SCOPE OF APPLICATION **All Project/Engineering**	HYUNDAI AutoEver	**SHT/SHTS** **1 / 26**
Responsibility: Classic AUTOSAR Team	AUTOSAR Tm User Manual	**DOC. NO**
AUTOSAR Tm User Manual		

Document Change Histroy				
Date (YYYY-MM-DD)	Ver.	Editor	Chap	Content
2020-06-19	1.0.0.0	VuCH1	All	Initial Version
2022-06-30	1.0.1.0	PhatCT1	4.2 4.3	Scope of the Release Change Log
2022-09-29	1.0.1.1	KiYoung Yun	4.3	Change Log
2023-03-17	1.0.2.0	NamNT1	4.2 4.3	Scope of the Release Change Log
2023-07-24	1.0.3.0	NamNT1	4.2 4.3 All	Scope of the Release Change Log Apply new UM template
2023-12-29	1.0.4.0	NamNT1	4.2 4.3	Scope of the Release Change Log

Edition Date: 2023/12/29	File Name Tm_UM.pdf	Creation NamNT1	Check JH Cho 2023/12/29	Approval JH Jeong 2023/12/29
Document Management System				

Table of Contents

1 Overview

It is written based on AUTOSAR standard SRS / SWS. If more detailed functional explanation is needed when using the module, see the Reference Manual. The interpretation of setting related category is as follows:

- Changeable (C): Items that can be set by the user
- Fixed (F): Items that cannot be changed by the user.
- Not Supported (N): Deprecated item

2 Reference

Sl. No.	Title	Version
1	AUTOSAR_SWS_TimeService.pdf	4.4.0

2.1 Acronyms, abbreviations and term

Abbreviation:	***Description:***
API	Application Programming Interface
AUTOSAR	AUTomotive Open System ARchitecture
BSW	Basic SoftWare
ID/Id	Identifier
RAM	Random Access Memory
SRS	Software Requirements Specification
STS	System Test Specification
SWS	SoftWare Specification
SID	Service Id
SW-C	SoftWare Component
API	Application Programming Interface
AUTOSAR	AUTomotive Open System ARchitecture
BSW	Basic SoftWare
Nop	No Operation

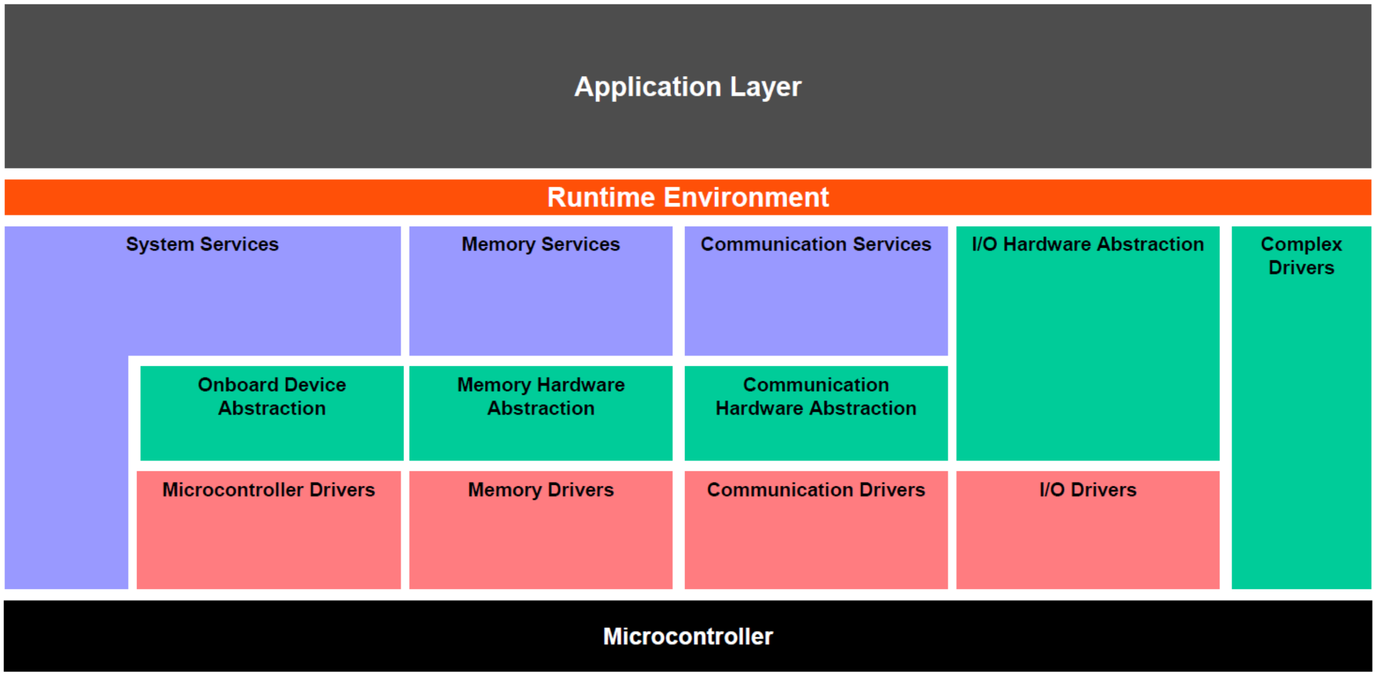
Term:	***Description:***
GPT Predef Timer	A GPT Predef Timer is a free running up counter provided by the GPT driver. Which GPT Predef Timer(s) are available depends on hardware (clock, hardware timers, prescaler, width of timer register, …) and configuration. A GPT Predef Timer has predefined physical time unit and range
Time Service Predef Timer	A Time Service Predef Timer is a free running up counter with predefined physical time unit and range. The hardware timer functionality is based on the corresponding GPT Predef Timer. For each Predef Timer a set of API services is provided by the Time Service module. The user can instantiate any timers (only limited by available memory) and can use the instances completely independently of each other.
Timer instance	A timer instance is a data object of an API data type `Tm_PredefTimer…bitType`, this means it is an instantiation of a Time Service Predef Timer on user software level. The user can instantiate any timers (only limited by available memory). The timer instances

	can be used completely independently of each other by methodes provided as API services.
Reference time	The reference time is a time value stored for each timer instance. It’s an implementation specific element of the API data types `Tm_PredefTimer…bitType`.

3 AUTOSAR System

3.1 Overview of Software Layers

The Layered Architecture of the AUTOSAR platform is as follows. The AUTOSAR platform can be divided into Service Layer, ECU Abstraction Layer, Complex Device Drivers, and Microcontroller Abstraction Layer.



3.2 AUTOSAR Time Service Module

The Time Service module is part of the Services Layer. The module provides services for time based functionality. Use cases are:

- Time measurement.
- Time based state machine.
- Timeout supervision.
- Busy waiting.

4 Product Release Notes

4.1 Overview

This chapter aims to provide release-related content for the HYUNDAI AUTOEVER Tm Module, and describes restrictions and specifics for the Tm Module Software product release version

4.2 Scope of the Release

All contents of this document are limited to the following HYUNDAI AUTOEVER Tm modules.

Module Name	AUTOSAR Version	Module Version
Tm	4.4.0	1.0.4

Module version means Sw version of each module's BswModule Description (Bswmd) file.

4.3 Change Log

4.3.1 Version 1.0.4.0

- Improvement
  - Checking Gentool limitation version and change it

Cause	Gentool framework lower limit version is not 1.0.0.0
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.2 Version 1.0.3.0

- Improvement
  - Tm_Version.c should support GPT AR 4.3

Cause	GPT of ORIN FSI MCAL is AR 4.3
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

- Improvement
  - GPT of ORIN FSI MCAL does not support GPT_PREDEF_TIMER_100US_32BIT. So, In case of Orin FSI Mcal TmEnablePredefTimer100us32bit shall be set to false.

Cause	Configuration Guide for Orin_FSI Mcal
Operation Impact	N/A
Configuration Impact	N/A
Required measure	N/A

of ASW	

4.3.3 Version 1.0.2.0

- Improvement
  - Tm_Version.c shall support 4.4.0 and 4.2.2 mcal

Cause	Tm_Version.h and .c files only support AR 4.2.2 version of Infinion 2G mcal. So the compile errror occurs because of the version miss match of MCal GPT and Tm.
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.4 Version 1.0.1.1

- Task
  - Attachment of UNECE Cyber Security report

Cause	Attachment of UNECE Cyber Security report
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.5 Version 1.0.1.0

- Defect
  - Traceability Coverage is not 100%

Cause	Traceability Coverage is not 100%
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

- Development
  - Editorial Changes of Work Products

Cause	Clarification of the scope of code usage
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

- Development
  - Naming convention and template change

Cause	Naming convention and template change
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

- Defect
  - Fix Compiler warning

Cause	Fix compiler warning
Operation Impact	N/A
Configuration Impact	N/A
Required measure of ASW	N/A

4.3.6 Version 1.0.0.0

- Development
  - Initial Version

Cause	Initial Version
Operation Impact	N/A
Configuration Impact	N/A

Required measure of ASW	N/A

4.4 Limitations

4.4.1 Limitations

1) Only Pre-Compile is supported

2) GPT of ORIN FSI MCAL does not support GPT_PREDEF_TIMER_100US_32BIT. So, In case of Orin FSI Mcal Tm/TmGeneral/TmEnablePredefTimer100us32bit shall be set to false.

4.5 Deviations

None

5 Configuration Guide

The Tm setting of the AUTOSAR platform distributed by HYUNDAI AUTOEVER is a setting reflecting HYUNDAI AUTOEVER Policy's policy. Therefore, you should consult with HYUNDAI AUTOEVER.

5.1 TmGeneral

Parameter Name	Value	Category
TmDevErrorDetect	FALSE	Changeable
TmEnablePredefTimer100us32bit	FALSE	Changeable
TmEnablePredefTimer1us32bit	FALSE	Changeable
TmEnablePredefTimer1us24bit	FALSE	Changeable
TmEnablePredefTimer1us16bit	FALSE	Changeable
TmVersionInfoApi	FALSE	Changeable
Wait Loop (Optional)	FALSE	Changeable
Wait Loop Number (Optional)	--	Changeable

6 Application Programming Interface (API)

6.1 Type Definitions

6.1.1 Tm_PredefTimer1us16bitType

Data type of Time Service Predef Timer 1us16bit, the structure contains the reference time.

6.1.2 Tm_PredefTimer1us24bitType

Data type of Time Service Predef Timer 1us24bit, the structure contains the reference time.

6.1.3 Tm_PredefTimer1us32bitType

Data type of Time Service Predef Timer 1us32bit, the structure contains the reference time.

6.1.4 Tm_PredefTimer100us32bitType

Data type of Time Service Predef Timer 100us32bit, the structure contains the reference time.

6.2 Macro Constants

None

6.3 Functions

6.3.1 GetVesionInfo

Function Name	Tm_GetVersionInfo	
Syntax	void Tm_GetVersionInfo(Std_VersionInfoType* VersionInfoPtr)	
Service ID [Hex]	0x1	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	None	
Parameters (Inout)	None	
Parameters (Out)	VersionInfoPtr	Pointer to where to store the version information of this module.
Return Value	None	
Description	Returns the version information of this module.	
Available via	Tm.h	

6.3.2 ResetTimer

Function Name	Tm_ResetTimer1us16bit
Syntax	Std_ReturnType Tm_ResetTimer1us16bit(Tm_PredefTimer1us16bitType* TimerPtr)
Service ID [Hex]	0x2

Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same timer instance	
Parameters (In)	None	
Reentrancy	None	
Parameters (In)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Inout)	Std_ReturnType	E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected
Parameters (Out)	Resets a timer instance (user point of view).	
Return Value	Tm.h	

Function Name	Tm_ResetTimer1us24bit	
Syntax	Std_ReturnType Tm_ResetTimer1us24bit(Tm_PredefTimer1us24bitType* TimerPtr)	
Service ID [Hex]	0x7	
Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same timer instance	
Parameters (In)	None	
Reentrancy	None	
Parameters (In)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Inout)	Std_ReturnType	E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected
Parameters (Out)	Resets a timer instance (user point of view).	
Return Value	Tm.h	

Function Name	Tm_ResetTimer1us32bit

Syntax	Std_ReturnType Tm_ResetTimer1us32bit(Tm_PredefTimer1us32bitType* TimerPtr)	
Service ID [Hex]	0xC	
Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same timer instance	
Parameters (In)	None	
Reentrancy	None	
Parameters (In)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Inout)	Std_ReturnType	E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected
Parameters (Out)	Resets a timer instance (user point of view).	
Return Value	Tm.h	

Function Name	Tm_ResetTimer100us32bit	
Syntax	Std_ReturnType Tm_ResetTimer100us32bit(Tm_PredefTimer100us32bitType* TimerPtr)	
Service ID [Hex]	0x11	
Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same timer instance	
Parameters (In)	None	
Reentrancy	None	
Parameters (In)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Inout)	Std_ReturnType	E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected
Parameters (Out)	Resets a timer instance (user point of view).	

Return Value	Tm.h

6.3.3 GetTimeSpan

Function Name	Tm_GetTimeSpan1us16bit	
Syntax	Std_ReturnType Tm_GetTimeSpan1us16bit(const Tm_PredefTimer1us16bitType* TimerPtr, uint16* TimeSpanPtr)	
Service ID [Hex]	0x3	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Inout)	None	
Parameters (Out)	TimeSpanPtr	Pointer to time span destination data in RAM
Return Value		E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected
Description	Delivers the time difference (current time - reference time).	
Available via	Tm.h	

Function Name	Tm_GetTimeSpan1us24bit	
Syntax	Std_ReturnType Tm_GetTimeSpan1us24bit(const Tm_PredefTimer1us24bitType* TimerPtr, uint32* TimeSpanPtr)	
Service ID [Hex]	0x8	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Inout)	None	

Parameters (Out)	TimeSpanPtr	Pointer to time span destination data in RAM
Return Value		E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected
Description	Delivers the time difference (current time - reference time).	
Available via	Tm.h	

Function Name	Tm_GetTimeSpan1us32bit	
Syntax	Std_ReturnType Tm_GetTimeSpan1us32bit(const Tm_PredefTimer1us32bitType* TimerPtr, uint32* TimeSpanPtr)	
Service ID [Hex]	0xd	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Inout)	None	
Parameters (Out)	TimeSpanPtr	Pointer to time span destination data in RAM
Return Value		E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected
Description	Delivers the time difference (current time - reference time).	
Available via	Tm.h	

Function Name	Tm_GetTimeSpan100us32bit
Syntax	Std_ReturnType Tm_GetTimeSpan100us32bit(const Tm_PredefTimer100us32bitType* TimerPtr, uint32* TimeSpanPtr)
Service ID [Hex]	0x12
Sync/Async	Synchronous
Reentrancy	Reentrant

Parameters (In)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Inout)	None	
Parameters (Out)	TimeSpanPtr	Pointer to time span destination data in RAM
Return Value		E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected
Description	Delivers the time difference (current time - reference time).	
Available via	Tm.h	

6.3.4 ShiftTimer

Function Name	Tm_ShiftTimer1us16bit	
Syntax	void Tm_ShiftTimer1us16bit(Tm_PredefTimer1us16bitType* TimerPtr, uint16 TimeValue)	
Service ID [Hex]	0x4	
Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same timer instance	
Parameters (In)	TimeValue	Time value in µs, the reference time has to be shifted.
Parameters (Inout)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Out)	None	
Return Value	None	
Description	Shifts the reference time of the timer instance.	
Available via	Tm.h	

Function Name	Tm_ShiftTimer1us24bit
Syntax	void Tm_ShiftTimer1us24bit(Tm_PredefTimer1us24bitType* TimerPtr, uint32 TimeValue)
Service ID [Hex]	0x9

Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same timer instance	
Parameters (In)	TimeValue	Time value in μs, the reference time has to be shifted. Range: 0-0xFFFFFF
Parameters (Inout)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Out)	None	
Return Value	None	
Description	Shifts the reference time of the timer instance.	
Available via	Tm.h	

Function Name	Tm_ShiftTimer1us32bit	
Syntax	void Tm_ShiftTimer1us32bit(Tm_PredefTimer1us32bitType* TimerPtr, uint32 TimeValue)	
Service ID [Hex]	0xe	
Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same timer instance	
Parameters (In)	TimeValue	Time value in μs, the reference time has to be shifted.
Parameters (Inout)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Out)	None	
Return Value	None	
Description	Shifts the reference time of the timer instance.	
Available via	Tm.h	

Function Name	Tm_ShiftTimer100us32bit
Syntax	void Tm_ShiftTimer100us32bit(Tm_PredefTimer100us32bitType* TimerPtr, uint32 TimeValue)

Service ID [Hex]	0x13	
Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same timer instance	
Parameters (In)	TimeValue	Time value in 100 µs, the reference time has to be shifted.
Parameters (Inout)	TimerPtr	Pointer to a timer instance defined by the user.
Parameters (Out)	None	
Return Value	None	
Description	Shifts the reference time of the timer instance.	
Available via	Tm.h	

6.3.5 SyncTimer

Function Name	Tm_SyncTimer1us16bit	
Syntax	void Tm_SyncTimer1us16bit(Tm_PredefTimer1us16bitType* TimerDstPtr, const Tm_PredefTimer1us16bitType* TimerSrcPtr)	
Service ID [Hex]	0x5	
Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same destination timer instance	
Parameters (In)	TimerSrcPtr	Pointer to the source timer instance defined by the user.
Parameters (Inout)	None	
Parameters (Out)	TimerDstPtr	Pointer to the destination timer instance defined by the user.
Return Value	None	
Description	Synchronizes two timer instances.	
Available via	Tm.h	

Function Name	Tm_SyncTimer1us24bit

Syntax	void Tm_SyncTimer1us24bit(Tm_PredefTimer1us24bitType* TimerDstPtr, const Tm_PredefTimer1us24bitType* TimerSrcPtr)	
Service ID [Hex]	0xa	
Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same destination timer instance	
Parameters (In)	TimerSrcPtr	Pointer to the source timer instance defined by the user.
Parameters (Inout)	None	
Parameters (Out)	TimerDstPtr	Pointer to the destination timer instance defined by the user.
Return Value	None	
Description	Synchronizes two timer instances.	
Available via	Tm.h	

Function Name	Tm_SyncTimer1us32bit	
Syntax	void Tm_SyncTimer1us32bit(Tm_PredefTimer1us32bitType* TimerDstPtr, const Tm_PredefTimer1us32bitType* TimerSrcPtr)	
Service ID [Hex]	0xf	
Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same destination timer instance	
Parameters (In)	TimerSrcPtr	Pointer to the source timer instance defined by the user.
Parameters (Inout)	None	
Parameters (Out)	TimerDstPtr	Pointer to the destination timer instance defined by the user.
Return Value	None	
Description	Synchronizes two timer instances.	
Available via	Tm.h	

Function Name	Tm_SyncTimer100us32bit	
Syntax	void Tm_SyncTimer100us32bit(Tm_PredefTimer100us32bitType* TimerDstPtr, const Tm_PredefTimer100us32bitType* TimerSrcPtr)	
Service ID [Hex]	0xf	
Sync/Async	Synchronous	
Reentrancy	Reentrant but not for the same destination timer instance	
Parameters (In)	TimerSrcPtr	Pointer to the source timer instance defined by the user.
Parameters (Inout)	None	
Parameters (Out)	TimerDstPtr	Pointer to the destination timer instance defined by the user.
Return Value	None	
Description	Synchronizes two timer instances.	
Available via	Tm.h	

6.3.6 BuisyWait

Function Name	Tm_BusyWait1us16bit	
Syntax	Std_ReturnType Tm_BusyWait1us16bit(uint8 WaitingTimeMin)	
Service ID [Hex]	0x6	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	WaitingTimeMin	Minimum waiting time in microseconds
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected

Description	Performs busy waiting by polling with a guaranteed minimum waiting time.
Available via	Tm.h

Function Name	Tm_BusyWait1us24bit	
Syntax	Std_ReturnType Tm_BusyWait1us24bit(uint8 WaitingTimeMin)	
Service ID [Hex]	0xb	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	WaitingTimeMin	Minimum waiting time in microseconds
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	Std_ReturnType	E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected
Description	Performs busy waiting by polling with a guaranteed minimum waiting time.	
Available via	Tm.h	

Function Name	Tm_BusyWait1us32bit	
Syntax	Std_ReturnType Tm_BusyWait1us32bit(uint8 WaitingTimeMin)	
Service ID [Hex]	0x10	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	WaitingTimeMin	Minimum waiting time in microseconds
Parameters (Inout)	None	
Parameters (Out)	None	

Return Value	Std_ReturnType	E_OK: The underlying GPT driver service has returned E_OK and no development error has been detected E_NOT_OK: The underlying GPT driver service has returned E_NOT_OK, or a development error has been detected
Description	Performs busy waiting by polling with a guaranteed minimum waiting time.	
Available via	Tm.h	

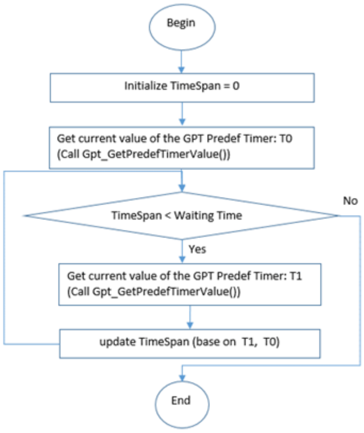
6.3.7 Note

- Parameter Wait Loop and Wait Loop Number is optional. It is used when user want to reduce the number of times Gpt_GetPredefTimerValue() is called in API Tm_BusyWait1us16bit(), Tm_BusyWait1us24bit(), Tm_BusyWait1us32bit() and Tm_BusyWait100us32bit(). Wait Loop shall be enabled before set value for Wait Loop.

- Parameter Wait Loop Number shall be set detail value when parameter TmEnablePredefTimer1us16bit or TmEnablePredefTimer1us24bit or TmEnablePredefTimer1us32bit is set "True" and Wait Loop is set "True". If not, the mobilgene Studio tool will report error when build project.

- The BusyWait services are based on polling, by the way calculate Time Span and compare it with target waiting time in the loop. The Time Span is calculated base on get current time value which is got from Gpt_GetPredefTimerValue(). So the accuracy of Time Span depend on execution time of pt_GetPredefTimerValue() and the accuracy of The value delivered by a Gpt_GetPredefTimerValue().

  Simple Flow chart of Waiting busy Service(of Tm_BusyWait1us16bit(), Tm_BusyWait1us24bit() and Tm_BusyWait1us32bit()) as below picture:


Begin
Initialize TimeSpan = 0
Get current value of the GPT Predef Timer: T0
(Call Gpt_GetPredefTimerValue())
TimeSpan < Waiting Time
No
Yes
Get current value of the GPT Predef Timer: T1
(Call Gpt_GetPredefTimerValue())
update TimeSpan (base on T1, T0)
End

When run Real time with below environment:

No.	Environment	Details

1	Microcontroller name	TC397A
2	Device id	NA
3	Evaluation Board name (external clock used)	NA
4	Compiler	tasking
5	Compiler Options	cctc.exe
6	Linker	--cpu=tc39x', '--create=object', '--iso=99', '--language=-gcc,+volatile,+strings', '--switch=linear', '--align=4', '--default-near-size=0', '--default-a0-size=0', '--default-a1-size=0', '-O2', '-OR', '--tradeoff=4', '-Wc--cache=$BUILDROOT/.cache', '-g'
7	Linker Options	ltc.exe
8	Trace 32	-D__CPU__=tc39x', '-D__PROC_TC39X__', '--core=mpe:vtc', '-OtXYcL', '-mCdfiklmNoQrSu', '--error-limit=42'

Execution time of Gpt_GetPredefTimerValue() when GPT Predef Time is GPT_PREDEF_TIMER_1US_16BIT then approximately 2us, when GPT Predef Time is GPT_PREDEF_TIMER_1US_24BIT or GPT_PREDEF_TIMER_1US_32 BIT then approximately 4us.

The value delivered by a Gpt_GetPredefTimerValue() function has an accuracy of +/- 1 tick (view SWS_Gpt_00384 in AUTOSAR_SWS_GPTDriver.PDF) (1 tick = 1us). So TimeSpan (see above flow chart) has an accurary of +/-2 us.

Tolerant of Waiting busy service = The accurary of TimeSpan + Execution time of Gpt_GetPredefTimerValue()
With :

The accurary of TimeSpan = +/-2 us
Execution time of Gpt_GetPredefTimerValue() = 2us or 4us (see above picture)

➔ Tolerant of API Tm_BusyWait1us24bit() and API Tm_BusyWait1us32bit() is approximately 6us. And Tolerant of API Tm_BusyWait1us16bit() is approximately 4us. So Actual waiting time is always greater than desired waiting time.

- The limitation of Waiting busy service is impacted by interrupt and difference of (Actual waiting time – Desired waiting time) not stable. Normally, difference of (Actual waiting time – Desired waiting time) is always equal to Tolerant of Waiting busy service (4us for Tm_BusyWait1us16bit() API and 6us for API Tm_BusyWait1us24bit() or Tm_BusyWait1us32bit() API). Sometime (not much) the difference of (Actual waiting time – Desired waiting time) may be greater than Tolerant of Waiting busy service.

7 Generator

7.1 Generator Option

Options	Description

-G,--Generation	Symbolic parameters to be used for fore generation (skip validation).
-H,--Help	Display this help message.
-I,--Input <I>	ECU description file path of the module for which generation tool need to run.
-L,--Log	Symbolic parameters to be used for generation error log.
-M,--Module <M>	Specify module name and version to be generated code for.
-O,--Output <O>	Project-relative path to location where the generated code is to be placed.
-T,--Top_path <T>	Symbolic parameters to be used for set path of module.
-V,--Validate	Symbolic parameters to be used for invoking validation checks.

7.2 Generator Error Message

ERR0600005 The parameter <parameter name> in the container <container name> should be configured.

This error message is displayed if the following parameters are not configured.

Parameter name	Container name
AR-RELEASE-VERSION	BSW-IMPLEMENTATION
SW-VERSION	BSW-IMPLEMENTATION
VENDOR-ID	BSW-IMPLEMENTATION

8 Appendix

None

9 Appendix

None