Development of an Embedded DSP Eurorack Synthesizer Module

Jonas Bengt Lux

2025-11-13

Project Plan - Hardware Rev2

1 Hardware (Week 1)

- Order and assemble **Hardware Revision 2**.
- Verify correct assembly and power integrity.

2 Functional Testing (Week 2–3)

- Validate core functionality of the module:
  - **STM32 CubeMX**
    - Configure pinout and peripherals in STM32 CubeMX.
    - Set up project via CMake.
    - Research and include available drivers and libraries:
      - FreeRTOS
      - WS2812 (optional)
      - CMSIS DSP
      - Custom Audio Codec Driver (to be implemented)
  - **MCU**
    - Ensure device is flashable and debuggable via SWD.
    - Set up debugging environment and build system (CMake) in Visual Studio Code.
  - **Audio Codec**
    - Confirm accessibility and configuration via I^2C.
    - Test all audio inputs and outputs:
      - Route input directly to output to verify circuitry and correct AD/DA conversion.
    - Evaluate signal integrity for suitability in DSP development.
  - **Interface**
    - Confirm ADC readings from potentiometers.
    - Test WS2812 PWM control.

3 Audio Engine (Week 4)

- Implement usable I^2C drivers for the audio codec.
- Set up double-buffered audio I/O.
- Define and provide data structures for:
  - DSP algorithm integration.
  - Audio playback (recording/playback buffers).
- Implement playback engine (variable pitch/playback speed).
- Set up FreeRTOS tasks:
  - Audio task (and possible subtasks)
  - Control Interface Task
  - User Interface Task

4 DSP Development (Week 5–6)

- Implement real-time DSP algorithms (main module functionality, TBD).
- Initially, implement or use an IIR filter (available in CMSIS library).
- Implement basic granular synthesis functionality (time permitting).

5 Control Interface (Week 7)

- Implement control inputs:
  - CV inputs.
  - V/Oct input with calibration (calibration and validation of tracking if time permits).
  - Gate inputs.

6 User Interface (Week 8)

- Integrate and test faders (slide potentiometers).
- Implement RGB LED control.
- Handle button input with proper debouncing.