

GrandMA3 command wing MIDI-controller.

A title for the robot

A project in IN5590 by Oscar Halse E-mail: oscarha@ifi.uio.no

1 Intro

This is the initial prototype for a long-term project to create a midi-controller with the same physical layout as the "GrandMA3 command wing" lighting controller (hereafter "Wing").

The wing contains just 3 sensor-types:

- Buttons
- Rotary encoders with button (hereafter "encoder")
- Motorized slide potentiometers (hereafter "fader")

Thus the goal of this prototype is to:

- Create a functioning button-matrix that can be expanded to at least 80 buttons while still using the same MCU.
- Read encoder-increments as well as button-clicks of 5 encoders without missing any increments.
- Read position of 5 faders as well as be able to receive position-targets and move the faders to targets in what appears to be real time.
- The faders should have a 8-bit position resolution (same as faders in the GrandMA2/3 software)

If all these goals are met then expanding the prototype to the full Wing becomes as simple as copy&pasting the designed components as many times as required and designing a new chassis to contain them.

2 Tools

- PCB-design in KiCad
- 3D modelling of chassis in Solidworks
- 3D printing on Ender 3, sliced in Cura.
- Arduino programmed with PlatformIO extension to VS Code
- Raspberry pi Pico programmed with Raspberry Pi pico extension to VS Code
- GrandMA2 onPC as client

Main components:

- Terwin SM60N faders
- Terwin E12 encoders
- Cherry MX buttons
- Arduino micro for fader readuout and control
- Raspberry Pi pico for button and encoder read-out
- Custom PCB's for connections

Encoders and buttons required no special optimization. Faders use a round-robin refresh of velocity relative to targets with P-controller.

For MIDI-communication on Arduino the midiusb library was used. For the Pi the usb_midi module of the circuitpython library was used. Arduino was programmed in C++ while Pi was programmed in Micropython.

3 Conclusion

The first three goals were met in full, meaning that all sensors could successfully be read and fader positions set. But the current iteration of the firmware is only able to produce a 7-bit position-target resolution at any satisfactory (imperceivable) latency.

For the faders a P-controller was used rather than PID-control as PID showed no resolution improvements due to hysteresis being the resolution-limiting factor, not damping. Oscilloscope measurements showed that slack in the motor+pulley system caused an input-lag such that a pulse short enough to move at 8-bit increments would have to be stopped before the fader had begun to move. Motor reactance and belt-flex are assumed to be the main culprits. Future work will attempt to either create LUT's that can be used to create these micro-pulses to achieve the final bit of resolution. Alternatively add higher order or non-linear terms to a PI-controller (include squares, logs, etc. of the proportional and integral terms) .

Extra figures

GrandMA3 command wing



Physical prototype.



UNIVERSITY
OF OSLO

GitHub repo →

