Small operating system scheduler (SOS)

By: Nada Abdelazim

HIGH LEVEL DESIGN

Layered architecture

*std types	**Application**				

LED	GInt	Push Button	SOS

HALL

DIO	TIMER	EXTI

MCAL

*std types

*macros and register memories
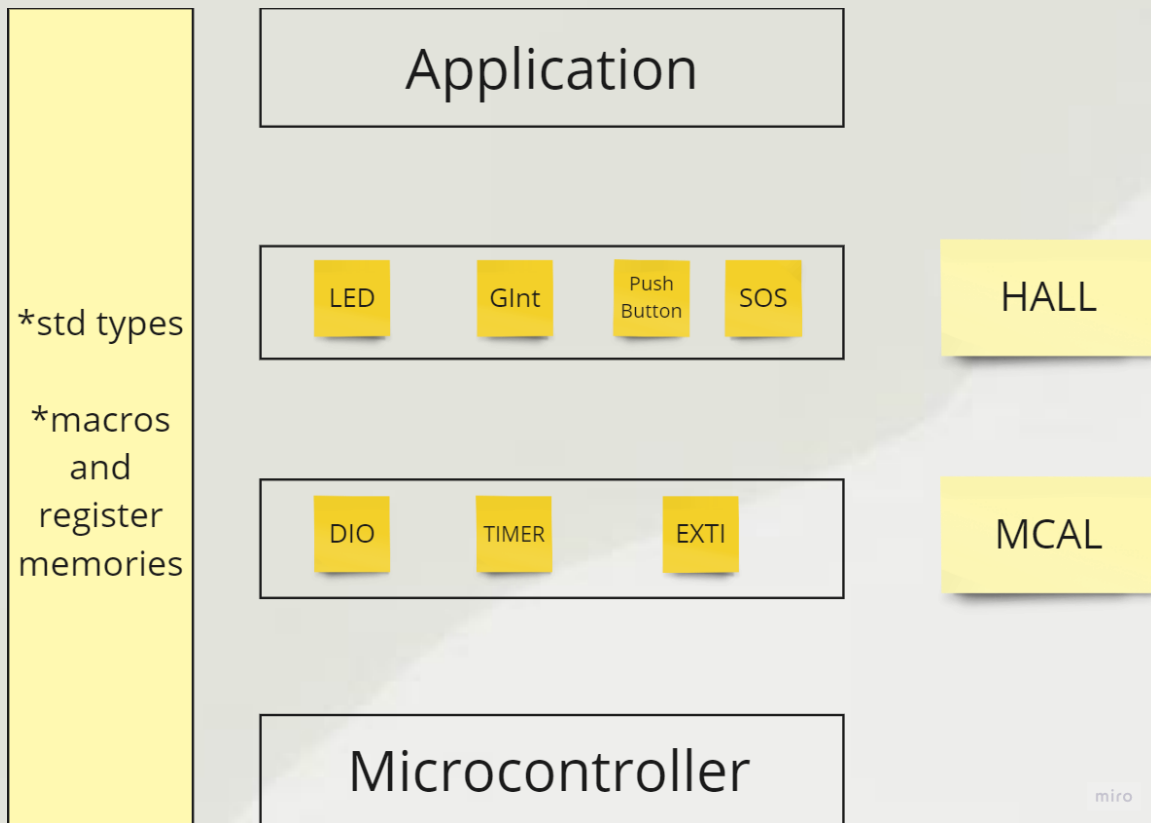
Microcontroller

Layers description:

(1)- Application layer:

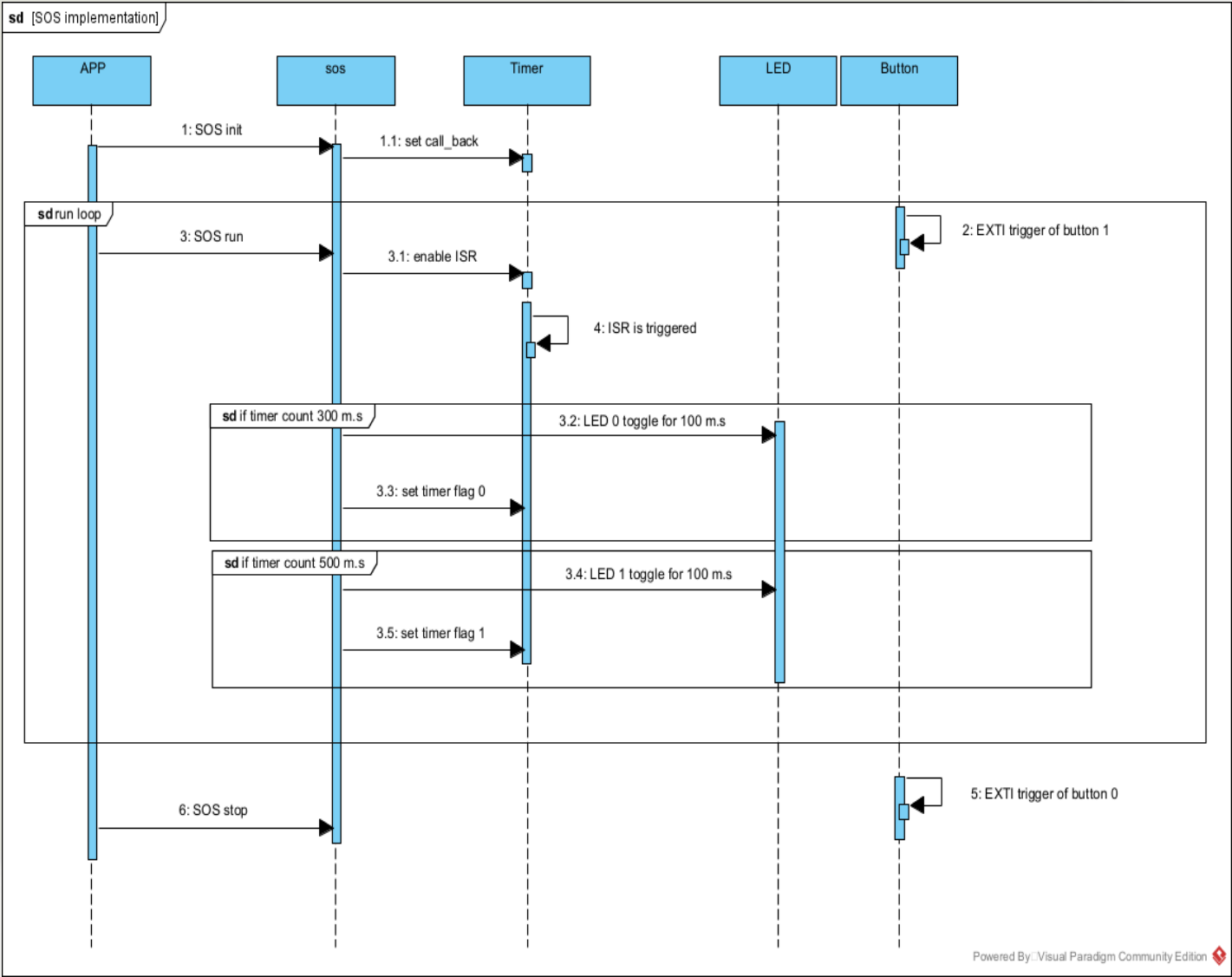Contains functions calls to implement the main project.

(2)- HAL:

Contains Drivers of the external electronic devices which will be connected to the microcontroller and the system overall.

(3)-MCAL: "Microcontroller Abstraction Layer"

Contains interfaces of the microcontroller's peripherals.

Sequence diagram for application

Class diagram for SOS

SOS

+sos_init (void):enu_system_status_t
+sos_deinit (void):enu_system_status_t
+sos_create_task(uint16,uint16,uint8,uint16):void
+sos_delete_task(uint16):void
+sos_modify_task(uint16):void
+sos_run:void
+sos_disable:void

LED

led_toggle(uint8): void

Push Button

pushButton_read(uint8): uint8*

Timer

-tick

set_callback
enable_ISR

Powered By Visual Paradigm Community Edition