameter.

Table 4-175. Attribute McuLossOfClockInterrupt (McuSystemInterruptEnable) detailed description

Property	Value
Label	Enable Loss of Clock Interrupt
Type	BOOLEAN
Origin	Custom
Symbolic Name	false

4.6.15 Form McuSIMConfig

Configures SIM_CHIPCTL, SIM_FCFG1, SIM_FTMOPT0, SIM_FTMOPT1, SIM_ADCOPT and SIM_MISCTRL registers.

Note: Implementation specific Container.

Is included by form : Form McuModuleConfiguration

Included forms :

- Form McuChipControlConfiguration
- Form McuFlexTimerConfiguration
- Form McuAdcOptionsConfiguration

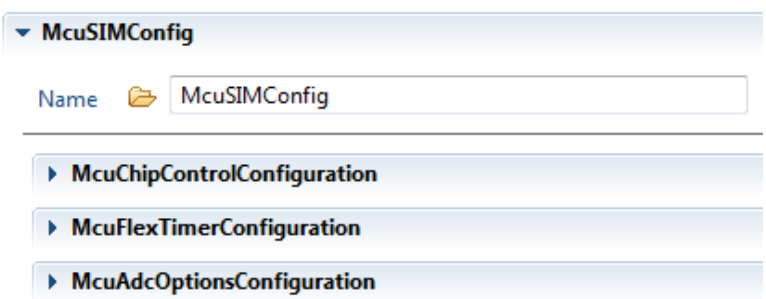


Figure 4-28. Tresos Plugin snapshot for McuSIMConfig form.

4.6.15.1 Form McuChipControlConfiguration

This container contains the configuration for the SIM_CHIPCTL register.

Is included by form : Form McuSIMConfig

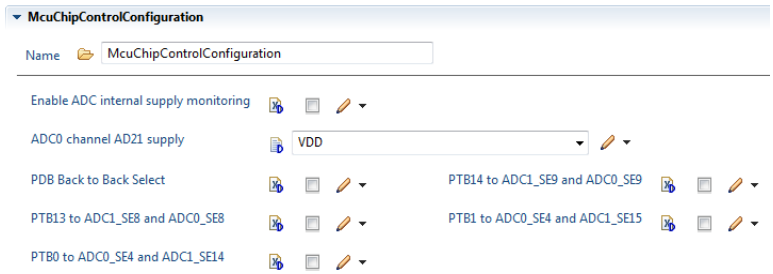


Figure 4-29. Tresos Plugin snapshot for McuChipControlConfiguration form.

4.6.15.1.1 McuEnableAdcSupplyMonitoring (McuChipControlConfiguration)

SIM_CHIPCTL[ADC_SUPPLYEN] - Enable for internal supply monitoring on ADC0 channel AD21.

unchecked - Disable internal supply monitoring.

checked - Enable internal supply monitoring.

Note: Implementation Specific Parameter.

Table 4-176. Attribute McuEnableAdcSupplyMonitoring (McuChipControlConfiguration) detailed description

Property	Value
Label	Enable ADC internal supply monitoring
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.1.2 McuAdcSupply (McuChipControlConfiguration)

SIM_CHIPCTL[ADC_SUPPLY] - Internal supplies monitored on ADC0 channel AD21.

0 - 5 V input VDD supply (VDD)

1 - 5 V input analog supply (VDDA)

2 - ADC Reference Supply (VREFH)

3 - 3.3 V Oscillator Regulator Output (VDD_3V)

4 - 3.3 V flash regulator output (VDD_flash_3V)

5 - 1.2 V core regulator output (VDD_LV)

Note: Implementation Specific Parameter.

Table 4-177. Attribute McuAdcSupply (McuChipControlConfiguration) detailed description

Property	Value
Label	ADC0 channel AD21 supply
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	VDD
Range	VDD VDDA VREFH VDD_3V VDD_FLASH_3V VDD_LV

4.6.15.1.3 McuPDBBackToBackSelect (McuChipControlConfiguration)

SIM_CHIPCTL[PDB_BB_SEL] - PDB back-to-back select

Selects ADC COCO source as pdb back-to-back mode.

unchecked - PDB0 channel 0 back-to-back operation with ADC0 COCO[7:0]; PDB1 channel 0 back-to-back operation with ADC1 COCO[7:0]; PDB2 channel 0 back-to-back operation with ADC2 COCO[7:0].

checked - Channel 0 of PDB0,PDB1 back-to-back operation with COCO[7:0] of ADC0, ADC1

Note: Implementation Specific Parameter.

Table 4-178. Attribute McuPDBBackToBackSelect (McuChipControlConfiguration) detailed description

Property	Value
Label	PDB Back to Back Select
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.1.4 McuPTB14InterleaveChannelSelect (McuChipControlConfiguration)

SIM_CHIPCTL[INTERLEAVE_SEL] - ADC interleave channel select

Select ADC interleave pins.

1 - PTB14 to ADC1_SE9 and ADC0_SE9

Note: Implementation Specific Parameter.

Table 4-179. Attribute McuPTB14InterleaveChannelSelect (McuChipControlConfiguration) detailed description

Property	Value
Label	PTB14 to ADC1_SE9 and ADC0_SE9
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.1.5 McuPTB13InterleaveChannelSelect (McuChipControlConfiguration)

SIM_CHIPCTL[INTERLEAVE_SEL] - ADC interleave channel select

Select ADC interleave pins.

1 - PTB13 to ADC1_SE8 and ADC0_SE8

Note: Implementation Specific Parameter.

Table 4-180. Attribute McuPTB13InterleaveChannelSelect (McuChipControlConfiguration) detailed description

Property	Value
Label	PTB13 to ADC1_SE8 and ADC0_SE8
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.1.6 McuPTB1InterleaveChannelSelect (McuChipControlConfiguration)

SIM_CHIPCTL[INTERLEAVE_SEL] - ADC interleave channel select

Select ADC interleave pins.

1 - PTB1 to ADC0_SE4 and ADC1_SE15

Note: Implementation Specific Parameter.

Table 4-181. Attribute McuPTB1InterleaveChannelSelect (McuChipControlConfiguration) detailed description

Property	Value
Label	PTB1 to ADC0_SE4 and ADC1_SE15
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.1.7 McuPTB0InterleaveChannelSelect (McuChipControlConfiguration)

SIM_CHIPCTL[INTERLEAVE_SEL] - ADC interleave channel select

Select ADC interleave pins.

1 - PTB0 to ADC0_SE4 and ADC1_SE14

Note: Implementation Specific Parameter.

Table 4-182. Attribute McuPTB0InterleaveChannelSelect (McuChipControlConfiguration) detailed description

Property	Value
Label	PTB0 to ADC0_SE4 and ADC1_SE14
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2 Form McuFlexTimerConfiguration

This container contains the configuration for the SIM_FTMOPT0 and FTMOPT1 registers.

Is included by form : Form McuSIMConfig

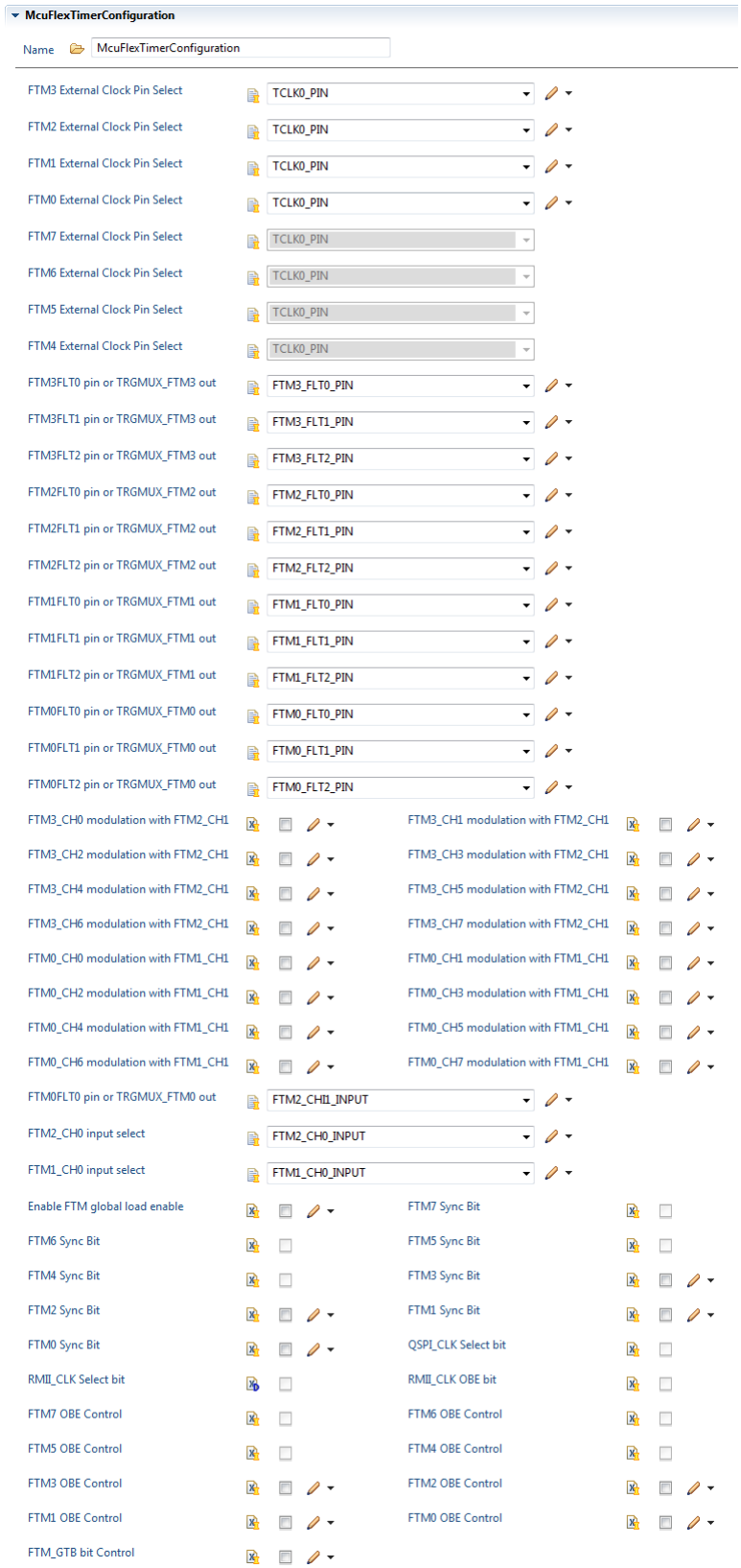


Figure 4-30. Tresos Plugin snapshot for McuFlexTimerConfiguration form.

4.6.15.2.1 McuFTM3ExternalClockPinSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM3CLKSEL] - FTM3 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM3 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM3 external clock driven by TCLK0 pin.

1 - FTM3 external clock driven by TCLK1 pin.

2 - FTM3 external clock driven by TCLK2 pin.

3 - No clock input.

Note: Implementation Specific Parameter.

Table 4-183. Attribute McuFTM3ExternalClockPinSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3 External Clock Pin Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	TCLK0_PIN
Range	TCLK0_PIN TCLK1_PIN TCLK2_PIN NO_CLOCK

4.6.15.2.2 McuFTM2ExternalClockPinSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM2CLKSEL] - FTM2 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM2 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM2 external clock driven by TCLK0 pin.

1 - FTM2 external clock driven by TCLK1 pin.

2 - FTM2 external clock driven by TCLK2 pin.

3 - No clock input.

Note: Implementation Specific Parameter.

Table 4-184. Attribute McuFTM2ExternalClockPinSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM2 External Clock Pin Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	TCLK0_PIN
Range	TCLK0_PIN TCLK1_PIN TCLK2_PIN NO_CLOCK

4.6.15.2.3 McuFTM1ExternalClockPinSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM1CLKSEL] - FTM1 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM1 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM1 external clock driven by TCLK0 pin.

1 - FTM1 external clock driven by TCLK1 pin.

2 - FTM1 external clock driven by TCLK2 pin.

3 - No clock input.

Note: Implementation Specific Parameter.

Table 4-185. Attribute McuFTM1ExternalClockPinSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM1 External Clock Pin Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	TCLK0_PIN
Range	TCLK0_PIN TCLK1_PIN TCLK2_PIN NO_CLOCK

4.6.15.2.4 McuFTM0ExternalClockPinSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM0CLKSEL] - FTM0 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM0 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM0 external clock driven by TCLK0 pin.

1 - FTM0 external clock driven by TCLK1 pin.

2 - FTM0 external clock driven by TCLK2 pin.

3 - No clock input.

Note: Implementation Specific Parameter.

Table 4-186. Attribute McuFTM0ExternalClockPinSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0 External Clock Pin Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false

Table continues on the next page...

Table 4-186. Attribute McuFTM0ExternalClockPinSelect (McuFlexTimerConfiguration) detailed description (continued)

Property	Value
Default	TCLK0_PIN
Range	TCLK0_PIN TCLK1_PIN TCLK2_PIN NO_CLOCK

4.6.15.2.5 McuFTM7ExternalClockPinSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM7CLKSEL] - FTM7 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM7 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM7 external clock driven by TCLK0 pin.

1 - FTM7 external clock driven by TCLK1 pin.

2 - FTM7 external clock driven by TCLK2 pin.

3 - No clock input.

Note: Implementation Specific Parameter.

Table 4-187. Attribute McuFTM7ExternalClockPinSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM7 External Clock Pin Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	TCLK0_PIN
Range	TCLK0_PIN TCLK1_PIN TCLK2_PIN NO_CLOCK

4.6.15.2.6 McuFTM6ExternalClockPinSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM6CLKSEL] - FTM6 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM6 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM6 external clock driven by TCLK0 pin.

1 - FTM6 external clock driven by TCLK1 pin.

2 - FTM6 external clock driven by TCLK2 pin.

3 - No clock input.

Note: Implementation Specific Parameter.

Table 4-188. Attribute McuFTM6ExternalClockPinSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM6 External Clock Pin Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	TCLK0_PIN
Range	TCLK0_PIN TCLK1_PIN TCLK2_PIN NO_CLOCK

4.6.15.2.7 McuFTM5ExternalClockPinSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM5CLKSEL] - FTM5 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM5 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM5 external clock driven by TCLK0 pin.

1 - FTM5 external clock driven by TCLK1 pin.

2 - FTM5 external clock driven by TCLK2 pin.

3 - No clock input.

Note: Implementation Specific Parameter.

Table 4-189. Attribute McuFTM5ExternalClockPinSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM5 External Clock Pin Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	TCLK0_PIN
Range	TCLK0_PIN TCLK1_PIN TCLK2_PIN NO_CLOCK

4.6.15.2.8 McuFTM4ExternalClockPinSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM7CLKSEL] - FTM4 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM4 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM4 external clock driven by TCLK0 pin.

1 - FTM4 external clock driven by TCLK1 pin.

2 - FTM4 external clock driven by TCLK2 pin.

3 - No clock input.

Note: Implementation Specific Parameter.

Table 4-190. Attribute McuFTM4ExternalClockPinSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM4 External Clock Pin Select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	TCLK0_PIN
Range	TCLK0_PIN TCLK1_PIN TCLK2_PIN NO_CLOCK

4.6.15.2.9 McuFTM3Fault0Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM3FLTxSEL] - FTM3 Fault 0 Select

Selects the source of FTM3 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM3_FLT0 pin

1 - TRGMUX_FTM3 out

Note: Implementation Specific Parameter.

Table 4-191. Attribute McuFTM3Fault0Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3FLT0 pin or TRGMUX_FTM3 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM3_FLT0_PIN
Range	FTM3_FLT0_PIN TRGMUX_FTM3_OUT

4.6.15.2.10 McuFTM3Fault1Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM3FLTxSEL] - FTM3 Fault 1 Select

Selects the source of FTM3 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM3_FLT1 pin

1 - TRGMUX_FTM3 out

Note: Implementation Specific Parameter.

Table 4-192. Attribute McuFTM3Fault1Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3FLT1 pin or TRGMUX_FTM3 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM3_FLT1_PIN
Range	FTM3_FLT1_PIN TRGMUX_FTM3_OUT

4.6.15.2.11 McuFTM3Fault2Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM3FLTxSEL] - FTM3 Fault 2 Select

Selects the source of FTM3 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM3_FLT0 pin

1 - TRGMUX_FTM3 out

Note: Implementation Specific Parameter.

Table 4-193. Attribute McuFTM3Fault2Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3FLT2 pin or TRGMUX_FTM3 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM3_FLT2_PIN
Range	FTM3_FLT2_PIN TRGMUX_FTM3_OUT

4.6.15.2.12 McuFTM2Fault0Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM2FLTxSEL] - FTM2 Fault 0 Select

Selects the source of FTM2 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM2_FLT0 pin

1 - TRGMUX_FTM2 out

Note: Implementation Specific Parameter.

Table 4-194. Attribute McuFTM2Fault0Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM2FLT0 pin or TRGMUX_FTM2 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM2_FLT0_PIN
Range	FTM2_FLT0_PIN TRGMUX_FTM2_OUT

4.6.15.2.13 McuFTM2Fault1Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM2FLTxSEL] - FTM2 Fault 1 Select

Selects the source of FTM2 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM2_FLT1 pin

1 - TRGMUX_FTM2 out

Note: Implementation Specific Parameter.

Table 4-195. Attribute McuFTM2Fault1Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM2FLT1 pin or TRGMUX_FTM2 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM2_FLT1_PIN
Range	FTM2_FLT1_PIN TRGMUX_FTM2_OUT

4.6.15.2.14 McuFTM2Fault2Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM2FLTxSEL] - FTM2 Fault 2 Select

Selects the source of FTM2 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM2_FLT0 pin

1 - TRGMUX_FTM2 out

Note: Implementation Specific Parameter.

Table 4-196. Attribute McuFTM2Fault2Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM2FLT2 pin or TRGMUX_FTM2 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM2_FLT2_PIN
Range	FTM2_FLT2_PIN TRGMUX_FTM2_OUT

4.6.15.2.15 McuFTM1Fault0Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM1FLTxSEL] - FTM1 Fault 0 Select

Selects the source of FTM1 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM1_FLT0 pin

1 - TRGMUX_FTM1 out

Note: Implementation Specific Parameter.

Table 4-197. Attribute McuFTM1Fault0Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM1FLT0 pin or TRGMUX_FTM1 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM1_FLT0_PIN
Range	FTM1_FLT0_PIN TRGMUX_FTM1_OUT

4.6.15.2.16 McuFTM1Fault1Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM1FLTxSEL] - FTM1 Fault 1 Select

Selects the source of FTM1 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM1_FLT1 pin

1 - TRGMUX_FTM1 out

Note: Implementation Specific Parameter.

Table 4-198. Attribute McuFTM1Fault1Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM1FLT1 pin or TRGMUX_FTM1 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM1_FLT1_PIN
Range	FTM1_FLT1_PIN TRGMUX_FTM1_OUT

4.6.15.2.17 McuFTM1Fault2Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM1FLTxSEL] - FTM1 Fault 2 Select

Selects the source of FTM1 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM1_FLT0 pin

1 - TRGMUX_FTM1 out

Note: Implementation Specific Parameter.

Table 4-199. Attribute McuFTM1Fault2Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM1FLT2 pin or TRGMUX_FTM1 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM1_FLT2_PIN
Range	FTM1_FLT2_PIN TRGMUX_FTM1_OUT

4.6.15.2.18 McuFTM0Fault0Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM0FLTxSEL] - FTM0 Fault 0 Select

Selects the source of FTM0 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM0_FLT0 pin

1 - TRGMUX_FTM0 out

Note: Implementation Specific Parameter.

Table 4-200. Attribute McuFTM0Fault0Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0FLT0 pin or TRGMUX_FTM0 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM0_FLT0_PIN
Range	FTM0_FLT0_PIN TRGMUX_FTM0_OUT

4.6.15.2.19 McuFTM0Fault1Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM0FLTxSEL] - FTM0 Fault 1 Select

Selects the source of FTM0 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM0_FLT1 pin

1 - TRGMUX_FTM0 out

Note: Implementation Specific Parameter.

Table 4-201. Attribute McuFTM0Fault1Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0FLT1 pin or TRGMUX_FTM0 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM0_FLT1_PIN
Range	FTM0_FLT1_PIN TRGMUX_FTM0_OUT

4.6.15.2.20 McuFTM0Fault2Select (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM0FLTxSEL] - FTM0 Fault 2 Select

Selects the source of FTM0 fault. Every bit means one fault input respectively.

The pin source for fault must be configured for the FTM module fault function through the appropriate pin control register in the port control module when it comes from external fault pin.

0 - FTM0_FLT0 pin

1 - TRGMUX_FTM0 out

Note: Implementation Specific Parameter.

Table 4-202. Attribute McuFTM0Fault2Select (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0FLT2 pin or TRGMUX_FTM0 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM0_FLT2_PIN
Range	FTM0_FLT2_PIN TRGMUX_FTM0_OUT

4.6.15.2.21 McuFTM3Ch0ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM3_OUTSEL] - FTM3 channel modulation select with FTM2_CH1

0 - No modulation with FTM2_CH1

1 - Modulation with FTM2_CH1

Note: Implementation Specific Parameter.

Table 4-203. Attribute McuFTM3Ch0ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3_CH0 modulation with FTM2_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.22 McuFTM3Ch1ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM3_OUTSEL] - FTM3 channel modulation select with FTM2_CH1

0 - No modulation with FTM2_CH1

1 - Modulation with FTM2_CH1

Note: Implementation Specific Parameter.

Table 4-204. Attribute McuFTM3Ch1ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3_CH1 modulation with FTM2_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.23 McuFTM3Ch2ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM3_OUTSEL] - FTM3 channel modulation select with FTM2_CH1

0 - No modulation with FTM2_CH1

1 - Modulation with FTM2_CH1

Note: Implementation Specific Parameter.

Table 4-205. Attribute McuFTM3Ch2ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3_CH2 modulation with FTM2_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.24 McuFTM3Ch3ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM3_OUTSEL] - FTM3 channel modulation select with FTM2_CH1

0 - No modulation with FTM2_CH1

1 - Modulation with FTM2_CH1

Note: Implementation Specific Parameter.

Table 4-206. Attribute McuFTM3Ch3ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3_CH3 modulation with FTM2_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.25 McuFTM3Ch4ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM3_OUTSEL] - FTM3 channel modulation select with FTM2_CH1

0 - No modulation with FTM2_CH1

1 - Modulation with FTM2_CH1

Note: Implementation Specific Parameter.

Table 4-207. Attribute McuFTM3Ch4ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3_CH4 modulation with FTM2_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.26 McuFTM3Ch5ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM3_OUTSEL] - FTM3 channel modulation select with FTM2_CH1

0 - No modulation with FTM2_CH1

1 - Modulation with FTM2_CH1

Note: Implementation Specific Parameter.

Table 4-208. Attribute McuFTM3Ch5ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3_CH5 modulation with FTM2_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.27 McuFTM3Ch6ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM3_OUTSEL] - FTM3 channel modulation select with FTM2_CH1

0 - No modulation with FTM2_CH1

1 - Modulation with FTM2_CH1

Note: Implementation Specific Parameter.

Table 4-209. Attribute McuFTM3Ch6ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3_CH6 modulation with FTM2_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.28 McuFTM3Ch7ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM3_OUTSEL] - FTM3 channel modulation select with FTM2_CH1

0 - No modulation with FTM2_CH1

1 - Modulation with FTM2_CH1

Note: Implementation Specific Parameter.

Table 4-210. Attribute McuFTM3Ch7ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3_CH7 modulation with FTM2_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.29 McuFTM0Ch0ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM0_OUTSEL] - FTM0 channel modulation select with FTM1_CH1

0 - No modulation with FTM1_CH1

1 - Modulation with FTM1_CH1

Note: Implementation Specific Parameter.

Table 4-211. Attribute McuFTM0Ch0ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0_CH0 modulation with FTM1_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.30 McuFTM0Ch1ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM0_OUTSEL] - FTM0 channel modulation select with FTM1_CH1

0 - No modulation with FTM1_CH1

1 - Modulation with FTM1_CH1

Note: Implementation Specific Parameter.

Table 4-212. Attribute McuFTM0Ch1ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0_CH1 modulation with FTM1_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.31 McuFTM0Ch2ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM0_OUTSEL] - FTM0 channel modulation select with FTM1_CH1

0 - No modulation with FTM1_CH1

1 - Modulation with FTM1_CH1

Note: Implementation Specific Parameter.

Table 4-213. Attribute McuFTM0Ch2ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0_CH2 modulation with FTM1_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.32 McuFTM0Ch3ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM0_OUTSEL] - FTM0 channel modulation select with FTM1_CH1

0 - No modulation with FTM1_CH1

1 - Modulation with FTM1_CH1

Note: Implementation Specific Parameter.

Table 4-214. Attribute McuFTM0Ch3ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0_CH3 modulation with FTM1_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.33 McuFTM0Ch4ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM0_OUTSEL] - FTM0 channel modulation select with FTM1_CH1

0 - No modulation with FTM1_CH1

1 - Modulation with FTM1_CH1

Note: Implementation Specific Parameter.

Table 4-215. Attribute McuFTM0Ch4ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0_CH4 modulation with FTM1_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.34 McuFTM0Ch5ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM0_OUTSEL] - FTM0 channel modulation select with FTM1_CH1

0 - No modulation with FTM1_CH1

1 - Modulation with FTM1_CH1

Note: Implementation Specific Parameter.

Table 4-216. Attribute McuFTM0Ch5ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0_CH5 modulation with FTM1_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.35 McuFTM0Ch6ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM0_OUTSEL] - FTM0 channel modulation select with FTM1_CH1

0 - No modulation with FTM1_CH1

1 - Modulation with FTM1_CH1

Note: Implementation Specific Parameter.

Table 4-217. Attribute McuFTM0Ch6ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0_CH6 modulation with FTM1_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.36 McuFTM0Ch7ModulationSelect (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM0_OUTSEL] - FTM0 channel modulation select with FTM1_CH1

0 - No modulation with FTM1_CH1

1 - Modulation with FTM1_CH1

Note: Implementation Specific Parameter.

Table 4-218. Attribute McuFTM0Ch7ModulationSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0_CH7 modulation with FTM1_CH1
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.37 McuFTM2Ch1InputSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM2CH1SEL] - FTM2 CH1 Select

Selects FTM2 CH1 input

0 - FTM2_CH1 input

1 - exclusive OR of FTM2_CH0,FTM2_CH1, and FTM1_CH1

Note: Implementation Specific Parameter.

Table 4-219. Attribute McuFTM2Ch1InputSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0FLT0 pin or TRGMUX_FTM0 out
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM2_CHI1_INPUT
Range	FTM2_CHI1_INPUT FTM2_CH0_XOR_FTM2_CH1_XOR_FTM1_CH1

4.6.15.2.38 McuFTM2Ch0InputSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM2CH0SEL] - FTM2 CH0 Select

Selects FTM2 CH0 input

0 - FTM2_CH0 input

1 - CMP0 output

2 - Reserved

3 - Reserved

Note: Implementation Specific Parameter.

Table 4-220. Attribute McuFTM2Ch0InputSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM2_CH0 input select
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM2_CH0_INPUT
Range	FTM2_CH0_INPUT CMP0_OUTPUT

4.6.15.2.39 McuFTM1Ch0InputSelect (McuFlexTimerConfiguration)

SIM_FTMOPT0[FTM1CH0SEL] - FTM1 CH0 Select

Selects FTM1 CH0 input

0 - FTM1_CH0 input

1 - CMP0 output

2 - Reserved

3 - Reserved

Note: Implementation Specific Parameter.

Table 4-221. Attribute McuFTM1Ch0InputSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM1_CH0 input select

Table continues on the next page...

Table 4-221. Attribute McuFTM1Ch0InputSelect (McuFlexTimerConfiguration) detailed description (continued)

Property	Value
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	FTM1_CH0_INPUT
Range	FTM1_CH0_INPUT CMP0_OUTPUT

4.6.15.2.40 McuFTMGlobalLoadEnable (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTMGLDOK] - FTM global load enable

unchecked - FTM Global load mechanism disabled.

checked - FTM Global load mechanism enabled

Note: Implementation Specific Parameter.

Table 4-222. Attribute McuFTMGlobalLoadEnable (McuFlexTimerConfiguration) detailed description

Property	Value
Label	Enable FTM global load enable
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.41 McuFTM7SyncBit (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM7SYNCBIT] - FTM7 Sync Bit

Note: Implementation Specific Parameter.

Table 4-223. Attribute McuFTM7SyncBit (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM7 Sync Bit

Table continues on the next page...

Table 4-223. Attribute McuFTM7SyncBit (McuFlexTimerConfiguration) detailed description (continued)

Property	Value
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.42 McuFTM6SyncBit (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM6SYNCBIT] - FTM6 Sync Bit

Note: Implementation Specific Parameter.

Table 4-224. Attribute McuFTM6SyncBit (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM6 Sync Bit
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.43 McuFTM5SyncBit (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM5SYNCBIT] - FTM5 Sync Bit

Note: Implementation Specific Parameter.

Table 4-225. Attribute McuFTM5SyncBit (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM5 Sync Bit
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.44 McuFTM4SyncBit (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM4SYNCBIT] - FTM4 Sync Bit

Note: Implementation Specific Parameter.

Table 4-226. Attribute McuFTM4SyncBit (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM4 Sync Bit
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.45 McuFTM3SyncBit (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM3SYNCBIT] - FTM3 Sync Bit

Note: Implementation Specific Parameter.

Table 4-227. Attribute McuFTM3SyncBit (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3 Sync Bit
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.46 McuFTM2SyncBit (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM2SYNCBIT] - FTM2 Sync Bit

Note: Implementation Specific Parameter.

Table 4-228. Attribute McuFTM2SyncBit (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM2 Sync Bit
Type	BOOLEAN

Table continues on the next page...

Table 4-228. Attribute McuFTM2SyncBit (McuFlexTimerConfiguration) detailed description (continued)

Property	Value
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.47 McuFTM1SyncBit (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM1SYNCBIT] - FTM1 Sync Bit

Note: Implementation Specific Parameter.

Table 4-229. Attribute McuFTM1SyncBit (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM1 Sync Bit
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.48 McuFTM0SyncBit (McuFlexTimerConfiguration)

SIM_FTMOPT1[FTM0SYNCBIT] - FTM0 Sync Bit

Note: Implementation Specific Parameter.

Table 4-230. Attribute McuFTM0SyncBit (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0 Sync Bit
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.49 McuQspiClkSelect (McuFlexTimerConfiguration)

SIM_MISCTRL0[QSPI_CLK_SEL] - QSPI asynchronous clock gating enable.

unchecked - QuadSPI internal reference clock is gated.

checked - QuadSPI internal reference clock is enabled.

Note: Implementation Specific Parameter.

Table 4-231. Attribute McuQspiClkSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	QSPI_CLK Select bit
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.50 McuRMII_ClkSelect (McuFlexTimerConfiguration)

SIM_MISCTRL0[RMII_CLK_SEL] - Set this bit to enable SOSCDIV1_CLK as ENET RMII clock in Internal loopback mode.

0 - Disable SOSCDIV1_CLK as ENET RMII clock in Internal loopback mode.

1 - Enable SOSCDIV1_CLK as ENET RMII clock in Internal loopback mode.

Note: Implementation Specific Parameter.

Table 4-232. Attribute McuRMII_ClkSelect (McuFlexTimerConfiguration) detailed description

Property	Value
Label	RMII_CLK Select bit
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.51 McuRMII_Clk_OBE (McuFlexTimerConfiguration)

SIM_MISCTRL0[RMII_CLK_OBE] - Output Buffer Enable for ENET RMII clock in internal loopback mode.

0 - Disable Output Buffer

1 - Enable Output Buffer

Note: Implementation Specific Parameter.

Table 4-233. Attribute McuRMII_Clk_OBE (McuFlexTimerConfiguration) detailed description

Property	Value
Label	RMII_CLK OBE bit
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.52 McuFTM7OBEControl (McuFlexTimerConfiguration)

SIM_MISCTRL0[FTM7_OBE_CTRL] - FTM7 OBE Control bit

unchecked - The FTM channel output is put to safe state when the FTM counter is enabled and the FTM channel output is enabled by Fault Control (FTM_MODE[FAULTM] != 0 and FTM_FLTCTRL[FSTATE] = 0) and PWM is enabled (FTM_SC[PWMENn] = 1). Otherwise the channel output is tristated. checked - The FTM channel output state is retained when the channel is in output mode. The output is tristated when the channel is in input capture [DECAPEN = 0, COMBINE = 0, MSnB:MSnA = 0] or dual edge capture mode [DECAPEN = 1] Note: Implementation Specific Parameter.

Table 4-234. Attribute McuFTM7OBEControl (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM7 OBE Control
Type	BOOLEAN
Origin	Custom

Table continues on the next page...

Table 4-234. Attribute McuFTM7OBEControl (McuFlexTimerConfiguration) detailed description (continued)

Property	Value
Symbolic Name	false
Default	false

4.6.15.2.53 McuFTM6OBEControl (McuFlexTimerConfiguration)

SIM_MISCTRL0[FTM6_OBE_CTRL] - FTM6 OBE Control bit

unchecked - The FTM channel output is put to safe state when the FTM counter is enabled and the FTM channel output is enabled by Fault Control (FTM_MODE[FAULTM] != 0 and FTM_FLTCTRL[FSTATE] = 0) and PWM is enabled (FTM_SC[PWMENn] = 1). Otherwise the channel output is tristated. checked - The FTM channel output state is retained when the channel is in output mode. The output is tristated when the channel is in input capture [DECAPEN = 0, COMBINE = 0, MSnB:MSnA = 0] or dual edge capture mode [DECAPEN = 1] Note: Implementation Specific Parameter.

Table 4-235. Attribute McuFTM6OBEControl (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM6 OBE Control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.54 McuFTM5OBEControl (McuFlexTimerConfiguration)

SIM_MISCTRL0[FTM5_OBE_CTRL] - FTM5 OBE Control bit

unchecked - The FTM channel output is put to safe state when the FTM counter is enabled and the FTM channel output is enabled by Fault Control (FTM_MODE[FAULTM] != 0 and FTM_FLTCTRL[FSTATE] = 0) and PWM is enabled (FTM_SC[PWMENn] = 1). Otherwise the channel output is tristated. checked - The FTM channel output state is retained when the channel is in output mode. The output

is tristated when the channel is in input capture [DECAPEN = 0, COMBINE = 0, MSnB:MSnA = 0] or dual edge capture mode [DECAPEN = 1] Note: Implementation Specific Parameter.

Table 4-236. Attribute McuFTM5OBEControl (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM5 OBE Control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.55 McuFTM4OBEControl (McuFlexTimerConfiguration)

SIM_MISCTRL0[FTM4_OBE_CTRL] - FTM4 OBE Control bit

unchecked - The FTM channel output is put to safe state when the FTM counter is enabled and the FTM channel output is enabled by Fault Control (FTM_MODE[FAULTM] != 0 and FTM_FLTCTRL[FSTATE] = 0) and PWM is enabled (FTM_SC[PWMENn] = 1). Otherwise the channel output is tristated. checked - The FTM channel output state is retained when the channel is in output mode. The output is tristated when the channel is in input capture [DECAPEN = 0, COMBINE = 0, MSnB:MSnA = 0] or dual edge capture mode [DECAPEN = 1] Note: Implementation Specific Parameter.

Table 4-237. Attribute McuFTM4OBEControl (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM4 OBE Control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.56 McuFTM3OBEControl (McuFlexTimerConfiguration)

SIM_MISCTRL0[FTM3_OBE_CTRL] - FTM3 OBE Control bit

unchecked - The FTM channel output is put to safe state when the FTM counter is enabled and the FTM channel output is enabled by Fault Control (FTM_MODE[FAULTM] != 0 and FTM_FLTCTRL[FSTATE] = 0) and PWM is enabled (FTM_SC[PWMENn] = 1). Otherwise the channel output is tristated. checked - The FTM channel output state is retained when the channel is in output mode. The output is tristated when the channel is in input capture [DECAPEN = 0, COMBINE = 0, MSnB:MSnA = 0] or dual edge capture mode [DECAPEN = 1] Note: Implementation Specific Parameter.

Table 4-238. Attribute McuFTM3OBEControl (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM3 OBE Control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.57 McuFTM2OBEControl (McuFlexTimerConfiguration)

SIM_MISCTRL0[FTM2_OBE_CTRL] - FTM2 OBE Control bit

unchecked - The FTM channel output is put to safe state when the FTM counter is enabled and the FTM channel output is enabled by Fault Control (FTM_MODE[FAULTM] != 0 and FTM_FLTCTRL[FSTATE] = 0) and PWM is enabled (FTM_SC[PWMENn] = 1). Otherwise the channel output is tristated. checked - The FTM channel output state is retained when the channel is in output mode. The output is tristated when the channel is in input capture [DECAPEN = 0, COMBINE = 0, MSnB:MSnA = 0] or dual edge capture mode [DECAPEN = 1] Note: Implementation Specific Parameter.

Table 4-239. Attribute McuFTM2OBEControl (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM2 OBE Control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.58 McuFTM1OBEControl (McuFlexTimerConfiguration)

SIM_MISCTRL0[FTM1_OBE_CTRL] - FTM1 OBE Control bit

unchecked - The FTM channel output is put to safe state when the FTM counter is enabled and the FTM channel output is enabled by Fault Control (FTM_MODE[FAULTM] != 0 and FTM_FLTCTRL[FSTATE] = 0) and PWM is enabled (FTM_SC[PWMENn] = 1). Otherwise the channel output is tristated. checked - The FTM channel output state is retained when the channel is in output mode. The output is tristated when the channel is in input capture [DECAPEN = 0, COMBINE = 0, MSnB:MSnA = 0] or dual edge capture mode [DECAPEN = 1] Note: Implementation Specific Parameter.

Table 4-240. Attribute McuFTM1OBEControl (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM1 OBE Control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.59 McuFTM0OBEControl (McuFlexTimerConfiguration)

SIM_MISCTRL0[FTM0_OBE_CTRL] - FTM0 OBE Control bit

unchecked - The FTM channel output is put to safe state when the FTM counter is enabled and the FTM channel output is enabled by Fault Control (FTM_MODE[FAULTM] != 0 and FTM_FLTCTRL[FSTATE] = 0) and PWM is enabled (FTM_SC[PWMENn] = 1). Otherwise the channel output is tristated. checked - The FTM channel output state is retained when the channel is in output mode. The output is tristated when the channel is in input capture [DECAPEN = 0, COMBINE = 0, MSnB:MSnA = 0] or dual edge capture mode [DECAPEN = 1] Note: Implementation Specific Parameter.

Table 4-241. Attribute McuFTM0OBEControl (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM0 OBE Control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.2.60 McuFTM_GTBControl (McuFlexTimerConfiguration)

SIM_MISCTRL0[FTM_GTB_SPLIT_EN] - FTM GTB split enable/disable bit

unchecked - All the FTMs have a single global time-base.

checked - FTM0-3 have a common time-base and others have a different common time-base. Please refer 'FTM global time base' in FTM chapter in RM for implementation details.

Note: Implementation Specific Parameter.

Table 4-242. Attribute McuFTM_GTBControl (McuFlexTimerConfiguration) detailed description

Property	Value
Label	FTM_GTB bit Control
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.6.15.3 Form McuAdcOptionsConfiguration

This container contains the configuration for the SIM_ADCOPT and SIM_MISCTRL registers.

Is included by form : Form McuSIMConfig

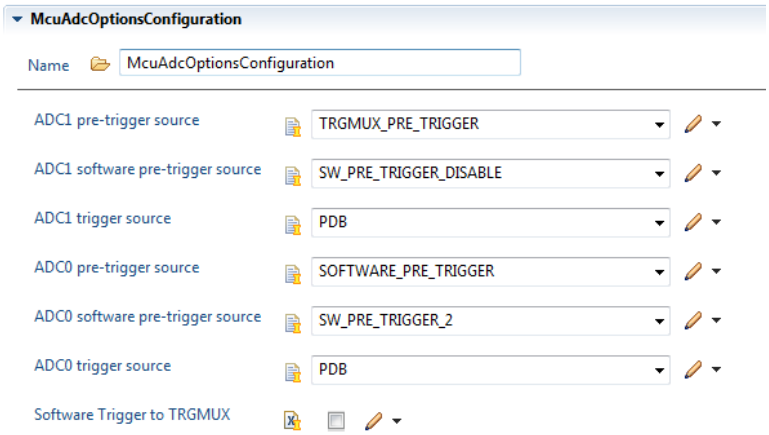


Figure 4-31. Tresos Plugin snapshot for McuAdcOptionsConfiguration form.

4.6.15.3.1 McuADC1PreTrigeerSourceSelect (McuAdcOptionsConfiguration)

SIM_ADCOPT[ADC1PRETRGSE] - ADC1 pre-trigger source select

Selects pre-trigger source for ADC1.

0 - PDB pre-trigger (default)

1 - TRGMUX pre-trigger

2 - Software pre-trigger

Note: Implementation Specific Parameter.

Table 4-243. Attribute McuADC1PreTrigeerSourceSelect (McuAdcOptionsConfiguration) detailed description

Property	Value
Label	ADC1 pre-trigger source
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	PDB_PRE_TRIGGER
Range	PDB_PRE_TRIGGER TRGMUX_PRE_TRIGGER SOFTWARE_PRE_TRIGGER

4.6.15.3.2 McuADC1SoftwarePreTrigeerSourceSelect (McuAdcOptionsConfiguration)

SIM_ADCOPT[ADC1SWPRETRG] - ADC1 software pre-trigger sources

0 - software pre-trigger disabled

4 - software pre-trigger 0

5 - software pre-trigger 1

6 - software pre-trigger 2

7 - software pre-trigger 3

Note: Implementation Specific Parameter.

Table 4-244. Attribute McuADC1SoftwarePreTrigeerSourceSelect (McuAdcOptionsConfiguration) detailed description

Property	Value
Label	ADC1 software pre-trigger source
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	SW_PRE_TRIGGER_DISABLE
Range	SW_PRE_TRIGGER_DISABLE SW_PRE_TRIGGER_0 SW_PRE_TRIGGER_1 SW_PRE_TRIGGER_2 SW_PRE_TRIGGER_3

4.6.15.3.3 McuADC1TrigeerSourceSelect (McuAdcOptionsConfiguration)

SIM_ADCOPT[ADC1TRGSEL] - ADC1 trigger source select

Selects trigger source for ADC1.

0 - PDB output

1 - TRGMUX output

Note: Implementation Specific Parameter.

Table 4-245. Attribute McuADC1TrigeerSourceSelect (McuAdcOptionsConfiguration) detailed description

Property	Value
Label	ADC1 trigger source
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	PDB
Range	PDB TRGMUX

4.6.15.3.4 McuADC0PreTrigeerSourceSelect (McuAdcOptionsConfiguration)

SIM_ADCOPT[ADC0PRETRGSE] - ADC0 pre-trigger source select

Selects pre-trigger source for ADC0.

0 - PDB pre-trigger (default)

1 - TRGMUX pre-trigger

2 - Software pre-trigger

Note: Implementation Specific Parameter.

Table 4-246. Attribute McuADC0PreTrigeerSourceSelect (McuAdcOptionsConfiguration) detailed description

Property	Value
Label	ADC0 pre-trigger source
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	PDB_PRE_TRIGGER
Range	PDB_PRE_TRIGGER TRGMUX_PRE_TRIGGER SOFTWARE_PRE_TRIGGER

4.6.15.3.5 McuADC0SoftwarePreTrigeerSourceSelect (McuAdcOptionsConfiguration)

SIM_ADCOPT[ADC0SWPRETRG] - ADC0 software pre-trigger sources

0 - software pre-trigger disabled

4 - software pre-trigger 0

5 - software pre-trigger 1

6 - software pre-trigger 2

7 - software pre-trigger 3

Note: Implementation Specific Parameter.

Table 4-247. Attribute McuADC0SoftwarePreTrigeerSourceSelect (McuAdcOptionsConfiguration) detailed description

Property	Value
Label	ADC0 software pre-trigger source
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	SW_PRE_TRIGGER_DISABLE
Range	SW_PRE_TRIGGER_DISABLE SW_PRE_TRIGGER_0 SW_PRE_TRIGGER_1 SW_PRE_TRIGGER_2 SW_PRE_TRIGGER_3

4.6.15.3.6 McuADC0TrigeerSourceSelect (McuAdcOptionsConfiguration)

SIM_ADCOPT[ADC2TRGSEL] - ADC0 trigger source select

Selects trigger source for ADC0.

0 - PDB output

1 - TRGMUX output

Note: Implementation Specific Parameter.

Table 4-248. Attribute McuADC0TrigeerSourceSelect (McuAdcOptionsConfiguration) detailed description

Property	Value
Label	ADC0 trigger source
Type	ENUMERATION
Origin	Custom
Symbolic Name	false
Default	PDB
Range	PDB TRGMUX

4.6.15.3.7 McuSoftwareTriggerToTRGMUX (McuAdcOptionsConfiguration)

SIM_MISCTRL[SW_TRG] - Software Trigger bit to TRGMUX

Note: Implementation Specific Parameter.

Table 4-249. Attribute McuSoftwareTriggerToTRGMUX (McuAdcOptionsConfiguration) detailed description

Property	Value
Label	Software Trigger to TRGMUX
Type	BOOLEAN
Origin	Custom
Symbolic Name	false
Default	false

4.7 Form CommonPublishedInformation

Common container, aggregated by all modules. It contains published information about vendor and versions.

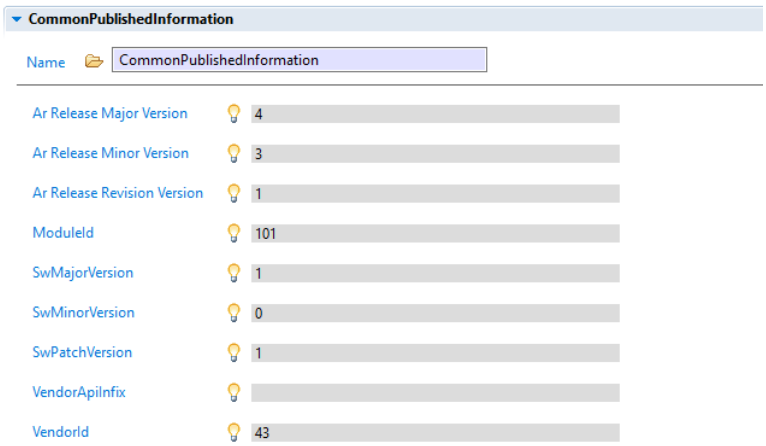


Figure 4-32. Tresos Plugin snapshot for CommonPublishedInformation form.

4.7.1 ArReleaseMajorVersion (CommonPublishedInformation)

Major version number of AUTOSAR specification on which the appropriate implementation is based on.

Table 4-250. Attribute ArReleaseMajorVersion (CommonPublishedInformation) detailed description

Property	Value
Label	AUTOSAR Major Version
Type	INTEGER_LABEL
Origin	Custom
Symbolic Name	false
Default	4
Invalid	Range >=4 <=4

4.7.2 ArReleaseMinorVersion (CommonPublishedInformation)

Minor version number of AUTOSAR specification on which the appropriate implementation is based on.

Table 4-251. Attribute ArReleaseMinorVersion (CommonPublishedInformation) detailed description

Property	Value
Label	AUTOSAR Minor Version
Type	INTEGER_LABEL
Origin	Custom
Symbolic Name	false
Default	3
Invalid	Range >=3 <=3

4.7.3 ArReleaseRevisionVersion (CommonPublishedInformation)

Revision version number of AUTOSAR specification on which the appropriate implementation is based on.

Table 4-252. Attribute ArReleaseRevisionVersion (CommonPublishedInformation) detailed description

Property	Value
Label	AUTOSAR Release Revision Version
Type	INTEGER_LABEL
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range >=1 <=1

4.7.4 ModuleId (CommonPublishedInformation)

Module ID of this module from Module List.

Table 4-253. Attribute ModuleId (CommonPublishedInformation) detailed description

Property	Value
Label	Module Id
Type	INTEGER_LABEL
Origin	Custom
Symbolic Name	false

Table continues on the next page...

Table 4-253. Attribute ModuleId (CommonPublishedInformation) detailed description (continued)

Property	Value
Default	101
Invalid	Range >=101 <=101

4.7.5 SwMajorVersion (CommonPublishedInformation)

Major version number of the vendor specific implementation of the module. The numbering is vendor specific.

Table 4-254. Attribute SwMajorVersion (CommonPublishedInformation) detailed description

Property	Value
Label	Software Major Version
Type	INTEGER_LABEL
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range >=1 <=1

4.7.6 SwMinorVersion (CommonPublishedInformation)

Minor version number of the vendor specific implementation of the module. The numbering is vendor specific.

Table 4-255. Attribute SwMinorVersion (CommonPublishedInformation) detailed description

Property	Value
Label	Software Minor Version
Type	INTEGER_LABEL
Origin	Custom
Symbolic Name	false
Default	0
Invalid	Range

Table 4-255. Attribute SwMinorVersion (CommonPublishedInformation) detailed description

Property	Value
	>=0 <=0

4.7.7 SwPatchVersion (CommonPublishedInformation)

Patch level version number of the vendor specific implementation of the module. The numbering is vendor specific.

Table 4-256. Attribute SwPatchVersion (CommonPublishedInformation) detailed description

Property	Value
Label	Software Patch Version
Type	INTEGER_LABEL
Origin	Custom
Symbolic Name	false
Default	1
Invalid	Range >=1 <=1

4.7.8 VendorApiInfix (CommonPublishedInformation)

In driver modules which can be instantiated several times on a single ECU, BSW00347 requires that the name of APIs is extended by the VendorId and a vendor specific name. This parameter is used to specify the vendor specific name. In total, the implementation specific name is generated as follows: <ModuleName>_<VendorId>_<VendorApiInfix><Api name from SWS>. E.g. assuming that the VendorId of the implementor is 123 and the implementer chose a VendorApiInfix of "v11r456" a api name Can_Write defined in the SWS will translate to Can_123_v11r456Write. This parameter is mandatory for all modules with upper multiplicity > 1. It shall not be used for modules with upper multiplicity =1.

Table 4-257. Attribute VendorApiInfix (CommonPublishedInformation) detailed description

Property	Value
Label	Vendor Api Infix

Table continues on the next page...

Table 4-257. Attribute VendorApiInfix (CommonPublishedInformation) detailed description (continued)

Property	Value
Type	STRING_LABEL
Origin	Custom
Symbolic Name	false
Default	
Enable	false

4.7.9 VendorId (CommonPublishedInformation)

Vendor ID of the dedicated implementation of this module according to the AUTOSAR vendor list.

Table 4-258. Attribute VendorId (CommonPublishedInformation) detailed description

Property	Value
Label	Vendor Id
Type	INTEGER_LABEL
Origin	Custom
Symbolic Name	false
Default	43
Invalid	Range >=43 <=43

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer’s technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer's applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

Document Number UM2MCUASR4.3 Rev0001R1.0.1
Revision 1.0