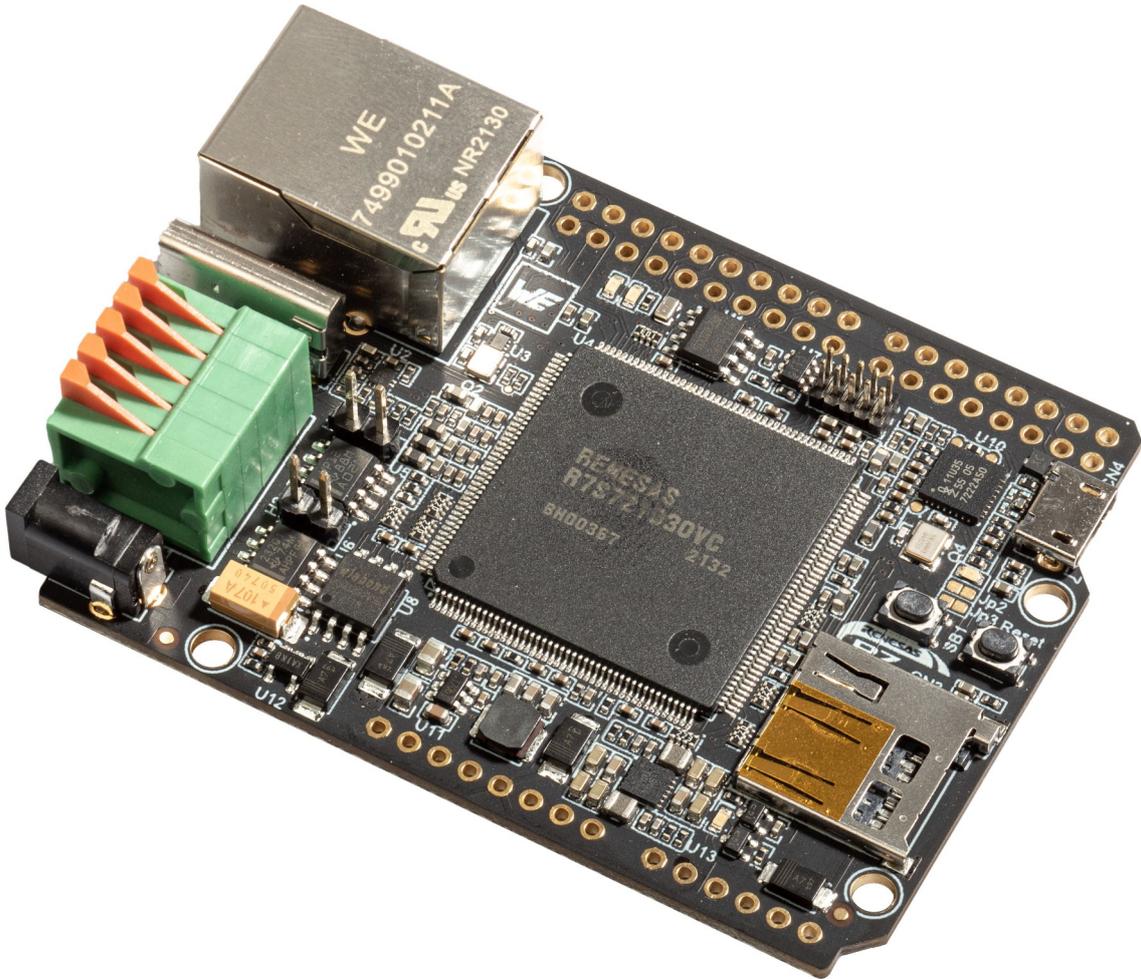


VK-RZA1R3 Mbed App Demo



VK-RZA1R3 v1.1 Board



Developer's manual

Content:

1. INTRODUCTION.....	3
2. ENVIRONMENT SETUP.....	3
3. BUILD.....	4
4. DEMO.....	4



Developer's manual

1. Introduction

VK-RZ/A1R3 (a.k.a VK-RZ/Arduino1 rev.3)

Is actually a development board, based on **Renesas RZ/A1LU ARM Cortex-A9 LSI**. (more info [here](#))
The main purpose of this application is demonstration of VK-RZA1R3 board's hardware capabilities and main components workability (USB, Camera, Ethernet, SD Card, BTNs, LEDs etc.)

2. Environment Setup

This demo is built on top of **mbed OS 6.12.0**, so at leased, you need to have [Mbed CLI 1](#) tools installed on your system (& **E²Studio** too, if you want to debug it). After you are sure **mbed** is recognized from the command line (**cmd**), you are good to go download mbed OS's source code from github & add some libs too, before building the demo itself.

To do this, follow the instructions:

- `hit win+R, type cmd, hit Enter` Open cmd.exe.
- `cd \D D:/Projects` Go to the download folder (let's say D:/Projects).
- `mbed new Demo` Download mbed-os by giving name (let's say Demo).
- `cd Demo/mbed-os` Go to the mbed-os folder
- `mbed update mbed-os-6.12.0` Downgrade current mbed OS version to 6.12.0
- `cd ..` Get back to the project
- `mbed add https://github.com/Vekatech/mbed-vk-boards`
Add VK-RZA1R3 board as custom board to mbed OS
- `mbed-vk-boards\patch\patch.py` Patch some system mbed OS files, not modifiable by custom boards technique
- `mbed add https://github.com/Vekatech/mbed-vk-libs`
Add VK-RZA1R3 board's bsp a.k.a VK-libs
- `mbed-vk-boards\demos\extract.py VK_RZ_A1R3`
Get the actual Demo finally

Now you are ready to produce bin file & flash it to the board, to see what the demo actually does. Skip this step [2] and use the precompiled bin (located in `BUILD\VK_RZ_A1R3\E2_GCC_ARM\Demo\Release\Demo.bin`) if you just want to see it and you are not interested in compile & debug.



Developer's manual

3. Build

You have 2 options. It can be done with **Mbed CLI 1** or with **E²Studio**.

If you prefer the mbed tools, follow the instructions:

- `hit win+R, type cmd, hit Enter` Open cmd.exe.
- `cd \D D:/Projects/Demo` Go to the demo folder.
- Modify `mbed_settings.py` file Fill the correct paths to the compilers (from the right side of the `ARM_PATH` & `GCC_ARM_PATH`)
- `mbed compile --custom-targets mbed-vk-boards -t GCC_ARM -m VK_RZ_A1R3` Produce the bin file.
- Drag & drop `BUILD\VK_RZ_A1R3\GCC_ARM\Demo.bin` to the MBED disk
Flash the bin file.

If you prefer E²Studio, follow the instructions:

- There is setuped eclipse project located in `BUILD\VK_RZ_A1R3\E2_GCC_ARM\Demo`.
- Open it with **E²Studio** and hit Build Produce the bin file.
- Hit Debug Flash the bin file & debug the Demo.

4. Demo

The App counts on camera module OV7725 to be plugged in to the board on start up. After downloading `Demo.bin` to the VK-RZ/A1R3's flash and clean start (Reset), this program waits plugging of Ethernet cable in to the RJ45 connector. When that happens, the board will wait, trying to take an IP from the DHCP server. After network is setuped, an micro RTSP server will be started waiting for clients. In case of a client, the red LED will light up (on AIR) and server will start broadcasting MJPEG stream with resolution 640x480. If micro SD card is inserted in to the slot, the green LED will light up. If USB C cable is connected, the board will turn to a 64K mass storage device and the host (PC) will mount it. With pressing SB1 user button, you can take pictures and they will be saved on SD card & mass storage device, (if card exists in the slot or host is found at the other end of the USB C cable) SD card will keep all taken pictures, but MSD will save only the latest one. Instead of guessing what's happening & what Demo doing, you can watch the mbed serial log as it is showed in following screenshots.



Developer's manual

```
COM4 - PuTTY
***** PROGRAM START *****
Network Setting up...

Connecting...
MAC Address is 00:02:f7:f0:00:00
IP Address is 192.168.2.163
NetMask is 255.255.255.0
Gateway Address is 192.168.2.254
Network Setup OK
running RTSP server
Saving Pic_25322[32885]
Saving Pic_27759[32849]
Saving Pic_30363[32854]
```

Mbed system console log

Here are some examples how to grab the stream from the RTSP server:

- Using VLC as a player:

Go to: "Media" → "Open Network Stream..."

for "URL" enter this: `rtsp://VK_RZ_A1R3's_IP:8554/mjpeg/1`

Check "Show more options" & in "Edit Options" add this: `:network-caching=0`

Press Play.

- Using FFMPEG as a player:

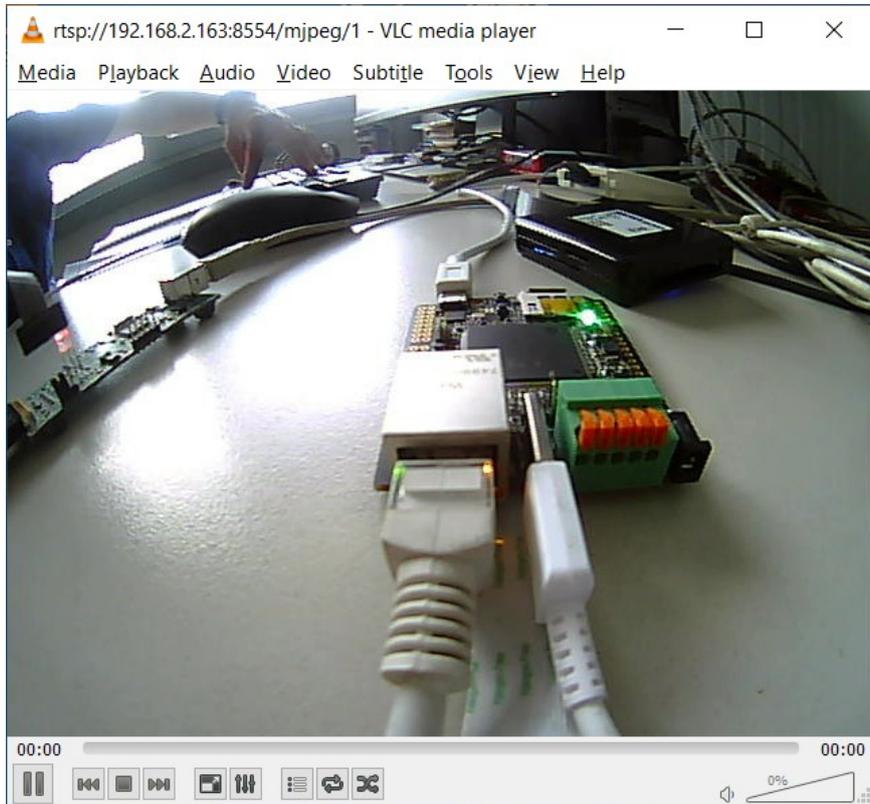
Open a CMD/BASH terminal

Insert: `ffmpeg rtsp://VK_RZ_A1R3's_IP:8554/mjpeg/1`

A window will be opened and you will be able to see the stream



Developer's manual



VLC Client stream



FFmpeg Client stream



Developer's manual

```
Select C:\WINDOWS\system32\cmd.exe

C:\Users\VEKATECH>ffplay rtsp://192.168.2.163:8554/mjpeg/1
ffplay version 4.0.2 Copyright (c) 2003-2018 the FFmpeg developers
  built with gcc 7.3.1 (GCC) 20180722
  configuration: --enable-gpl --enable-version3 --enable-sdl2 --enable-bzlib --enable-fontconfig --enable-gnutls --enable-
e-iconv --enable-libass --enable-libbluray --enable-libfreetype --enable-libmp3lame --enable-libopencore-amrnb --enable-
libopencore-amrwb --enable-libopenjpeg --enable-libopus --enable-libshine --enable-libsnappp --enable-libsoxr --enable-
libtheora --enable-libtwolame --enable-libvpx --enable-libwavpack --enable-libwebp --enable-libx264 --enable-libx265 --en
able-libxlm2 --enable-libzimg --enable-lzma --enable-zlib --enable-gmp --enable-libvidstab --enable-libvorbis --enable-
libvo-amrwbenc --enable-libmysofa --enable-libspeex --enable-libxvid --enable-libaom --enable-libbmfx --enable-amf --enab
e-ffnvcodec --enable-cuvid --enable-d3d11va --enable-nvenc --enable-nvdec --enable-dxva2 --enable-avisynth
  libavutil      56. 14.100 / 56. 14.100
  libavcodec     58. 18.100 / 58. 18.100
  libavformat    58. 12.100 / 58. 12.100
  libavdevice    58.  3.100 / 58.  3.100
  libavfilter    7. 16.100 / 7. 16.100
  libswscale     5.  1.100 / 5.  1.100
  libswresample  3.  1.100 / 3.  1.100
  libpostproc   55.  1.100 / 55.  1.100
[udp @ 000001d7397593c0] 'circular_buffer_size' option was set but it is not supported on this build (pthread support is
required)
[udp @ 000001d739749600] 'circular_buffer_size' option was set but it is not supported on this build (pthread support is
required)
Input #0, rtsp, from 'rtsp://192.168.2.163:8554/mjpeg/1':0B f=0/0
Metadata:
  title           :
Duration: N/A, start: 0.000000, bitrate: N/A
  Stream #0:0: Video: mjpeg, yuvj422p(pc, bt470bg/unknown/unknown), 640x480 [SAR 1:1 DAR 4:3], 30 tbr, 90k tbn, 90k tbc
[swscaler @ 000001d73e89d5c0] deprecated pixel format used, make sure you did set range correctly
  3.34 M-V:  0.034 fd=  1 aq=   0KB vq= 364KB sq=   0B f=2/2
```

FFmpeg version

```
COM4 - PuTTY
***** PROGRAM START *****
Network Setting up...

Connecting...
MAC Address is 00:02:f7:f0:00:00
IP Address is 192.168.2.163
NetMask is 255.255.255.0
Gateway Address is 192.168.2.254
Network Setup OK
running RTSP server
RTSP client started connection
Creating TSP streamer
Created streamer width=640, height=480
Creating RTSP session
RTSP received OPTIONS
RTSP received DESCRIBE
RTSP received SETUP
creating UDP socket at port 6970
creating UDP socket at port 6971
RTSP received PLAY
RTSP received OPTIONS
RTSP received OPTIONS
client closed socket, exiting
RTSP client closed connection
closing TCP socket
closing UDP socket
closing UDP socket
```

Mbed Console log (uRTSP server)



Developer's manual

Revision overview list

Revision number	Description changes
1.0	Initial

Vekatech Ltd.

63, Nestor Abadzhiev st.
4023 Plovdiv
Bulgaria
Tel.: +359 (0) 32 262362
info@vekatech.com

www.vekatech.com

VK-RZA1R3 Mbed App Demo rev. 1.0
Jun 10, 2022