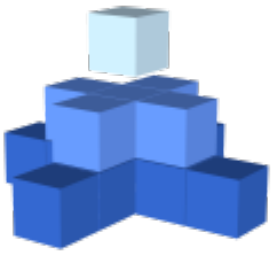
SRS ext.INT

System Requirements Specification



Status: Draft

Author: automatically generated from the SiSy model

Table of contents

1 Purpose

All elements of this project are parts of a course for the professional development of embedded systems. This Embedded Systems Engineering course is intended to develop a broad interdisciplinary understanding and knowledge of the participants as well as to develop practical skills for the realization of embedded systems.
The hardware platform for this course is the mySTM32 Board lite. It has a microcontroller of the STM32 family and all required input and output devices or add-ons.

2 Overall description of the task

A microcontroller application is to be developed in which an external signal, a falling edge, triggers an activity on the controller. The reaction of the controller to the external event should be realized by switching an LED.

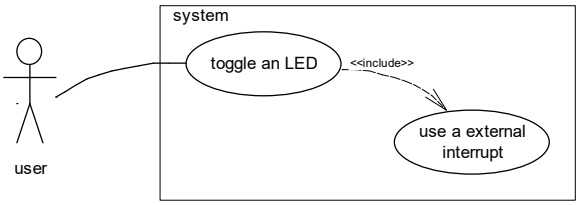


figure 1: uc: INT tasks, user's perspective

List of top level requirements:

- system : toggle an LED
- system : use a external interrupt

3 Functional requirements

After the system has started, the input for the key is configured to an external interrupt on a falling edge. As long as the system is switched on, the user can press the button. the LED toggles with each interrupt triggered by this.

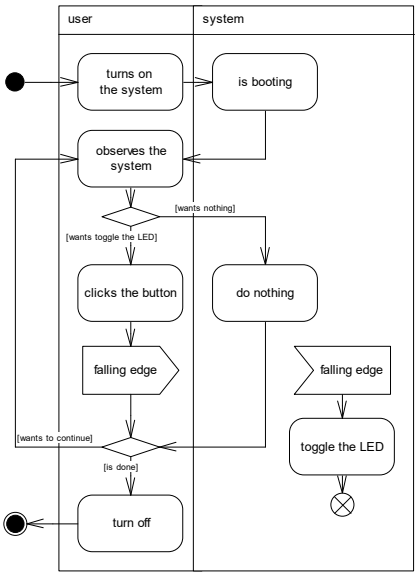


figure 2: toggle an LED

4 Hardware requirements

The hardware platform for this course is the mySTM32 Board lite. It has a microcontroller of the STM32 family and all required input and output devices or add-ons.

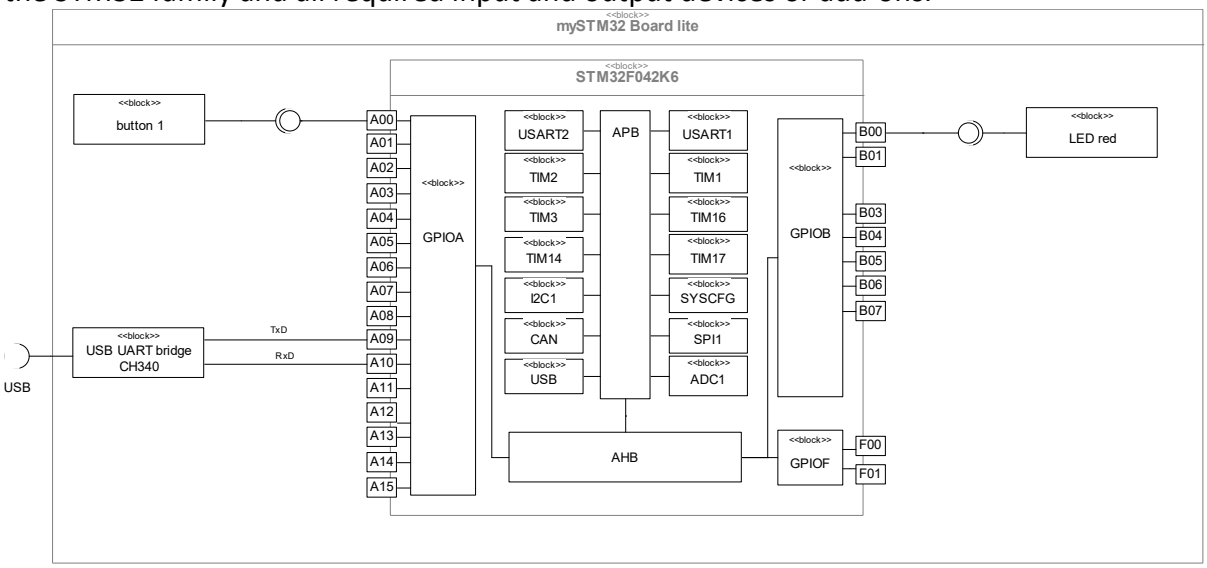


figure 3: INT HRM

- connected pinB0 : LED red
- connected USB
- connected pinA0 : button 1

5 Process requirements

A software process is the defined sequence of activities, the agreed rules, techniques, tools and the expected results of the activities for the production of software. Defined software processes ensure the plannability, controllability and quality of results in the manufacture of software. The following simple software process is agreed as a binding workflow for this course.

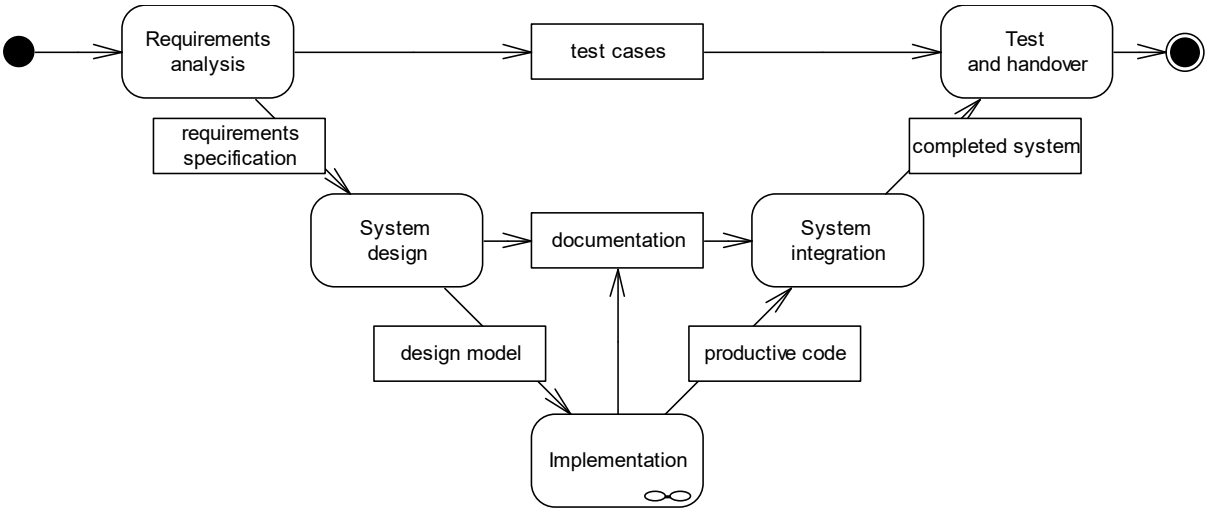


figure 4: act: lightweight model driven embedded software process

table 1: lightweight model driven embedded software process

Activity	Expected results
Requirements analysis	- User's perspective as use case diagram (as SysML / UML model) - required functionalities as activity diagrams (as SysML / UML model) - Test cases (as a document) - HRM hardware resource model (as SysML model) - SRS System Requirements Specification (as a document)
System design	- Class model of the concept level / architecture model (as UML model) - if necessary, state model (as UML model) - System documentation (as a document)
Implementation	- Class model of the realization (as UML model) - Behavioral models of the realization (as UML model) - Productive code (as a transferable format of the target platform, * .hex, * .elf) - System documentation (as a document)
System integration	- hardware software integration - the completed system
Test and handover	- the tested system - the technical system documentation (as a document) - the user documentation (as a document)

6 Attachment

List of figures

List of tables