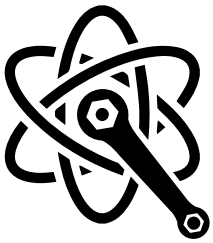

ARTIQ
Status report

January 4th, 2018

1 Sayma

1.1 Flash

We have fixed the flash problems on Sayma. The boards now start correctly upon power-up (minus the RTM FPGA that still requires the development of a loading mechanism, see `https://github.com/m-labs/artiq/issues/813`), and idle and startup kernels can be used by loading them using `artiq_mkfs` and `artiq_flash`. We were able to run an ARTIQ kernel on Sayma with no particular problem and toggle TTL lines.

1.2 DRTIO

We cleaned up and integrated the DRTIO Ultrascale GTH transceiver code with ARTIQ and managed to repeatably obtain a 3Gbps DRTIO link between two Sayma boards using the SFP. This has, however, only received limited testing since the debugging cycles on Sayma are particularly tedious (1.8V power supply bug shutting down the boards often and at random times, long bitstream loading times, no Ethernet).

We set up and used the Si5324 clock recovery circuit on Sayma with the ARTIQ code. We found no issues with the Si5324 and its ARTIQ driver.

The next step is to implement support in the transceiver code for multiple synchronized links on one DRTIO master, so that more than one DRTIO satellite can be connected to the master.

1.3 Ethernet

We investigated the issues with Ethernet and we managed to receive corrupted packets using the RGMII interface. We believe the corruption is due to poor I/O timing, and that this timing is difficult to fix with RGMII due to the incorrect use of a non-clock-capable pin on the FPGA for the RX clock input. We are investigating several routes to make Ethernet work, some of them involving board rework and/or MMC firmware reflashing (e.g. to turn the interface into MII which has better timing margins).

`https://github.com/m-labs/artiq/issues/854`

1.4 serwb

We have implemented scrambling and did some clean-up of the code. The unrelated initialization issue persists.

`https://github.com/m-labs/artiq/issues/856`

1.5 RF generation

Using the startup kernel mechanism (and later, a simple hardwired gateware ramp generator that reduces complexity and compilation times), we have attempted to output RF from Sayma and BaseMod (Allaki) using ARTIQ. We have not succeeded so far, even when the initialization of the full stack completes (RTM FPGA bridge, HMC830, HMC7043, AD9514). We are investigating the problem.

The remaining path to a usable Sayma stand-alone RF generator is:

1. fix the Ethernet problem. We are waiting for WUT to make the Sayma PHY operate consistently in MII mode, and/or for an ARTIQ test from them using their PHY that they managed to operate in MII mode.
2. fix the serwb intermittent initialization failures. This is likely a gateware problem and, until the bug is fixed, it can be worked around (to some extent) by putting the RTM FPGA loading under AMC control, and having the firmware retry (`https://github.com/m-labs/artiq/issues/813`).
3. fix the HMC830 intermittent locking failure (`https://github.com/m-labs/artiq/issues/860`). This may or may not be a setup problem (i.e. not a Sayma or ARTIQ bug) and it needs more testing.
4. set the attenuator on BaseMod to a sensible value. This can be done with a small addition to the ARTIQ runtime.
5. when the full stack initialization succeeds, understand why no RF is generated, and fix the problem.

1.6 Call for testers

It would generally be beneficial to the project if multiple groups attempted the tests, to rapidly spot one-off hardware issues or setup problems. To facilitate this, we have built up-to-date conda packages for the ARTIQ Sayma stand-alone platform. Flashing those binaries into Sayma and then loading the RTM FPGA manually using OpenOCD would enable the ramp generator (`https://github.com/m-labs/artiq/commit/745e695b0`) after the ARTIQ runtime starts and initializes the system.

Furthermore, those packaged binaries implement a standard MII PHY core for Ethernet, which has been shown to work on other boards such as the KC705. After reflashing the MMC firmware to make the Ethernet PHY operate correctly in MII mode (which WUT managed to do, but could not be reproduced), Ethernet theoretically would function.

Support for installing ARTIQ, flashing Sayma, and loading the RTM FPGA is available on the mailing list and IRC.

2 Kasli

The first Kasli boards have been produced and are mostly tested. Apart from minor issues (FTDI level translators, weak pull up, one transceiver not yet working correctly) there are no road blocks. After some work, MiSoC became functional on Kasli (flash, SDRAM, UART).

We have also developed the transceiver code for the 1000Base-X PCS/PMA/PHY that will enable Ethernet on this board, and integrated it. It compiles, but still needs testing and debugging on the hardware.

3 Urukul

Urukul has seen extensive testing including large parts of the functionality and performance. The behavior of the RF chains (including RF switches, RF attenuators and output power) meet or exceed specifications.

The phase noise performance is excellent and mostly as expected except for a ~kHz phase noise spur (at -75 dBc) that is being investigated. Work and integration of the ARTIQ drivers continues for both the AD9910 and AD9912 variants.

4 Other hardware

- The new revision of Sampler (Novo) prototypes (with the faster ADC) will arrive in a few days.
- Revision 1.1 of Urukul is going into production in a few days.
- The Booster (RF-PA) prototype was shipped.
- The Clocker module is being tested, some were already shipped.
- The MTCA mini-chassis prototype was produced, and is being tested.

5 ARTIQ-3 bugfix releases

5.1 ARTIQ 3.1

We have released ARTIQ 3.1, which contains important fixes for the ARP storm problem that arised due to the security scan appliance on the NIST network, and the Ethernet switch packet corruption issue.

5.2 ARTIQ 3.2

We will soon release ARTIQ 3.2, which contains further Ethernet improvements regarding throughput and lockup issues, backtrace code to make identifying runtime crashes easier, and a SPI fix (during testing and integration of Urukul we identified glitches on the SPI clock line at the start and end of the SPI transfer).

`https://github.com/m-labs/microscope`

6 New Focus 8742 piezo motor controller driver

We have developed a Python/ARTIQ driver layer for the New Focus 8742 piezo motor controller.

7 Development tools and libraries

7.1 Microscope FPGA logic analyzer

To make debugging FPGA designs easier, we have developed a small on-chip logic analyzer that integrates well with Migen. This was very useful to make DRTIO work on Sayma.

7.2 Core device backtraces

To aid in debugging Rust panics and (if any arise due to use of unsafe code) CPU exceptions, the core device now includes an unwinder that produces backtraces in these conditions (which will be included in ARTIQ 3.2). The backtraces are not yet symbolized; symbolizing them will be done by transitioning the runtime from a binary to a fully fledged ELF file, and embedding a symbolizer in the runtime. This requires the bootloader to be able to load ELF files, which is not yet done, and will not be part of ARTIQ-3.

7.3 New bootloader

All board support code has been ported to Rust from C, and a new ARTIQ bootloader has been written from scratch to replace the MiSoC BIOS. The ARTIQ bootloader only has a bare minimum of functionality right now (flash boot and network bringup), but will be extended to support boot over both serial and TCP using the same protocol (instead of boot over serial and UDP in MiSoC using different protocols), as well as provide post-mortem analysis functionality: a way to retrieve the core log before the last core device panic (or warm reboot), and a GDB server interface for exploring the final state of the runtime with a debugger.

7.4 mor1kx tightly-coupled memories (TCM)

To make deeply-embedded mor1kx implementations more efficient, we have started modifying the CPU core to include TCM support. This will be used in the future to operate the RTM FPGA on Sayma as a DRTIO leaf node running a minimal implementation of the satellite manager firmware.