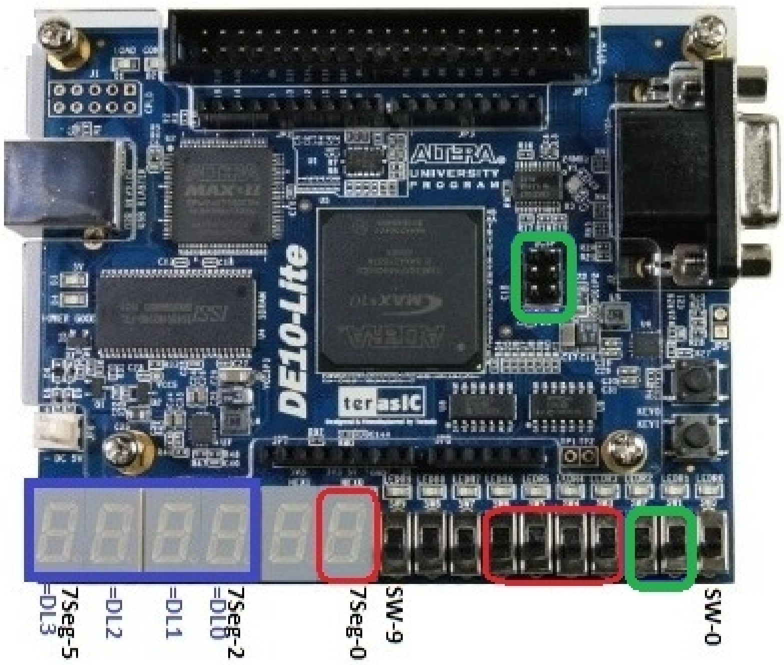
DEC RL01/RL02 DISK-DRIVE EMULATOR
User Manuel for the DE10-Lite board

DE10-Lite board with emulator interface
Embedded environment based on the Altera MAX 10 FPGA.

Emulates up to 4 RL01/RL02 drives simultaneously
Supports mixed environment of emulated + real RL drives
Access to 17 x 4 RL01/RL02 configurations sets
Ready for WLAN (based on ESP8266)

Configuration & Jumper-Settings

1. DE10-Lite board http://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=english&No=1021



SW-0: No functional meaning , only for LED expansion/test purpose.

SW-1 - SW-2 : WLAN operating mode, details in chapter WLAN mode.

SW-3 – SW-6 : SELECT mode. With these 4 switches, 16 sets (0 - F) with 4 units each can be selected. Current used set is displayed with 7seg-0. Details in chapter SELECT mode.

SW-7: **ONLINE / OFFLINE :** ON = ONLINE. Depending on the switch position, a different LED pattern is displayed.

SW-8: **DEBUG** mode : ON = DEBUG mode on

SW-9: **Format/Initialize** the micro SD-Card at the next restart. (= **Switch-1**)

7Seg-0: Shows the currently used disk set (0 – F) in SELECT mode.

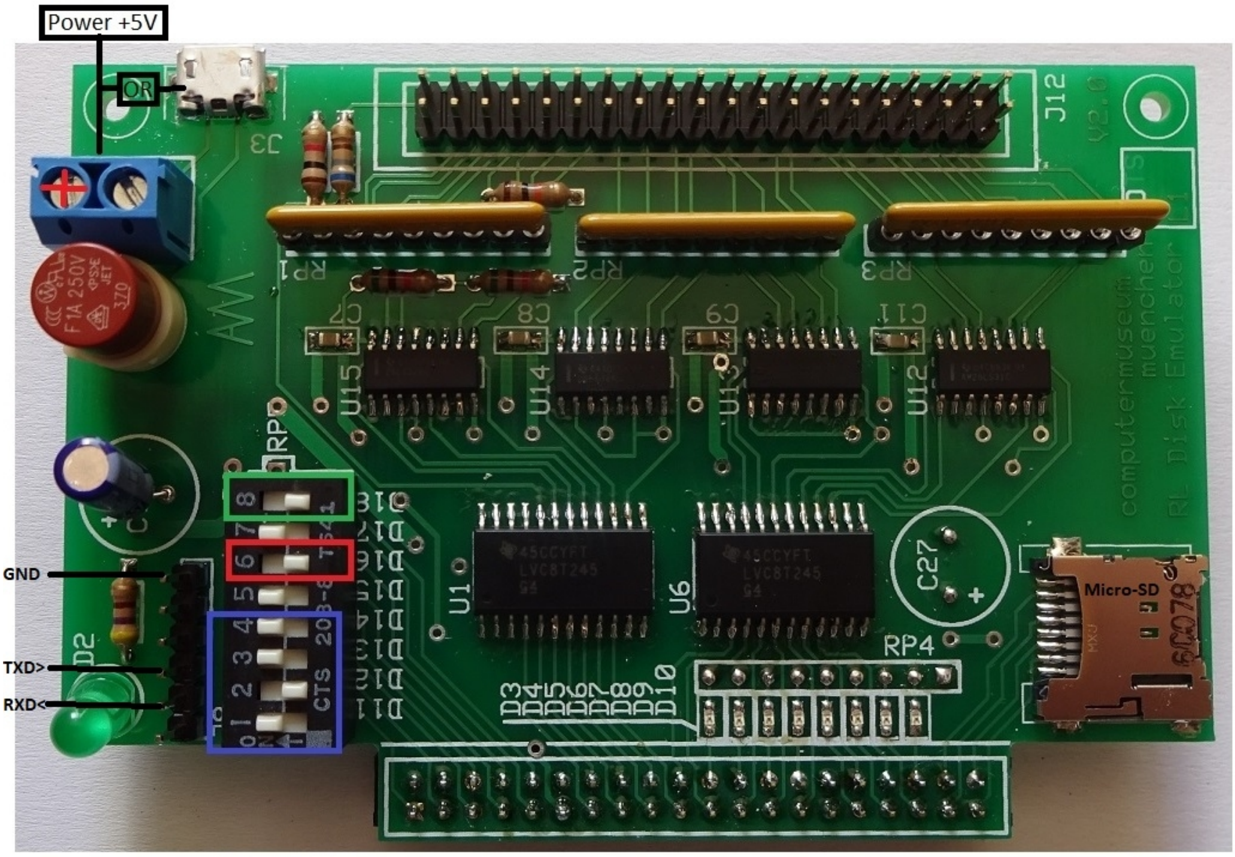
7Seg-1: Displays the RL unit-nr, which is currently active, also the external disks if available.

7Seg-2 -5: Displays the configured/emulated disk drives DL0(7seg-2) to DL3.
A configured disk is displayed with a circular cursor in the 7-segment display, otherwise the "-" character appears.

Button and Switches

Button 0 Reset / Restart
Button 1 Reconfigure. More details in section example.

2. Emulator interface, see also: http://fafner.dyndns.org/~heuberger/rlfertig/Readme.pdf



Implementation/architecture of the Interface board:

The interface board consists of the following components:

2 LVC8T245 = level converter
2 AM26LS31 = Transmitter
2 AM26LS32 = Receiver
8 LEDs
1 8 pin DIP-switch
3 pluggable resistor networks
1 holder for a micro SD card
2 connectors (40 pin)
1 6 pin connector for serial connection with 19200 baud based on + 3.3 Volt.
A “RoHS TTL-232R-3V3” USB converter will provide PC-connection.
2 5 Volt Power distributen connectors micro-USB connector: This is a simple and inexpensive way for a battery backup implementation with a standard handy-power bank.

Interface LED's (from right to left):

LED 0	heartbeat (blinking)
LED 1	Power OK
LED 2	Write in progress
LED 3	Read/Seek in progress
LED 4	Configured Unit dl3 active
LED 5	Configured Unit dl2 active
LED 6	Configured Unit dl1 active
LED 7	Configured Unit dl0 active

Interface switches (from top to bottom):

SWITCH 8	WLAN mode enable
SWITCH 7	Force Power OK
SWITCH 6	Select mode enable
SWITCH 5	RL drive type, RL01 or RL02
SWITCH 4	configure emulated RL drive dl0
SWITCH 3	configure emulated RL drive dl1
SWITCH 2	configure emulated RL drive dl2
SWITCH 1	configure emulated RL drive dl3

Pluggable resistor networks:

Necessary if the interface board is connected directly to the RL controller.

Serial Interface:

The serial interface is configured for **19200** baud based on a 6 pin connector with + 3.3 Volt.
A "RoHS TTL-232R-3V3" USB converter will provide PC-connection.

Battery Backup:

The additional micro-USB connector is available for connecting a standard Handy Power Bank.
This is a very simple and cost-effective Battery Backup implementatio.

Micro-SD:

The emulator software supports FAT32 for the Micro-SD card.
A **RL02** emulator-image file has a size of **11.521 KB** with file extension "**.DEC**"
A **RL01** emulator-image file has a size of **5.761 KB** with file extension "**.DEC**"

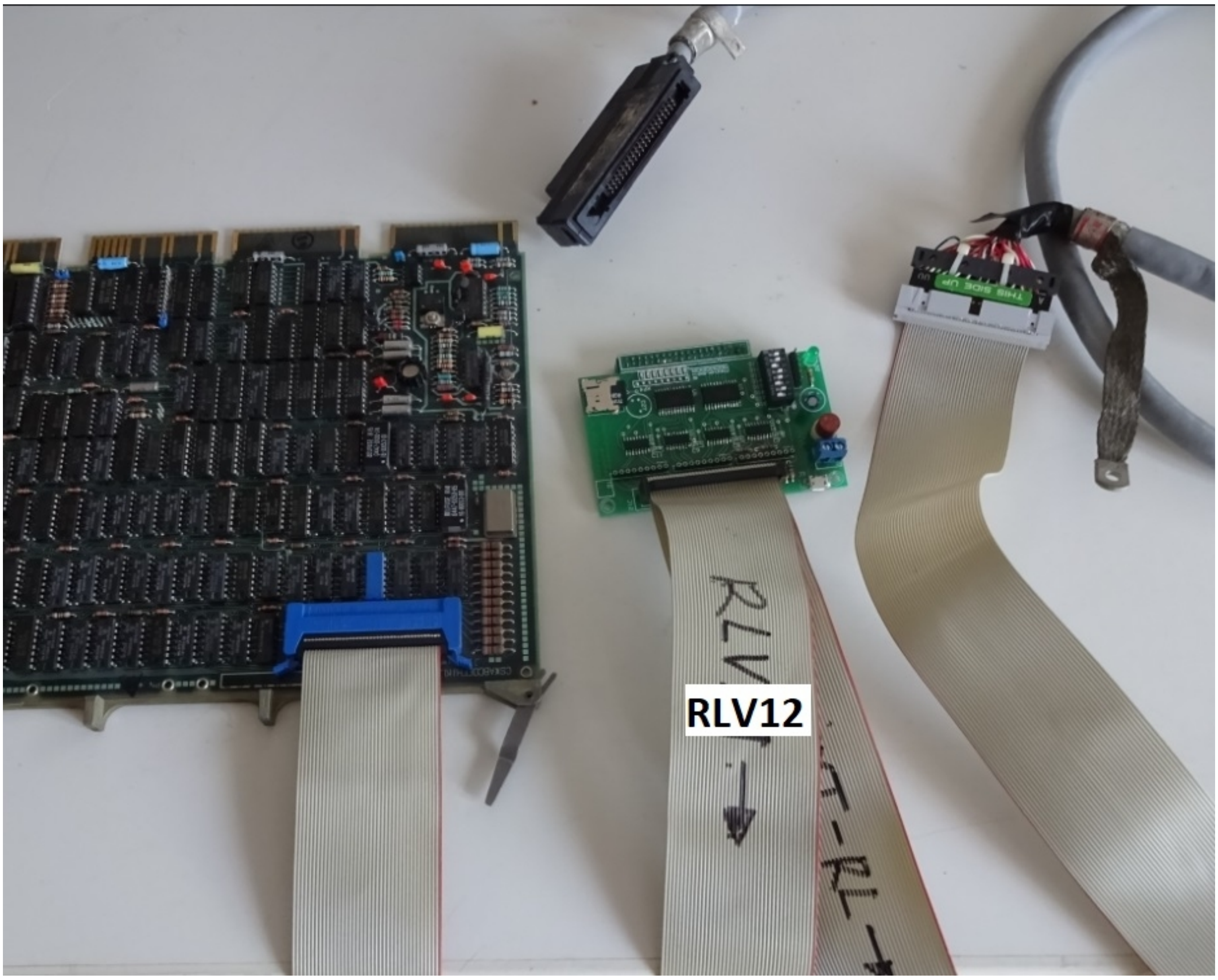
With the additional programs you can convert files: .DSK <==> .DEC
These programs are available at: www.2jo.de/pdp11/rlutils/rlutils.zip

3. Environment and Startup

Overview of the hardware and software setup including step-by-step procedures from installing the necessary software tools to use the DE10-Lite board.

This example shows a Q-BUS implementation with RLV12 controller

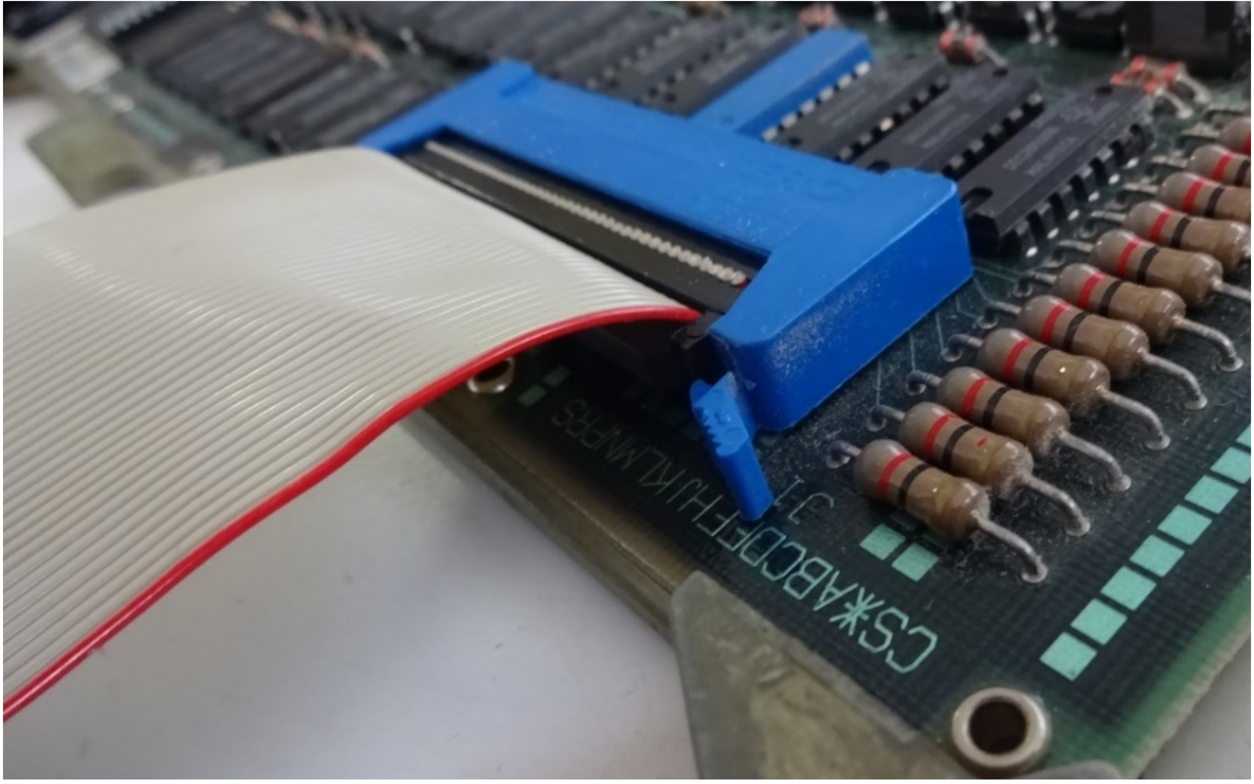
The following figure shows the connections based on a RLV12 Q-BUS controller-board to the emulator board and to an external RL disk drive.



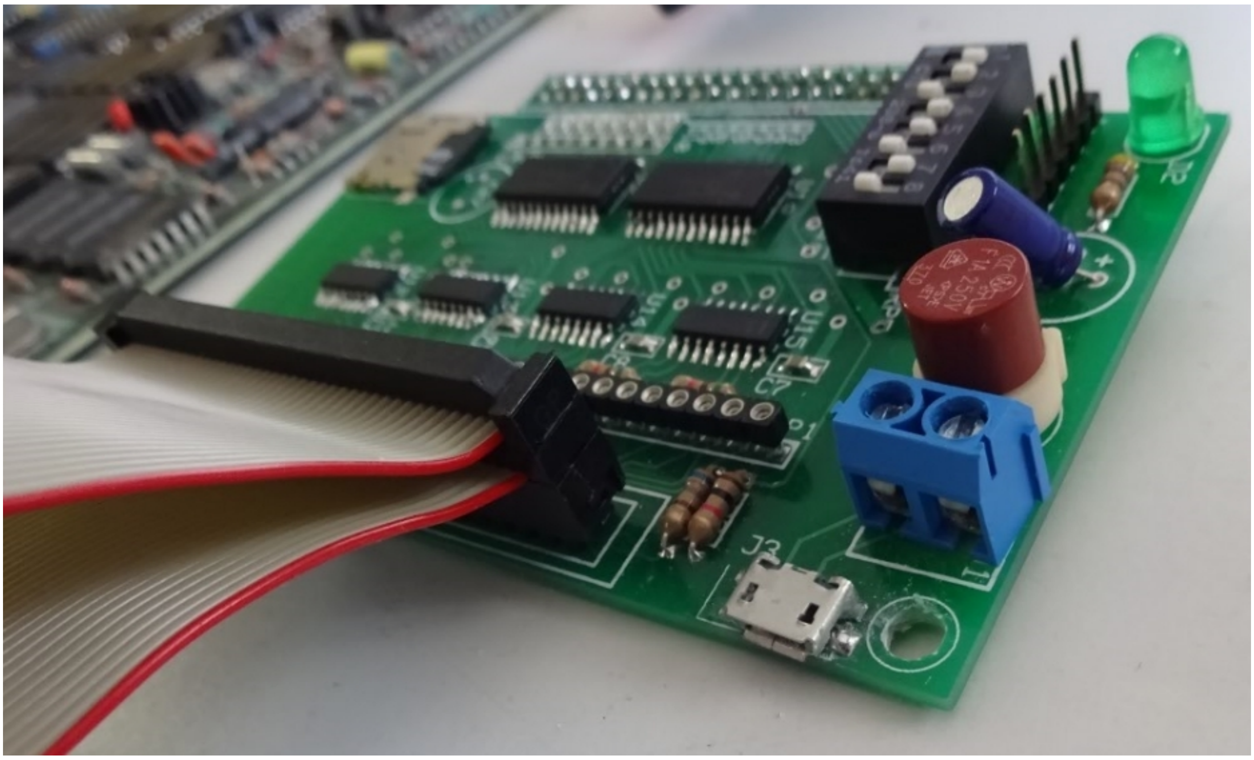
A 40-pin flat ribbon cable is required for the connections

The details of the connections are shown in the next 3 pictures

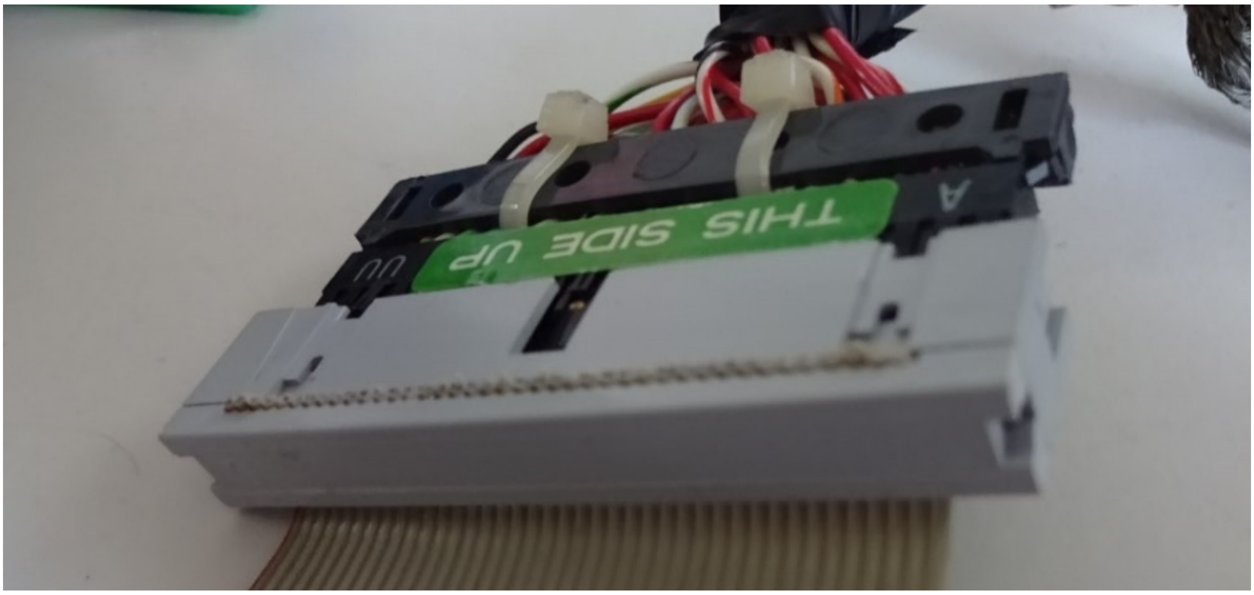
RLV12 (RLV11, RL11, RL8) **connection:**



Emulator board connection:

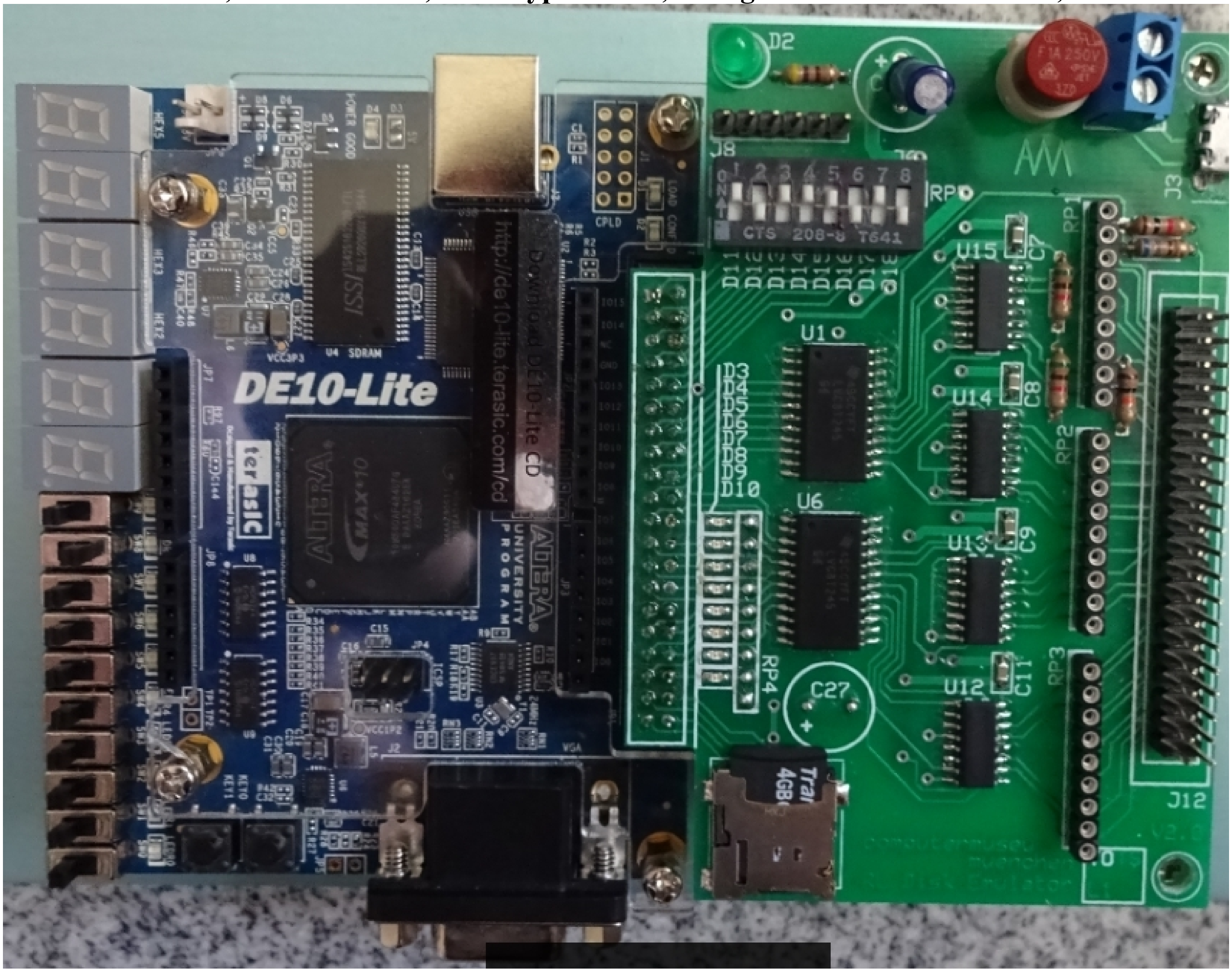


RL-BUS connection:



Disconnect if no external RL-drive is used , but install the 3 terminator resistor-networks on the emulator board.

Jumper settings for start-up/test example, NO external connection is required. OFFLINE mode, DEBUG mode, drive-type=RL02, configured : DL0: DL2: DL3:, Force POK



Jumper Settings details in the User Guide

Steps to bring up the emulator board up and running

Installing Quartus II Software Version 16.1

The Altera Complete Design Suite provides the necessary tools used for developing hardware and software solutions for Altera FPGAs. The Quartus II software is the primary FPGA development tool used to create reference designs along with the Nios II soft-core embedded processor integrated development environment.
User can download the latest software from https://www.altera.com/download/dnl-index.jsp
Use the Software Selector to get Quartus Version Prime 16.1 lite Edition downloaded and select MAX 10 FPGA support.

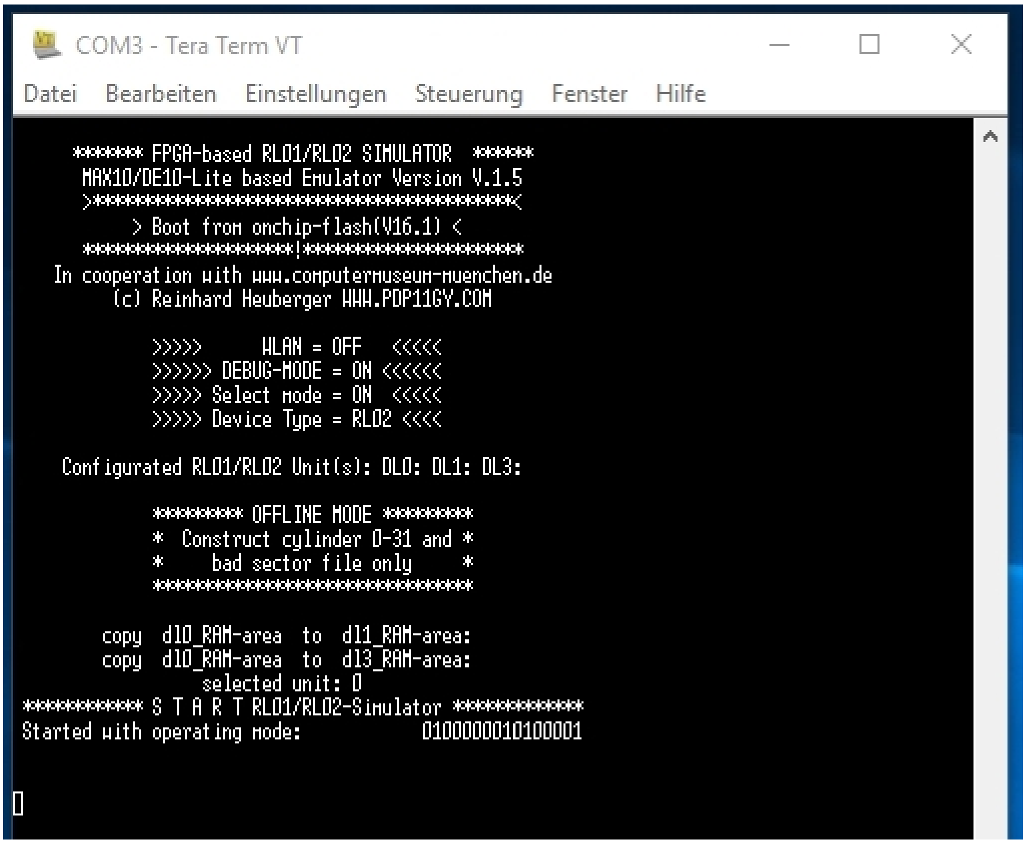
Now, Quartus can be started. Please follow the intructions in the flash folder if necessary.
If you want to make the design permanent, booting from the onchip_flash:

- **start the Quartus (Prime 16.1 lite Edition)**
- **Navigate to Tools/Programmer**
- **start Auto Detect and select 10M50DA**
- **right mouse click to "10M50DA" and select Change file**
- **select the .pof-file, in this case: flash/in_onchip_flash/MAX10_RL_emulator.pof**
- **Select the file @ Program/Configure and Press Start**

...... flashing ....

- **re-power the MAX10 DE10-Lite board.**

The system is ready for use and should start as follows:



```
    ******* FPGA-based RL01/RL02 SIMULATOR  ******
     MAX10/DE10-Lite based Emulator Version V.1.5
     >******************************************<
          > Boot from onchip-flash(V16.1) <
     *********************|*********************
  In cooperation with www.computermuseum-muenchen.de
        (c) Reinhard Heuberger WWW.PDP11GY.COM

            >>>>>      WLAN = OFF   <<<<<
            >>>>>> DEBUG-MODE = ON <<<<<<
            >>>>> Select mode = ON  <<<<<
            >>>>> Device Type = RL02 <<<<

   Configurated RL01/RL02 Unit(s): DL0: DL1: DL3:

            ********* OFFLINE MODE *********
            *  Construct cylinder 0-31 and *
            *     bad sector file only     *
            ********************************

       copy  dl0_RAM-area  to  dl1_RAM-area:
       copy  dl0_RAM-area  to  dl3_RAM-area:
                 selected unit: 0
************ S T A R T RL01/RL02-Simulator ************
Started with operating mode:          0100000010100001
```

Power up :

The heartbeat LED is blinking.
The 7-Segment displays HEX2 to HEX5 shows a circular cursor for the configured RL drives.

It takes about 10 seconds to start the NIOS processor and SD-RAM. When that's done,
the 8 LED's show a quick back and forth run which means that the NIOS II CPU has been started

As shown in the last picture, the system is now configured
Depending on Online or **Offline** mode, a different LED pattern is started

Offline Mode:

In this operating mode, no complete RL drives are emulated, access to the SD card is not possible and the emulator can be started without external connections, primary for verify purpose. **BUT,** if you connect the RL-Bus to the emulator board :
Access to an external "real" RL drives is possible (for test/verify purpose the external cable)
Limited access to cylinder 0-31 only is also possible. (about 0.3 MB)

Assuming RT-11 runs from another drive, such as RX01, RX02 or RX50, alternatively, my bootable RT-11 image files are available from my homepage, then the following commands can be used without problems (in this hardware example) :

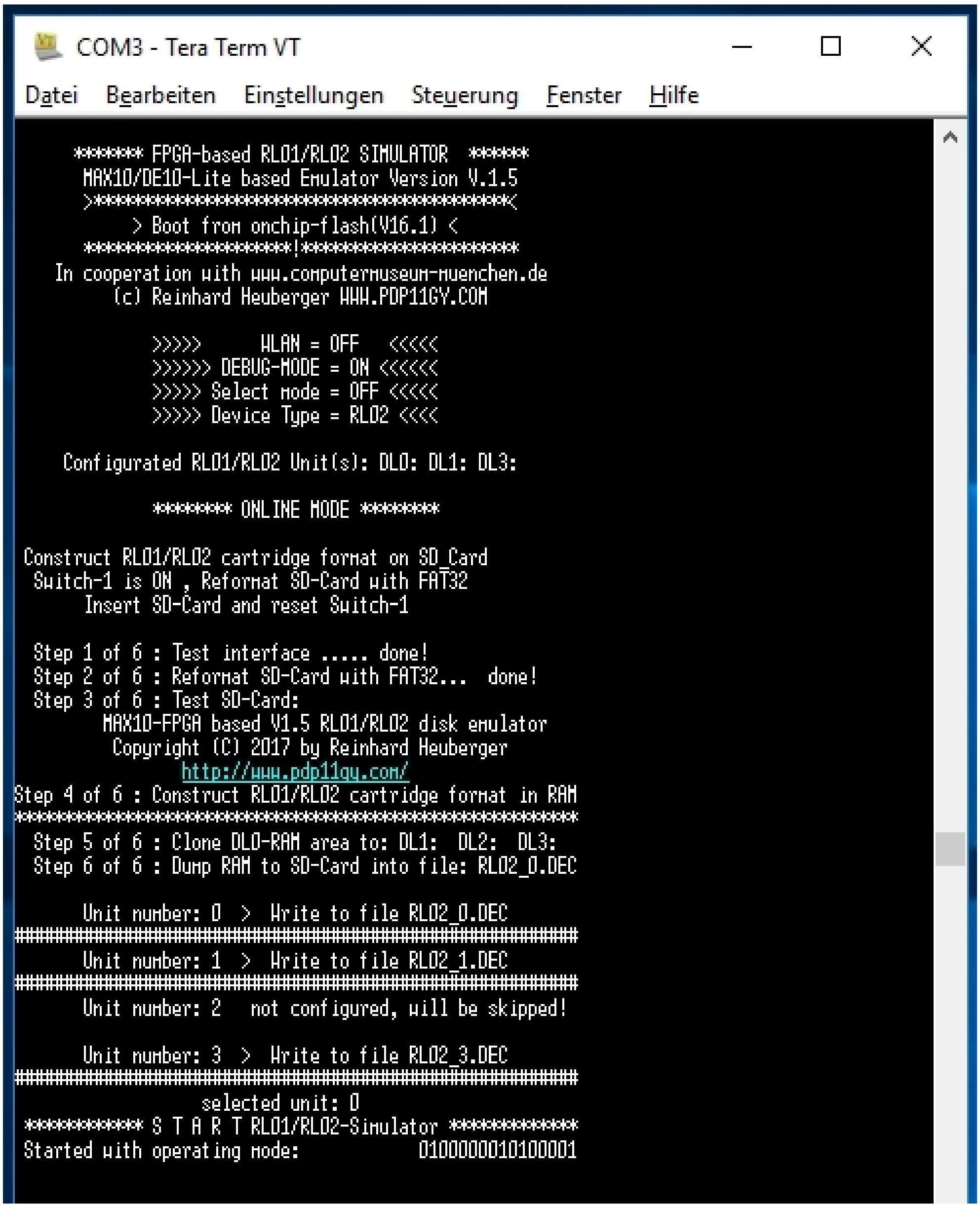
```
dump/term dl0:      ( or dl2: , dl3: )
init dl0:           ( or dl2: , dl3: )
copy/sys *.* dl0:   ( or dl2: , dl3: )    ( cancellation after 0.3 MB )
dir dl1:            ( external , "real" RL02 )
```

Online Mode:

SWITCH 7 is responsible to select Online mode.

4. Example

Assuming, we have a real RL02 disk drive, unit 2 and we want to copy the data from the real RL02 to the emulated RL02 disk drives. First, we have to remove the terminator from the emulator board and cabling the real RL02 to be at the end of the RL-bus with connected RL-bus terminator. The real RL02 disk drive is configured as unit dl2 and the emulator board is configured for RL02 units dl0, dl1 and dl3 : SWITCH 4, 5, 7 = ON , SWITCH 6 = OFF.
(See also picture on page 7)
The following message appears on the screen :

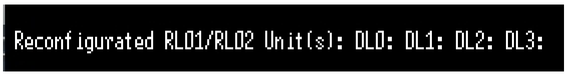


```
      ******* FPGA-based RL01/RL02 SIMULATOR  ******
       MAX10/DE10-Lite based Emulator Version V.1.5
       >*******************************************<
            > Boot from onchip-flash(V16.1) <
       *********************|**********************
    In cooperation with www.computermuseum-muenchen.de
          (c) Reinhard Heuberger WWW.PDP11GY.COM

              >>>>>      WLAN = OFF   <<<<<
              >>>>>> DEBUG-MODE = ON <<<<<<
              >>>>> Select mode = OFF <<<<<
              >>>>> Device Type = RL02 <<<<

     Configurated RL01/RL02 Unit(s): DL0: DL1: DL3:

              ******** ONLINE MODE ********

 Construct RL01/RL02 cartridge format on SD_Card
  Switch-1 is ON , Reformat SD-Card with FAT32
       Insert SD-Card and reset Switch-1

  Step 1 of 6 : Test interface ..... done!
  Step 2 of 6 : Reformat SD-Card with FAT32...  done!
  Step 3 of 6 : Test SD-Card:
         MAX10-FPGA based V1.5 RL01/RL02 disk emulator
          Copyright (C) 2017 by Reinhard Heuberger
                 http://www.pdp11gy.com/
Step 4 of 6 : Construct RL01/RL02 cartridge format in RAM
*********************************************************
  Step 5 of 6 : Clone DL0-RAM area to: DL1:  DL2:  DL3:
  Step 6 of 6 : Dump RAM to SD-Card into file: RL02_0.DEC

       Unit number: 0  >  Write to file RL02_0.DEC
#########################################################
       Unit number: 1  >  Write to file RL02_1.DEC
#########################################################
       Unit number: 2   not configured, will be skipped!

       Unit number: 3  >  Write to file RL02_3.DEC
#########################################################
                   selected unit: 0
 ************ S T A R T RL01/RL02-Simulator *************
 Started with operating mode:            0100000010100001
```

Now, we can copy the data from the real RL02 disk drive unit 2 to the emulated RL02 disk drives, for example (RT-11): copy/device dl2: dl0: (dl1: / dl3:)

Now comes a special feature:

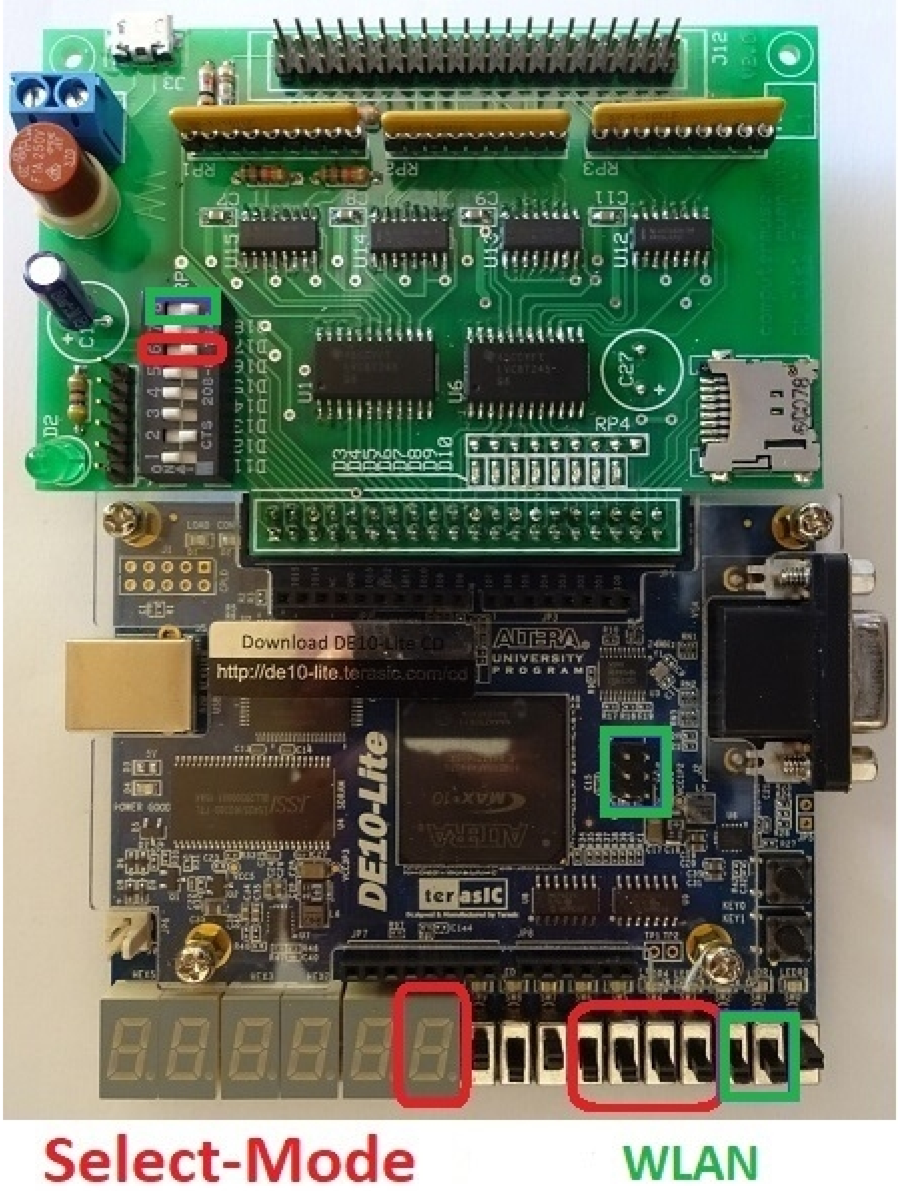
- Switch down the real RL02 disk drive
- Set SWITCH 2 = ON (DL2)
- Press button 2 on DE10-Lite board and following message will appear:



```
Reconfigurated RL01/RL02 Unit(s): DL0: DL1: DL2: DL3:
```

From now on, 4 RL02 units will be emulated with full access to the dl2 unit.

5. SELECT and WLAN mode

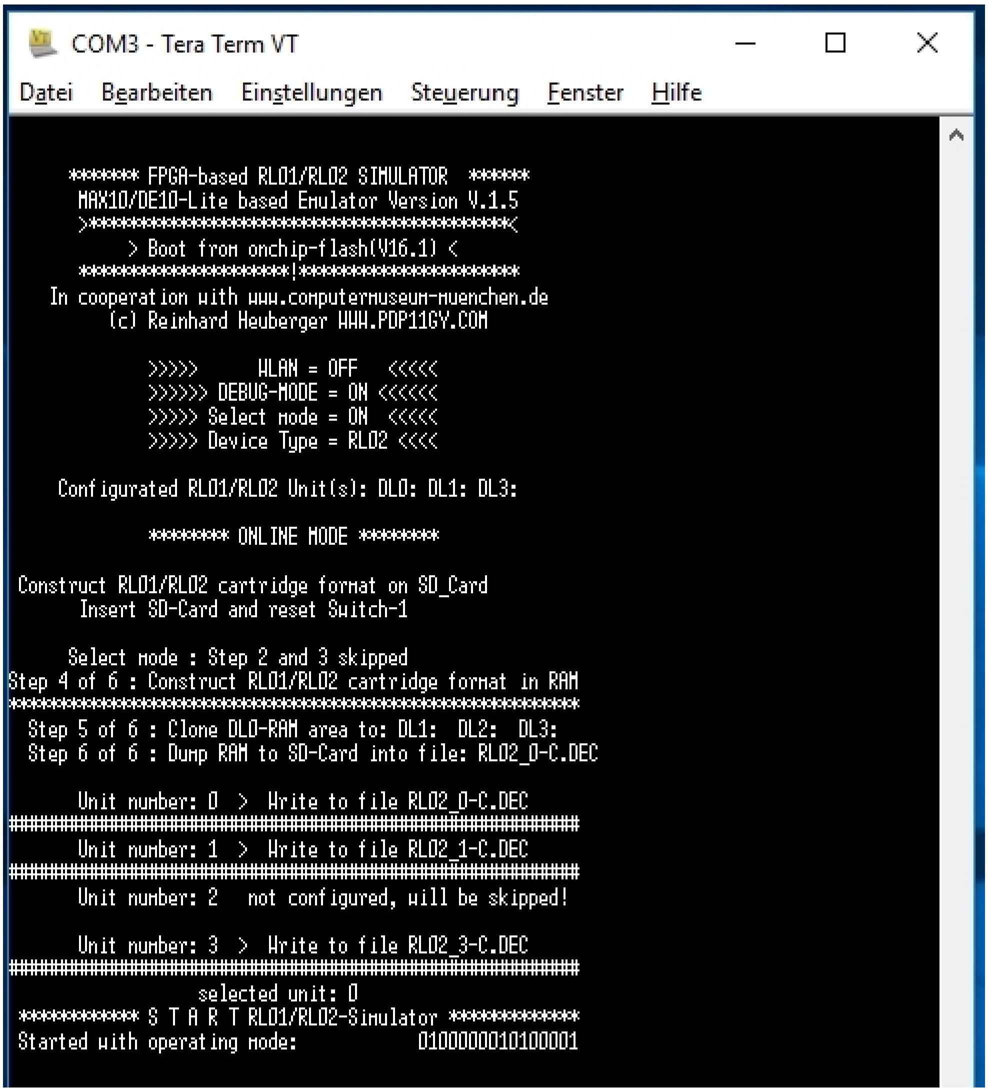


5.a SELECT mode , implemented with version V1.4 or higher

Without the SELECT mode, only one set consisting of 4 RL-images was supported. With the implementation of the Select mode, 16 additional sets, each consisting of a maximum of 4 RL-images are supported. This results in a total of 17 sets and means that a maximum of 68 RL-images are available and accessable in sets of 4 RL-images. **The SELECT mode is activated with SWITCH 6 on the interface board.**
Assuming that SW-3 = off, SW-4 = off, SW-5 = on and SW-6 = on, the hex-code C appears in the display 7seg-0. Now the RL-images **RL02_0-C**.DEC to **RL02_0-C**.DEC can be created.

With reference to the example on page 9 an initialization with SELECT mode is made without formatting the SD card: Step 2 and 3 are skipped.

Here is the SELECT mode example:



```
      ******* FPGA-based RL01/RL02 SIMULATOR  ******
       MAX10/DE10-Lite based Emulator Version V.1.5
       >*******************************************<
            > Boot from onchip-flash(V16.1) <
       *********************|**********************
    In cooperation with www.computermuseum-muenchen.de
          (c) Reinhard Heuberger WWW.PDP11GY.COM

              >>>>>      WLAN = OFF   <<<<<
              >>>>>>> DEBUG-MODE = ON <<<<<<
              >>>>> Select mode = ON  <<<<<
              >>>>> Device Type = RL02 <<<<

     Configurated RL01/RL02 Unit(s): DL0: DL1: DL3:

              ******** ONLINE MODE ********

 Construct RL01/RL02 cartridge format on SD_Card
       Insert SD-Card and reset Switch-1

      Select mode : Step 2 and 3 skipped
Step 4 of 6 : Construct RL01/RL02 cartridge format  in RAM
**********************************************************
  Step 5 of 6 : Clone DL0-RAM area to: DL1:  DL2:  DL3:
  Step 6 of 6 : Dump RAM to SD-Card into file: RL02_0-C.DEC

       Unit number: 0  >  Write to file RL02_0-C.DEC
##########################################################
       Unit number: 1  >  Write to file RL02_1-C.DEC
##########################################################
       Unit number: 2   not configured, will be skipped!

       Unit number: 3  >  Write to file RL02_3-C.DEC
##########################################################
                   selected unit: 0
 ************ S T A R T RL01/RL02-Simulator *************
 Started with operating mode:           0100000010100001
```

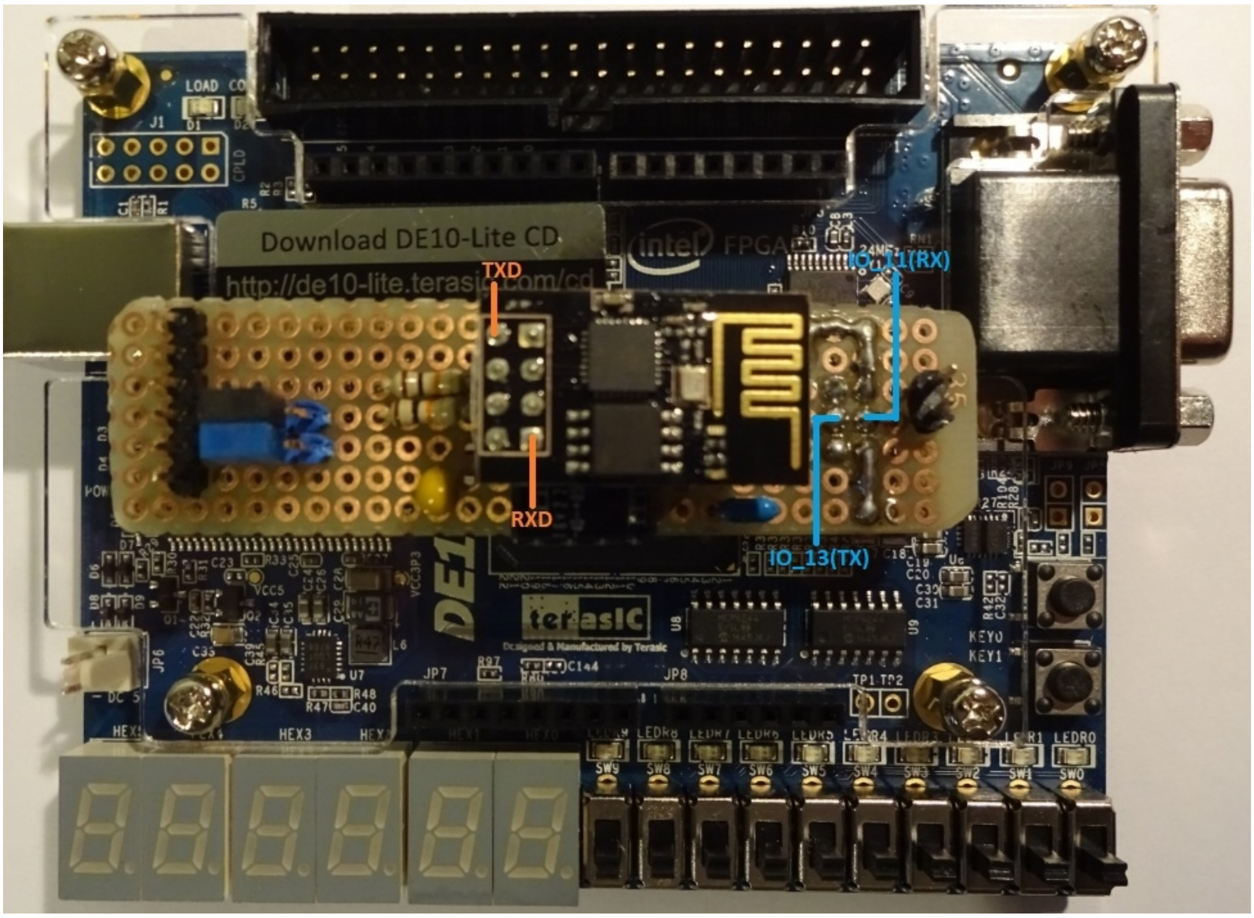
Note: If the SELECT mode is **not** switched on (SWITCH 6 off) the SD card will be formatted and the images RL02_0.DEC to RL02_3.DEC will be generated.

Summary: With this implementation you can create different disk images Sets of operating systems and/or data, selectable via different Set letters, 0 to F. For example, Set 0 is for RT11, set 1 for RSX etc….
To use a different disk Set, a restart is always required.

5.b WLAN mode, implemented with version V1.5

In general, the question arises whether it is useful for a disk emulator or not.
Does it make sense?
...to get the debug messages on your Handy / LapTop?
….to upload / download RL images to / from your Handy / LapTop ?

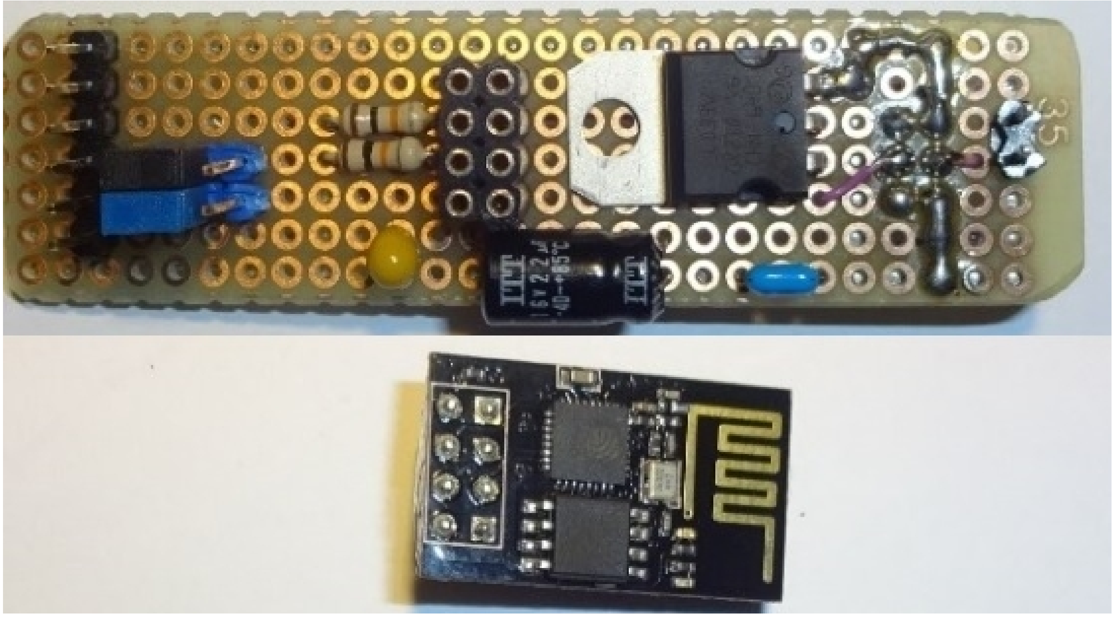
Well, I've tried it and in principle it works. I have not implemented everything yet and wait for a feedback from you. Then I will consider how I continue. Perhaps there is a collaboration.
Also, at the moment there is no PCB board available, only one prototype as shown in the following pictures.



The WLAN implementation is based on the **ESP8266.**

It would not be much effort to design a PCB board which holds the ESP8266 as in the picture above. In addition, only one DC converter +5V to +3.3V, one tantalum capacitor and two resistors are required. With reference to the DE10-Lite_User_Manuel, page **32** :
I had decided that I use the Arduino Connecter as interface to the ESP8266 as follows:

Arduino_IO_13 (TX) ---------------------> ESP8266 (RXD)
Arduino_IO_11 (RX) <-------------------- ESP8266 (TXD)
The serial connection runs at 115200 Baud, 8N1
Prototyp

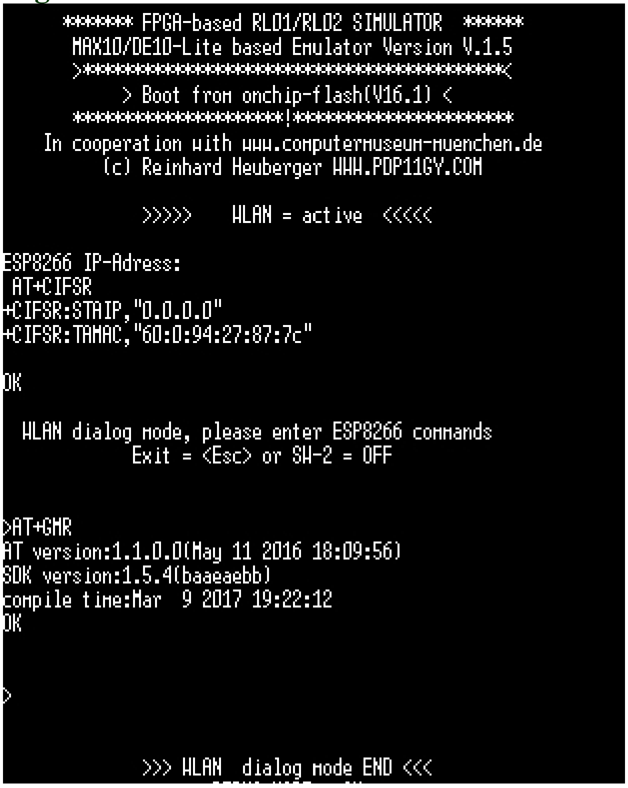


On the left is a 6-pin connector for direct serial connection and two jumpers. This may be necessary for debugging purposes or for an ESP8266 FW upgrade. On the right are 2 Pins for a HW reset. The two 10k resistors should be connected to + 3.3V and to the pins RST and CH_PD.
Many further details and information, as well as programming, can be found on the Internet

Software:

Currently 2 WLAN modes are implemented. If WLAN is activated, the two switches SW-1 and SW2 will be checked after startup.
SW-1=OFF , SW-2=ON : WLAN dialog mode
SW-1=ON , SW-2=OFF : WLAN config mode via file ESP8266.CFG
The remaining 2 switch combinations are not implemented yet.

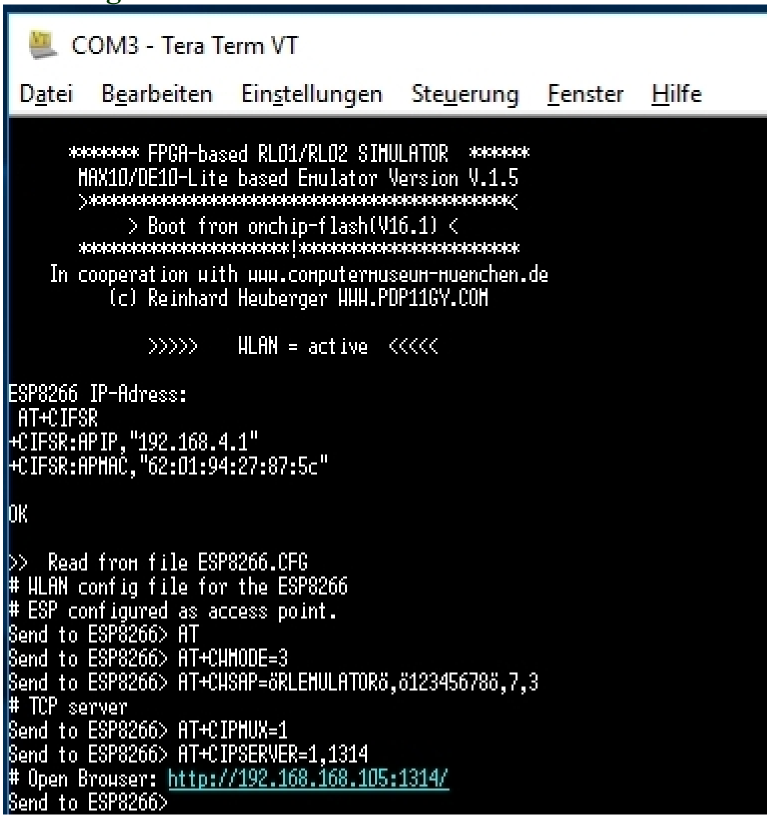
Example 1 , WLAN dialog mode:



```
      ******* FPGA-based RL01/RL02 SIMULATOR  ******
       MAX10/DE10-Lite based Emulator Version V.1.5
       >****************************************<
            > Boot from onchip-flash(V16.1) <
       *********************|*********************
    In cooperation with www.computermuseum-muenchen.de
          (c) Reinhard Heuberger WWW.PDP11GY.COM

              >>>>>    WLAN = active  <<<<<

ESP8266 IP-Adress:
 AT+CIFSR
+CIFSR:STAIP,"0.0.0.0"
+CIFSR:TAMAC,"60:0:94:27:87:7c"

OK

  WLAN dialog mode, please enter ESP8266 commands
             Exit = <Esc> or SW-2 = OFF


>AT+GMR
AT version:1.1.0.0(May 11 2016 18:09:56)
SDK version:1.5.4(baaeaebb)
compile time:Mar  9 2017 19:22:12
OK


>


              >>> WLAN  dialog mode END <<<
```

In the WLAN dialog mode, it is possible to enter an AT-command and send it directly to the ESP8266. In this example 1, the AT+GMR command was used to receive the current FW version.

Example 2 , WLAN config mode via file ESP8266.CFG



```
      ******* FPGA-based RL01/RL02 SIMULATOR  ******
       MAX10/DE10-Lite based Emulator Version V.1.5
       >****************************************<
            > Boot from onchip-flash(V16.1) <
       *********************|*********************
    In cooperation with www.computermuseum-muenchen.de
          (c) Reinhard Heuberger WWW.PDP11GY.COM

              >>>>>    WLAN = active  <<<<<

ESP8266 IP-Adress:
 AT+CIFSR
+CIFSR:APIP,"192.168.4.1"
+CIFSR:APMAC,"62:01:94:27:87:5c"

OK

>>  Read from file ESP8266.CFG
# WLAN config file for the ESP8266
# ESP configured as access point.
Send to ESP8266> AT
Send to ESP8266> AT+CWMODE=3
Send to ESP8266> AT+CWSAP=öRLEMULATORö,ö12345678ö,7,3
# TCP server
Send to ESP8266> AT+CIPMUX=1
Send to ESP8266> AT+CIPSERVER=1,1314
# Open Browser: http://192.168.168.105:1314/
Send to ESP8266>
```

In this mode, the file ESP8266.CFG will be read from the micro SD card and the commands will be executed. This file can be modified with every editor and looks as follows:

```
# WLAN config file for the ESP8266
# ESP configured as access point.
AT
AT+CWMODE=3
AT+CWSAP=”RLEMULATOR”,”12345678”,7,3
# TCP server
AT+CIPMUX=1
AT+CIPSERVER=1,1314
# Open Browser: http://192.168.168.105:1314/
```

In principle, the basic things are up and running . Whether and how I complete the WLAN implementation, I don’t know yet, because it makes a lot of work and also depends on the feedback. …. and again …. does it make sense?

For further questions, please contact me.
INFO@pdp11gy.com