

Customer Requirements Specification Univ Sw Project

Author: Gabriel Sofonea

Date: 24.02.2009

Current version: 1.0

Change-history:

Date	Ver.	Name	Change description	Chapter
24.02.2009	1.0	Sofonea	Initial version	all

Table of contents

1 ABSTRACT4

2 GLOSSARY5

3 REFERENCED DOCUMENTS6

4 SCOPE OF DOCUMENT7

1 Abstract

This document is the organization's understanding (in writing) of a customer or potential client's needs and dependencies *at a particular point in time* (usually) prior to any actual design or development work. It's a two-way insurance policy that assures that both the client and the organization understand the other's requirements from that perspective at a given point in time.

It will go through several stages during the course of the project:

1. **Draft:** The first version, or draft version, is compiled after requirements have been discovered, recorded, classified, and prioritized.
2. **Proposed:** The draft document is then proposed as a potential requirements specification for the project. The proposed document should be reviewed by several parties, who may comment on any requirements and any priorities, either to agree, to disagree, or to identify missing requirements. Readers include end-users, developers, project managers, and any other stakeholders. The document may be amended and re-proposed several times before moving to the next stage.
3. **Validated:** Once the various stakeholders have agreed to the requirements in the document, it is considered validated.
4. **Approved:** The validated document is accepted by representatives of each party of stakeholders as an appropriate statement of requirements for the project. The developers then use the requirements document as a guide to implementation and to check the progress of the project as it develops.

3 Referenced Documents

- [1] C163 16-Bit CMOS Single-Chip Microcontroller – User Manual
- [2] 16-Bit CMOS Single-Chip Microcontroller – Instruction Set Manual
- [3] KitCON-163 – Hardware Manual
- [4] SK-163 Starter KIT
- [5] "V-Model" – Continental Standard for Software development process

4 Scope of document

Scope of the document is to describe the customer requirements for the Univ Sw Project.

5 Requirements

5.1 General requirements

<USP_GEN_001> Application shall act as a slave with the external tester being the master (PC Application)



<USP_GEN_002> Serial Communication should be used to send messages between ECU and PC.

<USP_GEN_003> Target ECU: 16-Bit CMOS Single-Chip Microcontroller (delivered)

<USP_GEN_004> Project should be delivered as a package with all sources and documentation

5.2 ECU-Application requirements

<USP_ECU_001> C Language should be used with Keil uVision environment (delivered)

<USP_ECU_002> Application shall provide a communication interface via Serial

<USP_ECU_003> Use LED's from port P2.8 and P2.9 for Line-change Lights (left / right) and Warning lights.



<USP_ECU_004> The Line-change Lights LED should blink with a 1Hz frequency and 60% Dutycycle.

5.3 PC-Application requirements

<USP_PC_001> Supported Operating System shall be Windows 2000/XP

<USP_PC_002> C++ (VC) shall be used

<USP_PC_003> Application shall simulate commands from a real Car dashboard (Acceleration, Warning Lights, Line-change Lights and Warning Lamps)

<USP_PC_004> Acceleration shall be simulated with a TrackBar  component or EditSpin  Component

<USP_PC_005> Each time the acceleration, Line-change or Warning value is changed application shall send a message to the board with this value.

<USP_PC_006> Line-change Lights and Warning lights should be simulated with buttons (left / right)

<USP_PC_007> Dashboard display should be refreshed every 1 second with the current values from the board.

5.4 Quality requirements

<USP_QA_001> All source code should be commented according to Doxygen comment style (header for files, methods, classes) and indented

<USP_QA_002> "V-Model" Continental Standard SW development should be used in both parts of the modules.

Requested documents:

- Requirements Document (common for ECU and PC parts)
- Rough design document
 - i. Use case diagram
- Detail design
 - i. Sequence diagram
 - ii. Class diagram
 - iii. State machine diagram
- Integration tests (common for ECU and PC parts)
 - i. Test cases document
 - ii. Test report document