

MNT Reform Kintex-7 Module (RKX7)

Dear MNT Customer,

thank you for purchasing MNT RKX7, an open and customizable system-on-module for building portable interactive computing applications with FPGA technology. We hope you will enjoy it.

Package Contents

- MNT RKX7 module with 2mm MODE2 jumper
- 6-pin JTAG DuPont adapter cable
- Optional: RGB to eDP adapter module with ANX9807 chip and 50-pin split parallel RGB FPC cable

Installation

Push the module into the CPU slot of an MNT Reform motherboard at a 45 degree angle and then press it down until it clicks into place.

To interface RKX7 with the standard display panel of an MNT Reform laptop, connect the parallel RGB to eDP adapter PCB to the FPGA module using the included 50 pin FPC cable. Then, connect the display's DuPont cable to the eDP adapter's dual row pin header. Ensure the correct orientation of the DuPont plug: a white dot on the plug marks pin 1, which has to match the position of a break in the silkscreen rectangle of J2 on the adapter.

Boot

Jumper J3 on RKX7 breaks out the MODE2 pin of the Kintex-7 with the option to set it to high or low depending on the jumper position. By default, MODE1 is pulled low by R83 and MODE0 is set high by R79 (01). You can change these bits by populating other resistors in this group (see "Boot Mode Jumper" section in the schematic).

Normally, you only need to toggle between JTAG or SPI boot by toggling MODE2 via left (low) or right jumper (high) setting on J3, respectively, yielding $M[2:0] = 001$ or 101 . For details, see Xilinx UG470 page 21.

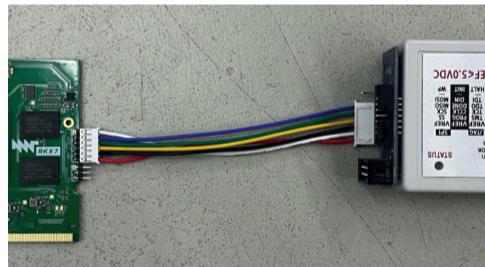
JTAG Workflow

You can load gateway .bit files into the FPGA via the JTAG port and a JTAG adapter and tool of your choice, for example the Xilinx Platform Cable USB and xc3sprog:

```
xc3sprog -v -c xpc build/mnt_rkx7/gateway/mnt_rkx7.bit
```

For convenience, a JTAG breakout cable is included with RKX7. You can connect the 2mm pitch side to J2 (white line, GND, goes to pin 1), and the 2.45mm pitch side goes to your JTAG adapter.

Color	Function
White (Pin 1)	GND
Blue	TMS
Green	TDI
Yellow	TDO
Black	TCK
Red	VREF (3.3V)



Warranty

MNT Research covers the warranty to the extent required by law for the unmodified product. Warranty does not cover any modifications made by the user. Warranty does not cover damage resulting from not following exact assembly steps. Use at your own risk!

Conformity

The product is compliant with relevant European Union harmonisation legislation.

Safety Instructions

- The module is a highly sensitive electronic part that can be damaged by static electricity. Make sure to ground yourself before touching.
- RKX7 is not a consumer device. It is intended for research and development purposes and should only be handled by professionals.



Specifications

- FPGA: AMD/Xilinx Kintex-7 XC7K160T-FFG676 or XC7K325T-FFG676
- Memory: 2GB DDR3 (2x Micron IS43TR16512B-125KBLI in 32-bit topology)
- 100 MHz main clock oscillator
- HDMI 1.4 (Full HD) implemented via Sii9022 encoder
- 1 GBit Ethernet via KSZ9031RNX PHY
- 3x UART, 2x I2C, 1x SAI, 1x PWM backlight ports connected to FPGA fabric
- 128 MBit configuration flash (Winbond W25Q128JVSIM)
- JTAG header
- 6 synchronous buck converters powered by single 5V input, 1-3A
- Size: 68x67.5mm
- 200 pin SO-DIMM card edge (mates with TE 1717254-1 or TE 1473005-1)

LiteX RISC-V Demo Platform

As a demo, we provide a LiteX-VexRiscV 100 MHz 32 bit RISC-V system. You can find this demo under `bitfiles/mnt_rkx7_litex_demo.bit` and load it into the FPGA using the procedure described above.

You can interact with the LiteX BIOS by connecting a UART to USB adapter (3.3V) to the UART1/SER1 pins on your carrier board (i.e. MNT Reform motherboard). The default baud rate is 115200. From the BIOS, you can run memory tests or boot a 32 bit RISC-V Linux via Ethernet (TFTP, netboot command) or SD card (`sdcardboot` command).

You can find prebuilt Linux kernel and Device Tree in the `tftp` folder of the RKX7 repository. You can serve the files with `uftp`, i.e.: `sudo uftp -n images`

After a successful boot, the Linux console will appear on your UART connection and you can log in as root with no password. You can connect to the network by using `udhcpd eth0`, mount an SD card and download additional files using `wget`.

The demo system includes a Full HD framebuffer (1920x1080, 32 bit) and video output that uses the 50-pin expansion connector for parallel RGB and sync signals. With the RGB to eDP adapter, you can connect an eDP panel. The demo is geared towards the display panel included with the MNT Reform laptop.

Modifying and building the LiteX SoC

If you want to build VexRiscV, LiteX, LiteX BIOS and Linux kernel from scratch, you can follow the Linux-on-LiteX-VexRiscV installation instructions:

<https://github.com/litex-hub/linux-on-litex-vexriscv>

We use a small patch for the LiteX BIOS that (de)asserts a number of GPIOs at startup (reset line for the USB hub on the motherboard and reset line of the eDP converter chip, if connected). The patch is located in the `patches/litex` directory.

The board and target files for MNT RKX7 are included in LiteX and named `mnt_rkx7`. The default FPGA target is the XC7K325T. If you have the XC7K160T, change it in the file `litex-boards/litex-boards/platforms/mnt_rkx7.py` in the line mentioning `Xilinx7SeriesPlatform`.

The platform file is also a useful reference for the pad/signal assignments and I/O voltages. You can of course customize these to your liking. It is recommended to enable the USB host functionality by setting `default=True` in the line containing "Enable USB host support" in `litex-boards/litex-boards/targets/mnt_rkx7.py`.

Finally, you can build the SoC like this (exchange the IP address with your TFTP server address, and you can also increase the number of CPU cores or add an FPU):

```
cd linux-on-litex-vexriscv
./make.py --board=mnt_rkx7 --build --remote-ip 192.168.1.100 --aes-instruction 1 --cpu-count 1 --with-coherent-dma
```

Device Tree

The device tree generated by Linux-on-LiteX-VexRiscV needs some additions for RKX7 peripherals. After customizations to your LiteX SoC, you should rebuild the `rv32.dtb` Device Tree file. You can use the `mnt_rkx7_full.dts` wrapper which includes, modifies and extends the default `mnt_rkx7.dts` generated by LiteX like this:

```
dtc -o images/rv32.dtb mnt_rkx7_full.dts
```

eDP

The parallel RGB to eDP converter module employs the Analogix ANX9807 chip. Linux includes a driver for the related ANX6345, but it needs some changes to work with the ANX9807. These are included in the `patches/linux` directory. If you want to compile the Buildroot Linux included in Linux-on-LiteX-VexRiscV, you have to apply the patches to enable the eDP converter and MNT Reform display panel.

HDMI

Alternatively, RKX7 features a RGB to HDMI chip (Sii9022A) that you can use to drive an HDMI display. To initialize the chip, you can use the Linux driver and device tree, but an easier way is to use the `sii9022a-init.sh` script that sets up the registers using the `i2cset` tool.

SPI Flash

RKX7 features a W25Q128JVS SPI flash chip for automatic FPGA configuration memory for the FPGA. To store your own data in the flash, you need to go through gateway/software on the FPGA or Linux. Reference the "Boot" section for SPI boot jumper settings.

Get the sources at: <https://source.mnt.re/reform/reform-kintex-som>

Need help? Join our community forum: <https://community.mnt.re>

Or join our IRC channel `#reform` on `irc.libera.chat`

