**Slave**

Debug

Michel Keijzers, © 2017

# History

Table 1: History

|  |  |
| --- | --- |
| **Date/period** | **Actions** |
| Nov 8, 2017 | Initial Version |

# Table of Contents

Contents

[History 1](#_Toc498978467)

[Table of Contents 2](#_Toc498978468)

[List of Tables 3](#_Toc498978469)

[List of Figures 4](#_Toc498978470)

[1 Introduction 5](#_Toc498978471)

[2 Requirements 6](#_Toc498978472)

[2.1 Design 8](#_Toc498978473)

# List of Tables

[Table 1: History 1](#_Toc498978474)

[Table 15: Requirements GS 5](#_Toc498978475)

# List of Figures

**No table of figures entries found.**

# Introduction

This document describes the Debug slave. It will receive debug messages from all (wireless) devices and send them to a PC (via USB).

# Requirements

## Generic

Table 39: : Requirements EG

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Base ID** | **Version** | **Category** | **Item** | **Description** |
| DbgG1 |  | 1.0 | Debug | Devices | Debugging from all devices will be possible, without using the normal communication (like RF).  *Rationale: using the normal communication might interfere the normal messages sent over it.* |
| DbgG2 |  | 1.0 | Debug | Amount | All devices can send debug messages to the Debug device.  *Rationale: Debugging should be possible over multiple devices.* |
| DbgG3 |  | 1.0 | Debug | Amount | The hardware RX pin should only be used for sending debug messages to the computer showing diagnostics messages.  *Rationale: Using the hardware RX pin for other purposes, would corrupt the messages.* |

## Hardware

TODO

## Software

TODO

# Inputs/Outputs

TODO

# Design

## Software Serial pin layout

Since only one directional communication is needed, the following pin layout will be used:

* Pin 0 (RX): For hardware debug port/serial
* Pin 1 (TX): For hardware debug port/serial
* Pin 2 .. 9: 8 software serial RX pins for 8 devices
* Pin 10..13 can be used for LEDs (TODO)

## Diagnostics LEDs

Table : Diagnostics LEDs

|  |  |  |
| --- | --- | --- |
| **Function** | **LED Color** | **Description** |
| Power | Blue (generic) | Off: Power off  On: Power on |
| RF | Yellow (generic) | Off: empty message transmitting/receiving  Slow blinking: contact with controller  Double fast blinking per second: no contact with slave  Triple fast blinking per second: problem with RF  On: non empty message transmitting/receiving |

Note that if the GUI Device shows errors whenever possible.

## Arduino Type

Since only one hardware RX is needed, and the functionality is rather simple, an Arduino Uno will provide enough.

## Breadboard Layout

The breadboard contains the LEDs as described in paragraph **Error! Reference source not found.**. This will take up a 400 holes breadboard or half 830 breadboard.

## Proto Layout

TODO

## Component List

TODO

# Software

## Design

To meet REQ DbgG3, for each device a Software Serial will be used. 8 can be used to debug 8 different devices. A loop will be made to iterate over all 8 software serial RX’s, and print it to the hardware serial.

## Memory Usage

Each software serial buffer uses 64 bytes, the hardware serial too, this means (8 + 1) \* 64 = 576 bytes for buffers.

## Timing Performance

When a lot of messages will be received, buffers can be full, however the debug messages are not critical.

# Testing

## Unit Tests

TODO

## Integration Tests

TODO