

Software Requirements Specification (SRS)

Table of Contents

1. 1. Einleitung	1
1.1. 1.1 Zweck des Dokuments	1
1.2. 1.2 Projektziel	2
1.3. 1.3 Begriffe und Abkürzungen	2
2. 2. Systemübersicht	2
2.1. 2.1 Hardware	2
2.2. 2.2 Systemkontext	2
3. 3. Funktionale Anforderungen	2
3.1. 3.1 Display Initialisierung	2
3.2. 3.2 SPI Kommunikation	2
3.3. 3.3 Grafikfunktionen	2
3.4. 3.4 Reset Handling	3
4. 4. Nicht-funktionale Anforderungen	3
4.1. 4.1 Timing	3
4.2. 4.2 Speicher	3
4.3. 4.3 Echtzeit	3
4.4. 4.4 Portierbarkeit	3
5. 5. Schnittstellen	3
5.1. 5.1 SPI Pins	3
5.2. 5.2 GPIO Pins	3
5.3. 5.3 Versorgung	4
6. 6. Randbedingungen	4
7. 7. Akzeptanzkriterien	4

1. 1. Einleitung

1.1. 1.1 Zweck des Dokuments

Dieses Dokument beschreibt die vollständigen funktionalen und nicht-funktionalen Anforderungen für das Projekt "STM32 Nucleo G431KB + Waveshare 1.51" Transparent OLED". Es dient als verbindliche Grundlage für Design, Implementierung und Test.

1.2. 1.2 Projektziel

Ziel ist die Entwicklung eines STM32-basierten Embedded-Systems zur Ansteuerung eines transparenten OLED-Displays über SPI.

1.3. 1.3 Begriffe und Abkürzungen

- MCU: Microcontroller Unit
- SPI: Serial Peripheral Interface
- GPIO: General Purpose Input Output
- SSD1309: OLED Display Controller
- HAL: Hardware Abstraction Layer
- SRS: Software Requirements Specification

2. 2. Systemübersicht

2.1. 2.1 Hardware

- STM32 Nucleo G431KB
- Waveshare 1.51" Transparent OLED Driver Board
- Versorgung: 3.3V / 5V

2.2. 2.2 Systemkontext

MCU steuert Display direkt über SPI. Kein weiteres Subsystem oder Host-PC.

3. 3. Funktionale Anforderungen

3.1. 3.1 Display Initialisierung

Das System muss: - SSD1309 korrekt initialisieren - Reset-Sequenz ausführen - Display in betriebsbereiten Zustand versetzen

3.2. 3.2 SPI Kommunikation

Das System muss: - SPI Master sein - Kommandos und Daten getrennt senden (DC Pin) - Chip Select korrekt steuern

3.3. 3.3 Grafikfunktionen

Das System muss folgende Funktionen bereitstellen: - SetPixel(x,y) - ClearScreen() -

DrawText(string) - Refresh()

3.4. 3.4 Reset Handling

Das System muss: - Hardware Reset Pin ansteuern - Software-Reset über SSD1309 unterstützen

4. 4. Nicht-funktionale Anforderungen

4.1. 4.1 Timing

- SPI Takt maximal 8 MHz
- Display Refresh ≥ 20 Hz

4.2. 4.2 Speicher

- Framebuffer maximal 1024 Byte
- Keine dynamische Speicherallokation

4.3. 4.3 Echtzeit

- Initialisierung < 100 ms
- Refresh non-blocking

4.4. 4.4 Portierbarkeit

Code muss portierbar auf andere STM32 sein.

5. 5. Schnittstellen

5.1. 5.1 SPI Pins

- MOSI \rightarrow DIN
- SCK \rightarrow CLK

5.2. 5.2 GPIO Pins

- CS: Chip Select
- DC: Data/Command
- RST: Reset

5.3. 5.3 Versorgung

- VCC: 3.3V oder 5V
- Logik: 3.3V

6. 6. Randbedingungen

- STM32 HAL Library
- Kein RTOS
- Keine DMA-Pflicht
- Kein externes Betriebssystem

7. 7. Akzeptanzkriterien

Das Projekt gilt als erfolgreich, wenn: - Display zeigt statisches Testbild - Textausgabe funktioniert - Keine Initialisierungsfehler - Kein sichtbares Flackern