|  |  |
| --- | --- |
| Document Title | Specification of Timer0 Driver |
| **Document Owner** |  |
| **Document Responsibility** | Esraa El-Sayed El-Beltagy |
| **Document Identification No** |  |
| **Document Classification** |  |
|  |  |
| **Document Version** | 1.0 |
| **Document Status** |  |
| **Part of Release** |  |
| **Revision** |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Document Change History** | | |
| **Date** | **Version** | **Changed by** | **Change Description** |
| 16.04.2017 | 1.0.0 |  | Initial Release |

**Disclaimer**

This specification and the material contained in it is for the purpose of information only. The companies that have contributed to it shall not be liable for any use of the specification.

The material contained in this specification is protected by copyright and other types of Intellectual Property Rights. The commercial exploitation of the material contained in this specification requires a license to such Intellectual Property Rights.

This specification may be utilized or reproduced without any modification, in any form or by any means, for informational purposes only.

For any other purpose, no part of the specification may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

**Table of Contents**

[1 Introduction and functional overview 5](#_Toc511603712)

[1.1 Objective 5](#_Toc511603713)

[1.2 Context Diagram 5](#_Toc511603714)

[1.3 Acronyms and abbreviations 5](#_Toc511603715)

[2 External interfaces 6](#_Toc511603716)

[2.1 Std\_Types.h 6](#_Toc511603717)

[2.1.1 Types 6](#_Toc511603718)

[2.1.2 Interfaces 6](#_Toc511603719)

[2.1.3 Constants 6](#_Toc511603720)

[2.1.4 Variables 6](#_Toc511603721)

[3 Static Design 7](#_Toc511603722)

[3.1 File structure 7](#_Toc511603723)

[3.1.1 Used Files 7](#_Toc511603724)

[3.1.2 File inclusion 7](#_Toc511603725)

[3.2 Types 7](#_Toc511603726)

[3.2.1 Imported types 7](#_Toc511603727)

[3.2.2 Types Definitions 7](#_Toc511603728)

[3.2.2.1 7](#_Toc511603729)

[3.3 Symbol Definition 8](#_Toc511603730)

[3.4 Function definitions 8](#_Toc511603731)

[3.4.1 8](#_Toc511603732)

[3.5 Call-back notifications 9](#_Toc511603733)

[3.6 Scheduled functions 9](#_Toc511603734)

[4 Dynamic Design 9](#_Toc511603735)

[4.1 Mode Management 9](#_Toc511603736)

[4.2 Sequence Diagram 9](#_Toc511603737)

[5 Shared Resources 9](#_Toc511603738)

[6 Configuration specification 10](#_Toc511603739)

[6.1 Containers and configuration parameters 10](#_Toc511603740)

[6.1.1 Variants 10](#_Toc511603741)

[6.1.2 Parameters 10](#_Toc511603742)

[6.1.2.1 10](#_Toc511603743)

[7 Configuration Constraints 11](#_Toc511603744)

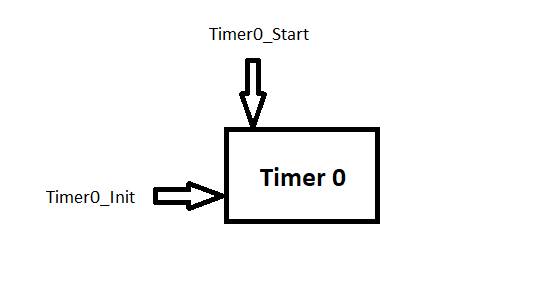
[8 Integration Constraints 11](#_Toc511603745)

# Introduction and functional overview

## Objective

The objective of this component is to use it by our operating system to count the ticks and help OS to decide which task should run.

## Context Diagram



## Acronyms and abbreviations

Acronyms and abbreviations that have a local scope.

|  |  |
| --- | --- |
| ***Abbreviation / Acronym:*** | ***Description:*** |
| OS | Operating System |
| Int | Integer |
| char | Character |
| Sync | Synchronous |
| Resync | Asynchronous |
| in | input |
| out | output |
| App | Application |

# External interfaces

## Std\_Types.h

### Types

|  |  |
| --- | --- |
| **Name** | **Description** |
| U8 | Unsigned char |
| U16 | Unsigned short Int |
| U32 | Unsigned long Int |
| f32 | Float |
| f64 | Double |
| S8 | Short char |
| S16 | Signed short Int |
| S32 | Signed long Int |

### Interfaces

None.

### Constants

None.

### Variables

None.

# Static Design

## File structure

### Used Files

|  |  |
| --- | --- |
| ***File*** | ***Description*** |
| Timer0.h | This file includes all prototype of function, #defines, extern constant and types of Timer0 which will be accessible to the other modules. |
| Timer0.c | This file includes all functions implementation and constant of Timer0. |
| Timer0\_cfg.h | This file includes all configurations for Timer0 module.  EX: initialization value of timer register, comparing value of timer register or prescaler. |
| Timer0\_register.h | This file includes all addresses of Timer0 registers. |

### File inclusion

Timer0.he

Timer0\_cfg.h

Timer0.ce

STD\_TYPES.h

Timer0\_register.h

## Types

None.

## Symbol Definition

None.

## Function definitions

This is a list of functions provided for upper layer modules.

### 

|  |  |  |
| --- | --- | --- |
| ***Service name:*** | Timer0\_Init | |
| ***Syntax:*** | Void Timer0\_Init (void); | |
| ***Service ID[hex]:*** | Timer0\_Service\_001 | |
| ***Sync/Async:*** | Sync | |
| ***Reentrancy:*** | Non reentrant | |
| ***Parameters (in):*** | None |  |
| ***Parameters***  ***(in/out):*** | None | |
| ***Parameters (out):*** | None | |
| ***Return value:*** | None |  |
| ***Description:*** | This function is used to initialize Timer0. | |

|  |  |  |
| --- | --- | --- |
| ***Service name:*** | Timer0\_Start | |
| ***Syntax:*** | Void Timer0\_Start(void); | |
| ***Service ID[hex]:*** | Timer0\_Service\_002 | |
| ***Sync/Async:*** | Sync. | |
| ***Reentrancy:*** | Non reentrant. | |
| ***Parameters (in):*** | None |  |
| ***Parameters***  ***(in/out):*** | None. | |
| ***Parameters (out):*** | None. | |
| ***Return value:*** | None. |  |
| ***Description:*** | This function is used to make Timer0 starts counting. | |

## Call-back notifications

This chapter lists all functions provided by the Timer0 module to lower layers.

Timer0 module does not provide any callback notifications.

## Scheduled functions

This chapter lists all functions called directly by the Basic Software Module Scheduler.

Timer0 module has no scheduled functions.

# Dynamic Design

## Mode Management

None.

## Sequence Diagram

None.

# Shared Resources

There are no shared resources between Timer0 module and other modules.

# Configuration specification

This chapter defines configuration parameters and their clustering into containers.

## Containers and configuration parameters

The following chapters summarize all configuration parameters.

### Variants

Configuration variants describe sets of configuration parameters:

* VARIANT-PRE-COMPILE (PC)

Only parameters with "Pre-compile time" configuration are allowed in this variant.

* VARIANT-LINK-TIME (LT)

Only parameters with "Pre-compile time" and "Link time" are allowed in this variant.

* VARIANT-POST-BUILD (PB)

Parameters with "Pre-compile time", "Link time" and "Post-build time" are allowed in this variant.

Parameters

### 

### Parameters

#### 

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | TIMER0\_u8\_COMPARING\_VALUE | | |
| ***Description*** | This macro is used to set the compare value of Timer0 register. | | |
| ***Multiplicity*** | 1 | | |
| ***Type*** | Scaler | | |
| ***Default value*** | 0 - (Range: 0-255) | | |
| ***Configuration Class*** | ***Pre-compile time*** | | |
| ***Configuration Class***  ***Scope / Dependency*** | ***Link time*** | X | All Variants |
| ***Post-build time*** | -- |  |
| scope: local | -- |  |
|  |  | | |

#### 

|  |  |  |  |
| --- | --- | --- | --- |
| ***Name*** | TIMER0\_u8\_PRESCALER | | |
| ***Description*** | This macro is used to set the prescaler value of Timer0. | | |
| ***Multiplicity*** | 1 | | |
| ***Type*** | Scaler | | |
| ***Default value*** | 1 - (Range: 1-5 ) | | |
| ***Configuration Class*** | ***Pre-compile time*** | | |
| ***Configuration Class***  ***Scope / Dependency*** | ***Link time*** | X | All Variants |
| ***Post-build time*** | -- |  |
| scope: local | -- |  |
|  |  | | |

# Configuration Constraints

* The number of bits of Timer0 peripheral is 8 bits so the value of initialization of timer must be in range from 0 to 255.
* The prescaler value must be from 1 to 5 (map to No prescaler, divide by 8, divide by 64, divide by 256 and divide by 1024).

# Integration Constraints

Timer0\_Start() function will be called once by app module at the start of the program.

**9 History**