Component Design Document

For

<OS Module>

**Table of Contents**

Revision History 1

*1. Introduction* 2

1.1 Objective 2

1.2 Context Diagram 2

2. *External Interface* 2

2.1 STD\_TYPES.h 2

2.2 Timer.h ………………………………………………………………………………………...2

3. *Static Design* 2

3.1 Files 2

3.2 Types 3

3.3 Symbol Define 3

3.4 Const 3

3.5 Interface (Services) 3

*4.* *Dynamic Design* 4

4.1 Mode Management 4

4.2 Sequence Diagram 4

*5.* *Shared Resources* 4

5.1 Analysis 4

5.2 Protection 4

6. *Configuration Parameters* 4

6.1 Pre-compile time 4

6.2 Link time 5

6.3 Post-build 5

*7.* *Configuration Constrains* 5

*8.* *Integration Constrains* 5

*9.* *History* 5

Revision History

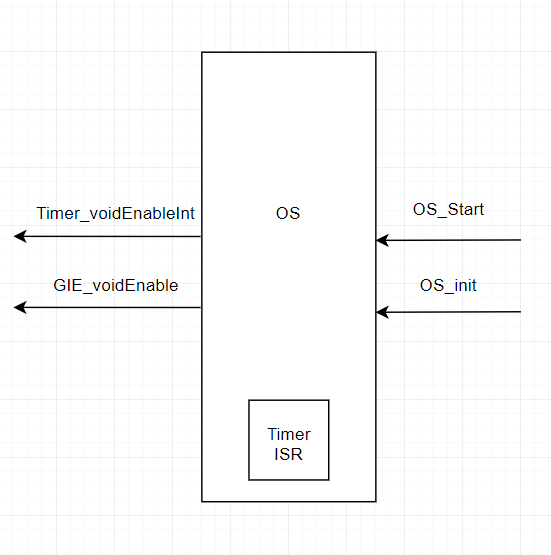
|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Youssef Medhat | 17/4/2018 | Created | 1.0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# *1. Introduction*

## Objective

The RTOS module is responsible of the management of the software different tasks and synchronization between them .It also control each task periodicity and start time.

## Context Diagram



# *External Interface*

## File name: STD\_TYPES.h

### Types

|  |  |
| --- | --- |
| U8 | unsigned char |
| U16 | unsigned short int |
| U32 | unsigned long int |

## File name: Timer\_int.h

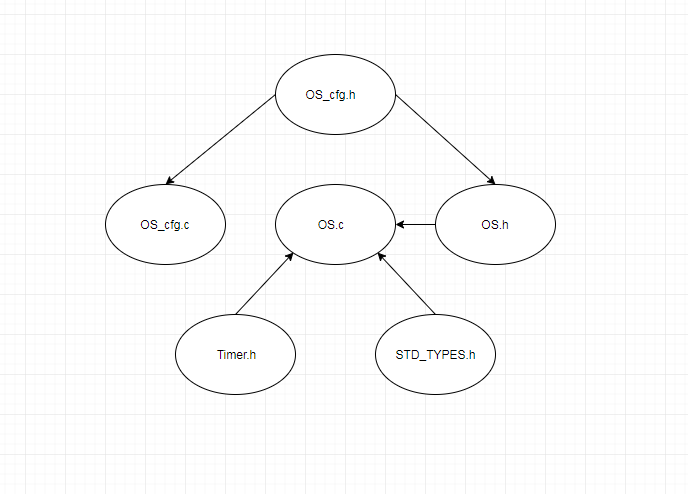
# Static Design

## Files

### Used Files

|  |  |
| --- | --- |
| FILE NAME | Description |
| OS.h | Contains all the public interfaces and functions to be available for anyone who uses the OS driver |
| OS.c | Contains the implementation of those interfaces and functions used in OS driver |
| OS\_cfg.h | Contains the different configuration that the user can use for the OS driver |
| OS\_cfg.c | Contains the implementation of those configuration selected by the user in the OS driver |

### Files Inclusion



## Types

|  |  |
| --- | --- |
| Requirement ID | OS\_001 |
| Covers | HLD\_001 |
| Name | OS\_cfg\_t |
| Type | Structure |
| Range | void (\*pftask)(void)  u16 periodicity |
| Description | It use to configure the specifications of each task which are the task function name and periodicity |

## Const

|  |  |
| --- | --- |
| Requirement ID | OS\_002 |
| Covers | HLD\_001 |
| Name | OS\_cfg |
| Type | OS\_cfg\_t array |
| Range | It’s an empty array |
| Description | It’s an array contains the configuration specs of each task and to be filled by the configurator |

|  |  |
| --- | --- |
| Requirement ID | OS\_003 |
| Covers | HLD\_001 |
| Name | NoOfTasks |
| Type | u8 |
| Range | Number of tasks |
| Description | Its describe the number of the task in the system to be configure |

## Interface (Services) OS\_int

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement ID | OS\_004 | | |
| Covers | HLD\_001 | | |
| Name/prototypes | void OS\_init (void); | | |
| Service ID | fr\_001 | | |
| Re-entrant / Non re- entrant | Non re- entrant | | |
| Synchronous/Asynchronous | Synchronous | | |
| Return Value | | void | No return value |
| Input parameter | | void | No input parameters |
| Output parameter | | void | No output parameters |
| Input /Output Parameter | | void | No input/Output parameters |

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement ID | OS\_004 | | |
| Covers | HLD\_001 | | |
| Name/prototypes | void OS\_Start (void); | | |
| Service ID | Fr\_002 | | |
| Re-entrant / Non re- entrant | Non re- entrant | | |
| Synchronous/Asynchronous | Synchronous | | |
| Return Value | | void | No return value |
| Input parameter | | void | No input parameters |
| Output parameter | | void | No output parameters |
| Input /Output Parameter | | void | No input/Output parameters |

# *Dynamic Design*

## Mode Management

## Sequence Diagram

# *Shared Resources*

## Analysis

## Protection

# *Configuration Parameters*

## Pre-compile time

## Link time

## Post-build

# *Configuration Constrains*

# *Integration Constrains*

You have to call the OS\_init before start running the OS.

# *History*

18/4/2018: Document Created