Controller (New)

A new and improved controller for the programmable load, based around an STM32MP1 dual Cortex A7/M4F.

Overview

This is a new controller board for the programmable load, designed around the STM32MP1 microprocessor, which contains a Cortex A7 core (suitable for running a full-blown operating system) as well as a Cortex M4 core (suited to running real time sensitive tasks with an RTOS) on the same package. To avoid needing to deal with BGA packages, high speed DDR routing, and a bunch of power rails, a MYIR MYC-YA15X SoM is used.

Connectivity

As with the previous designs of the load, it will feature an USB device mode connection, as well as standard 100Mbps Ethernet with a standard IPv4 and IPv6 stack. Additionally, a CAN expansion bus is provided to connect multiple units together, and possibly to other, larger loads down the road.

Internally, the controller provides both a high-speed SPI for the analog interface (to allow fast control of ADCs and DACs by software) as well as a low-speed I²C interface, which is primarily intended for identification of the analog board, and some auxiliary control tasks like thermal management and input selection.

User Interface

Most of the user interface remains the same from the previous iteration, including the button and indicator layout on the front panel, barring some minor spacing changes.

However, the small greyscale OLED is replaced by a 800x480 4” capacitive touch LCD. This is driven directly by the LCD interface controller in the Cortex A7 side of the microcontroller, and provides an user interface that allows interaction by a hybrid means of the front panel buttons and touch controls.

Power

The switching AC/DC supply in earlier versions is replaced by a regular mains transformer, a standard rectifier and smoothing capacitors, followed by several DC/DC modules to generate the required voltages. Eliminating the AC/DC switching supply reduces the noise in the system, and provides greater isolation from mains.

Peripherals

Below are listed the general peripherals used by the controller. These are selected so that they consist of the peripherals that are available on pins broken out by the MYIR STM32MP1 SoM.

Cortex A7

These peripherals are reserved for the exclusive use of the Cortex A7 side. This runs a full OS, handles networking, expansion connectivity, the user interface, and so forth.

Communications

- I2C2: Front panel / local I²C bus
  - Through PCA9543A bus mux
- SPI3: Front panel display, CAN controller (through /CS-based mux)
  - Front panel display controller
  - On-board CAN controller
- UART4: Kernel TTY
- ETH: Ethernet MAC, in 100Mbps mode
  - PHY connected via RMII
  - STM32 generates 50MHz refclk for PHY
- USBH/USBOTG: Support for one USB host, one OTG device
  - Device exposes standard load interface
  - Support for boot over USB

GPIOs

- PB9: Front panel reset
- PC0: Ethernet PHY reset
- PC12: I2C2 mux reset
- PF6: Expansion connector power enable
- PF7: Expansion CAN termination
- PF9: Expansion CAN controller reset
- PZ6: Status LED (bicolor, 0/1/Z mode)

External IRQs

- PD2: I2C2 mux irq
- PA5: Ethernet PHY irq
- PC12: SPI3 irq

Cortex M4

These peripherals are reserved for the exclusive use of the Cortex M4 core, which runs the real-time control loop and a few other tasks better suited to an RTOS environment.

Communications

- I2C1: Analog interface (low speed)
  - Used for IDPROM, fan controller, temperature sensing, etc.
  - Runs at ~400kbps
- SPI5: Analog interface (high speed)
  - Runs at up to 20MHz
  - Up to 7 devices (3-bit select code)
- UART5: Debug console

Timers

- TIM2: Magnetic transducer (beeper)
  - PA3, Ch4 PWM output
- TIM4: Front LCD backlight
  - PB7, Ch2 PWM output

GPIOs

- PA12: SPI select bit 1
- PD8: Analog interface reset
- PD13: Status LED (green)
- PE11: SPI select bit 2
- PF8: Status LED (blue)
- PG5: Status LED (red)

External IRQs

- PG3: Analog interface
- PA4: Front panel encoder “A”
- PA5: Front panel encoder “B”
- PG8: External trigger
- PF10: Zero crossing detect

Hardware Errata

This page documents some issues with the hardware.

Rev 3

- Footprint for RTC backup battery (BT301) is backwards
  - The + terminal of the battery connects to the GND node, and vice versa.
  - Resolution: Populate battery in reverse
- RTC charger should be powered by system +3V3 rail
  - It's powered by the SoC's own 3V3 rail now, which toggles off during reboot
  - Use instead the system +3V3 rail
- Incorrect power-on reset behavior
  - ~~There is no reset pulse generated on boot-up (by an external reset generator... which we don't have. lol) which causes spurious boot failures~~
    - ~~So, we should add a supervisor for the reset line~~
    - ~~Generate a power on reset pulse of ~1 sec with the +5V rail~~
  - ~~We may need to shorten the reset delay on the +3V3 voltage supervisor line~~
    - ~~Alternatively – directly use the +3V3 input to switch it~~
  - This was caused by the RTC charger coming off the +3V3 SoM rail. Using the system +3V3 rail resolves this
- Pulse shaper stuff
  - Adjusting it sucks as TP501 is on the *analog* side of the filter, rather than the nice, pristine digital from U507
    - Fix: Add another test point at the output
    - Alternatively, we can probably work around this in software (with a special calibration UI?)
  - Possibly expand the adjustment headroom
- Front panel connection
  - The pinout of the connector is mirrored (left to right) with respect to the front panel when assembled
  - We can probably leave this (using a longer flex to compensate) or fix it on either the front panel or controller board revision (probably the controller board)
- Rear case fan (M501) connector sucks
  - The location is awful (it will be right underneath the analog board)
  - Relocate it elsewhere - but where?
    - Where VBUS LED is chilling
    - In the back by the expansion connector
      - We'd need to relocate some of the expansion IO stuff, like the termination and so forth
- Power supply section sucks to assemble
  - Is there anything we can do to make this less awful

- Also, add an easier way to power the board off DC
  - Spring/screw terminal near power supply area