DRS4 Evaluation Board User's Manual

Board Revision 3.1
as of February 2011

Last revised: February 15, 2011

Stefan Ritt
Paul Scherrer Institute
CH-5232 Villigen PSI
Switzerland

Email: stefan.ritt@psi.ch
Phone: +41 56 310 3728

Revision History

Date	Modification
2 March 09	Initial Revision
27 April 09	Mention input range, added timing calibration description
3 Aug. 09	Added LED description
12 July 10	Updated documentation for evaluation board V3
12 Sept. 10	Added maximum input voltage

Table of Contents

1. Introduction

The DRS4 chip, which has been designed at the Paul Scherrer Institute, Switzerland by Stefan Ritt and Roberto Dinapoli is a Switched Capacitor Array (SCA) capable of digitizing eight channels at sampling speeds up to 5 GSPS. This chip is available through the PSI technology transfer program for other institutes and organizations. In order to simplify the design process to integrate the DRS4 chip into custom electronics, an evaluation board has been designed, which demonstrates the basic operation of the chip. It has SMA connectors for four input channels CH1 to CH4, an USB 2.0 connector and a LEMO trigger input (Figure 1). The board is powered through the USB port and contains an on-board trigger logic. It comes with MS Windows® and Linux drivers and two application programs. It is basically equivalent to a four channel 5 GSPS digital oscilloscope.

This manual describes the software installation, the usage of the application programs, and gives hints for developers seeking to build new electronics around the DRS4 chip.

1.1. Board description

Since the DRS4 chip has differential inputs, the board uses four active buffers (THS4508 from Texas Instruments ®) to converted the 50-Ohm terminated single ended inputs into differential signals. Analog switches in front of the buffers (ADG901 form Analog Devices®) are used to de-couple the inputs during calibration. Two reference voltages are generated by the on-board 16-bit DAC to measure the offset and gain of all DRS4 storage cells for calibration. The four analog inputs are AC coupled and have a input range of 1 V peak-to-peak. The absolute maximum input voltage range is -0.5V to +2.8V. The DRS4 is read out with a 14-bit ADC (AD9245 from Analog Devices®) and a FPGA (Xilinx® Spartan 3). The USB connection is implemented with a micro controller (Cypress® CY2C68013A). The high speed modus of the USB 2.0 bus allows for data transfer rates of more than 20 MB/sec.

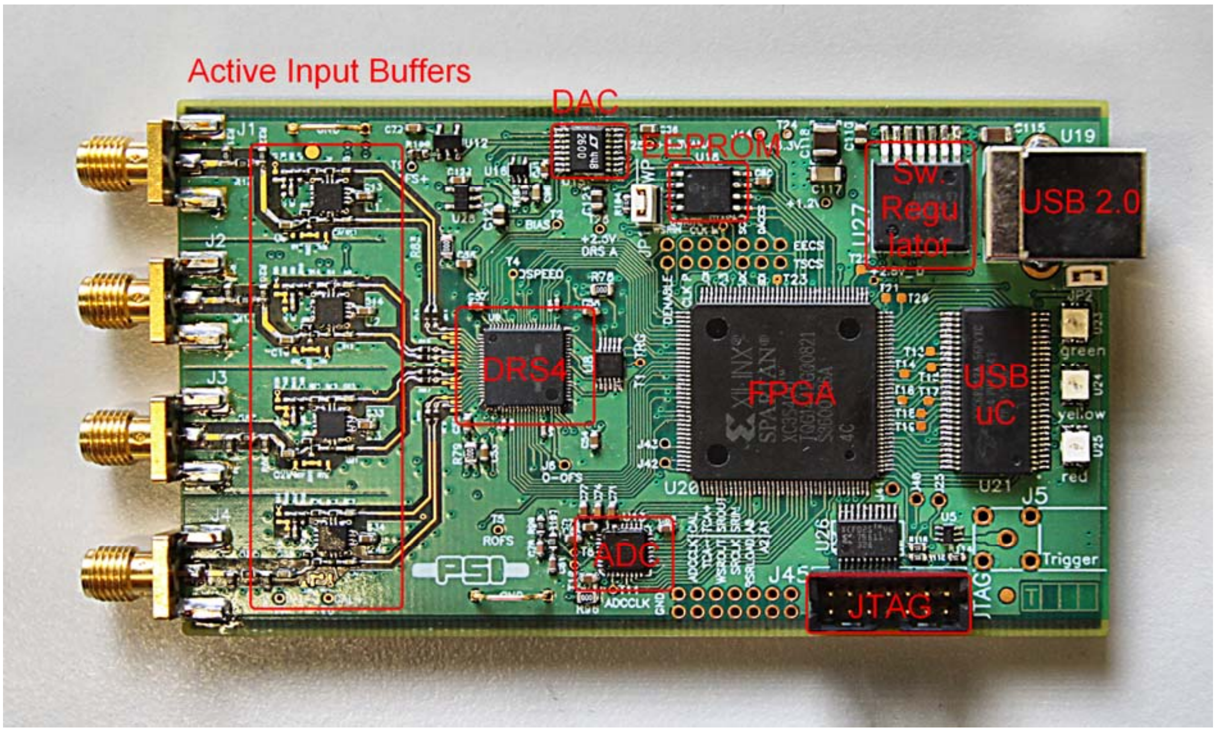


Figure 1: Picture of the DRS4 Evaluation Board V3 with different components

For trigger purposes, a 50 Ω terminated TTL compatible input is implemented (Lemo connector). Since the input is 50 Ω terminated, care has to be taken that the trigger source is able to drive at least 2.2 V into 50 Ω.

A on-board discriminator with programmable level allows for self triggering on any of the four input channels. An 1 MBit EEPROM (25LC1025 from Microchip®) is used to store the board serial number and calibration information. Two 14-pin headers carry all important logical signals which allow easy debugging with a logic analyzer or oscilloscope. A JTAG adapter can be used to update the FPGA firmware through a Xilinx® Platform Cable Adapter.

The specifications of the board inputs is summarized in following table:

Analog inputs		
Termination	50 Ω	AC coupled
Input range	1 V p-p	
Maximum allowed input voltage		
DC	± 10V	
Long pulse (<2µs)	± 20V	
Short pulse (<200ns)	± 30V	
Trigger input		
Termination	50 Ω	5 V TTL compatible
Maximum allowed input voltage	-0.5 V to +5.5 V	
High Level Input Voltage	2.2 V (max)	

1.2. LEDs

The DRS4 evaluation board is equipped with four LEDs. They are operated by the micro controller and the FPGA and have following meaning:

LED	Meaning
Green	This LED becomes green when the on-board micro-controller booted successfully. If this LED stays dark, there is either no power or the micro-controller lost it's program, which requires a re-programming of the EEPROM.
Yellow	When the on-board FPGA boots correctly this LED becomes lit. If it stays dark, it might be that the FPGA program was lost and requires re-programming. After booting, this LED indicates the board status. If lit, the DRS4 chip is active and sampling data. If stopped by software or a trigger, this LED turns off. A special pulse stretcher ensures that even in high trigger rate environments this LED does not flash with more than ~10Hz so the blinking can still be seen by eye.
Red	When lit, this LED indicates a error condition

1.3. Firmware Description

Both the Windows and the Linux distribution contain a subdirectory “firmware” which contains the FPGA and Microcontroller firmware for the DRS4 Evaluation Board. The FPGA firmware is written in pure VHDL, thus making it easy to port it to other FPGA devices such as Altera® or Lattice®. Only a few Xilinx® basic components such as clock managers and I/O blocks have been instantiated and must be adapted when another FPGA manufacturer than Xilinx® is chosen. The FPGA source code is contained in several files with following contents:

`src/drs4_eval1.vhd`	Top level entity. Routing of clock signals, global reset signal, LEDs and LEMO input
`src/drs4_eval1_app.vhd`	Main file containing state machines for DRS4 readout, serial interface to DAC, EEPROM and temperature sensor, trigger logic and reference clock generation
`src/usb_dpram.vhd`	Instantiates block ram for waveform storage
`src/usb_racc.vhd`	Interface to CY2C68013A microcontroller in slave FIFO mode. Implements a set of status and control registers through which the main application can be controlled
`src/usr_clocks.vhd`	Generates 66 MHz, 132 MHz, 264 MHz and a phase shifted 66 MHz clock out of the 33 MHz quartz input frequency via the Xilinx® Digital Clock Managers (DCM)
`ucf/drs4_eval1.ucf`	Constraint file. Assigns package pins and defines clock constraints
`3s400/drs4_eval1.ise`	Xilinx® ISE 9.2i project file
`3s400/drs4_eval1.bit`	Compiled firmware image directly for Spartan 3s400 FPGA
`3s400/drs4_eval1.mcs`	Compiled firmware image for FPGA EEPROM XCF02S
`3s400/drs4_eval1.ipf`	Xilinx® Impact project file to program FPGA via download cable

The firmware for the USB microcontroller from Cypress® is written in C and must be compiled with the Keil® 8051 C compiler. It contains the standard include and library files from the Cypress EZ-USB® development kit plus some DRS specific files:

`CY7C68013A/drs_eval.c`	Main micro controller firmware file
`CY7C68013A/dscr.a51`	USB descriptor tables
`CY7C68013A/drs_eval.hex`	Compiled firmware file (Intel HEX format)
`CY7C68013A/drs_eval1.iic`	Compiled firmware file (For Cypress EZ-USB Console download)
`CY7C68014A/*`	Remaining files are standard files from EZ-USB development kit

The FPGA firmware implements a set of control and status registers, through which the DRS4 can be controlled and read out. The mapping of the control registers is as follows:

#	Ofs.	Bit	Name	Comment
0	0x00	0	start_trig	Write a "1" to start the domino wave
0	0x00	1	reinit_trig	Write a "1" to stop & reset the DRS chip
0	0x00	2	soft_trig	Write a "1" to stop the DRS chip & read the data to RAM
0	0x00	3	eeprom_write_trig	Write contents of RAM into EEPROM (32kB page)
0	0x00	4	eeprom_read_trig	Read contents of EEPROM into RAM (32kB page)
0	0x02	18	led	1=on, 0=blinks once at beginning of DRS chip readout
0	0x02	19	tcal_en	Switch on (1) / off (0) 264 MHz calib. sig. for DRS chips
0	0x02	20	tcal_source	System clock (0) or separate quartz (1) clock source
0	0x02	21	transp_mode	1=send DRS inputs to outputs ("transparent mode")
0	0x02	22	enable_trigger1	Write a "1" to enable external trigger (LEMO)
0	0x02	23	readout_mode	0:start from first bin, 1:start from domino stop
0	0x02	24	neg_trigger	1=trigger on high to low transition
0	0x02	25	acalib	Write "1" to enable amplitude calibration
0	0x02	27	dactive	0:stop domino wave during readout, 1:keep it running
0	0x02	28	standby	1: put chip in standby mode
0	0x02	29	trigger_source1	Analog trigger source bits CH1-CH4
0	0x02	30	trigger_source2	Analog trigger source bits CH1-CH4
0	0x02	31	enable_trigger2	Write a "1" to enable analog trigger
1	0x04	31..16	DAC0	Set DAC 0 (=A, ROFS)
1	0x06	15..0	DAC1	Set DAC 1 (=B, CMOFS)
2	0x08	31..16	DAC2	Set DAC 2 (=C, CAL-)
2	0x0A	15..0	DAC3	Set DAC 3 (=D, CAL+)
3	0x0C	31..16	DAC4	Set DAC 4 (=E, BIAS)
3	0x0E	15..0	DAC5	Set DAC 5 (=F, TLEVEL)
4	0x10	31..16	DAC6	Set DAC 6 (=G, O-OFS)
4	0x12	15..0	DAC7	Set DAC 7 (=H, -)
5	0x14	31..24	configuration	Bit0: DMODE, Bit1: PLLEN, Bit2: WSRLOOP
5	0x14	23..16	channel_config	1=1x8k,0x11=2x4k,0x33=4x2k,0xFF=8x1k
5	0x16	7..4	first_chn	First channel address to read out (0..9)
5	0x16	3..0	last_chn	Last channel address to read out (1..9)
6	0x18	31..16	trigger_delay	Trigger delay in ticks of roughly 2.3 ns
6	0x1A	15..0	sampling_freq	Sampling frequency in ticks (=1024/f_{samp}*0.120-2)
7	0x1E	15..0	eeprom_page	Page number for EEPROM communication

While the mapping of the status registers is like this:

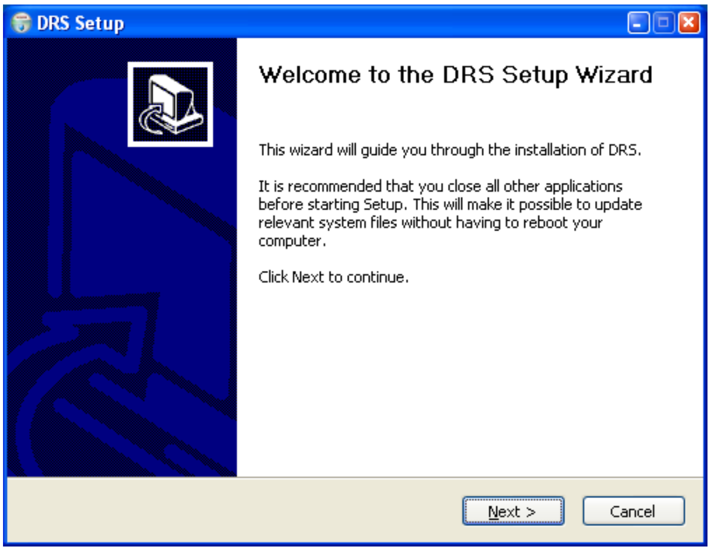
#	Ofs.	Bits	Name	Comment
0	0x00	31..16	board_magic	0xC0DE, Magic number for DRS board identification
0	0x02	15..8	board_type	5: Eval. Board V2; 7: Eval. Board V3
0	0x02	7..0	drs_type	4 for DRS4
1	0x04	0	running	"1" while domino wave running or readout in progress
2	0x08	31..16	stop_cell	position of cell where sampling stopped at last trigger
8	0x20	31..16	temperature	temperature in 0.0625 deg. C units
9	0x24	31..16	serial_cmc	Serial number CMC board
9	0x26	15..0	version_fw	firmware version (SVN revision)

All registers are implemented as 32-bit registers, so they can be mapped easily into some VME address space for example if one decides to build a VME board containing the DRS4.

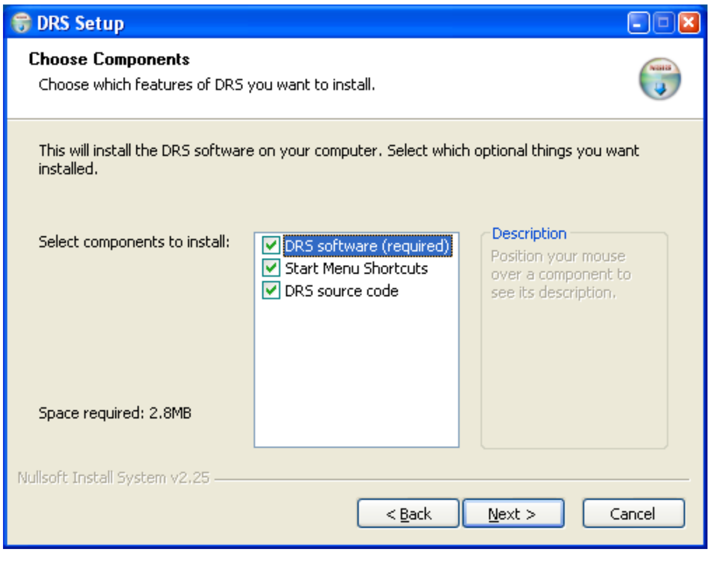
2. Installation

2.1. Windows XP

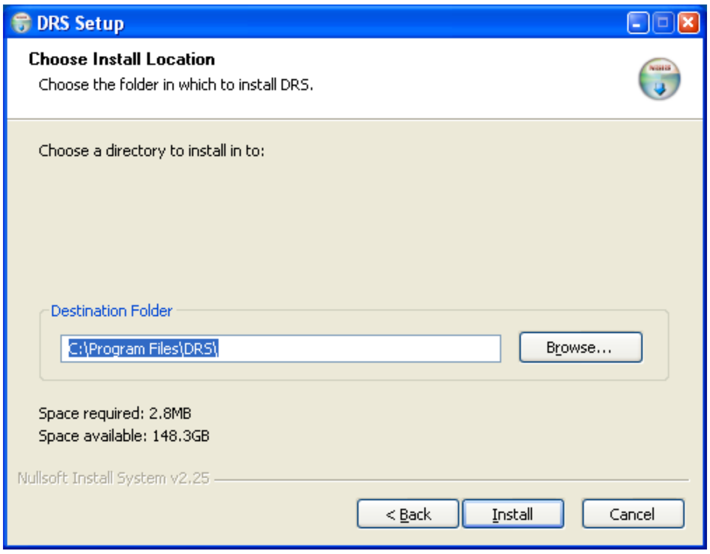
Under MS Windows® it is important to install the necessary driver before connection the DRS4 Evaluation Board with the PC. The current distribution can be downloaded from http://drs.web.psi.ch/download . The Windows version contains a single program **`drs-xx.exe`** (where **`xx`** is the version) which can be executed to install the driver, applications, documentation and source code. Executing this file starts the installer:



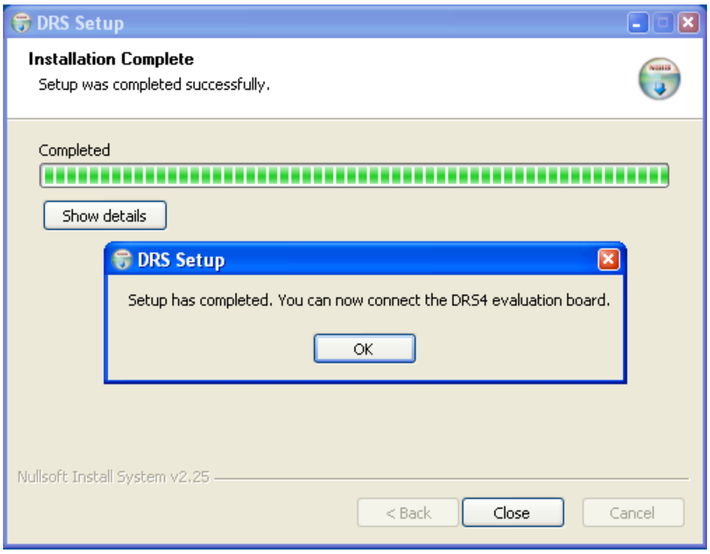
You can select which components to be installed:



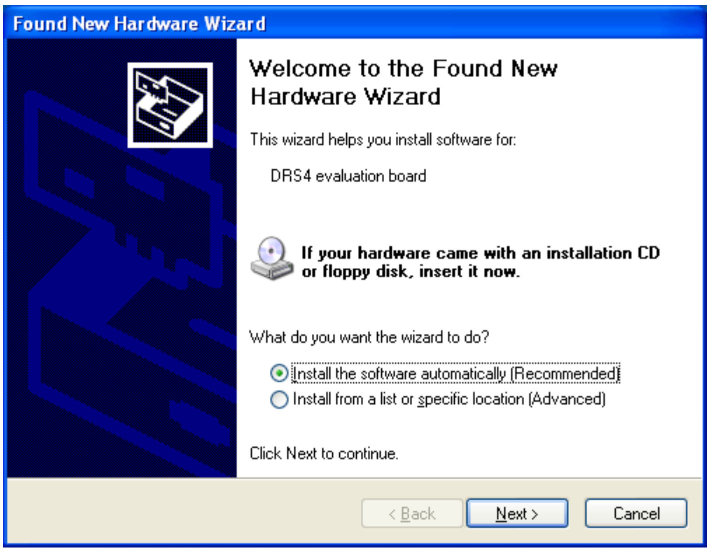
Then you can select the installation directory:



After the installer has finished, you can connect the DRS4 Evaluation Board to the Computer:



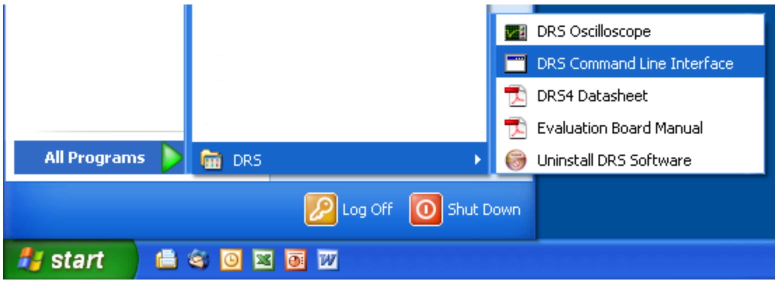
Now you will see the "Found New Hardware" dialog:



Where you can click "Install the software automatically" and then click "Next".

After successful installation of the driver, you will see the following window:

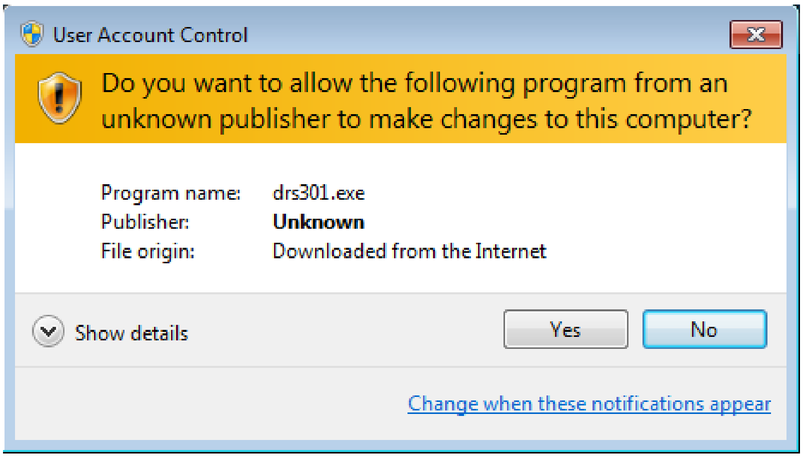
And a new group in your Start Menu:



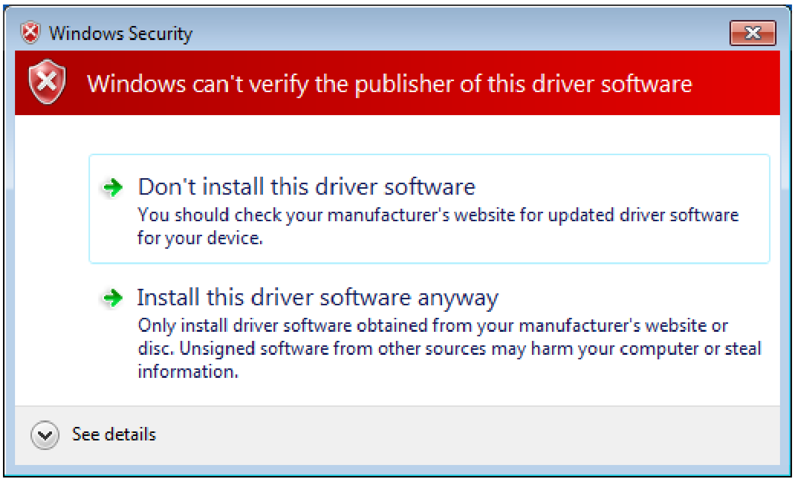
The software comes with two applications, a command line interface and an oscilloscope. These applications are explained in section 3.

2.2. Windows 7

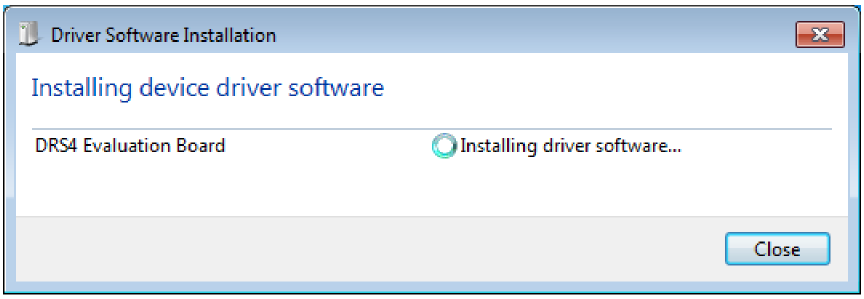
The installation under Windows 7 (32 and 64 bit) is basically the same than for Windows XP. You need the DRS software version 3.0.1 or later if you are using a 64-bit system, since version 3.0.0 and prior does not contain the 64-bit version of the **libusb** library. When you start the software installation, you get an additional screen which you have to confirm:



Make sure that you are logged in as an administrator to install the software. During the installation process, you will see a notice about some unverified driver software. Please select *"Install this driver software anyway"* to install the driver.



After the software has been installed, you can connect the DRS4 Evaluation Board to the computer. The driver installation should then start automatically and you will see this notification:



When this has finished, you can start the "drscl" and "drsosc" programs.

2.3. Linux

The drivers and applications are distributed for Linux in source code and must be compiled on each system. First untar the tar ball:

```
[/usr/local]$ tar -xzvf drs-1.0.tar.gz
drs-1.0/
drs-1.0/doc/
drs-1.0/doc/DRS4_rev06.pdf
drs-1.0/doc/manual.pdf
drs-1.0/include/
drs-1.0/include/ConfigDialog.h
drs-1.0/include/DOFrame.h
drs-1.0/include/DOScreen.h
. . .
```

Then change the directory and do a „make". Note that to compile the oscilloscope application it is necessary to have the wxWidgets package version 2.8.9 or later installed. You can obtain this package in source form from http://www.wxwidgets.org/downloads/. If this package is present, you can change to the drs directory and issue a make:

```
[/usr/local]$ cd drs-1.0
[/usr/local/drs-1.0]$ make
g++ -g -O2 -Wall -Wuninitialized -fno-strict-aliasing -Iinclude -DOS_LINUX
-DHAVE_LIBUSB -c src/musbstd.c
g++ -g -O2 -Wall -Wuninitialized -fno-strict-aliasing -Iinclude -DOS_LINUX
-DHAVE_LIBUSB -c src/mxml.c
...
```

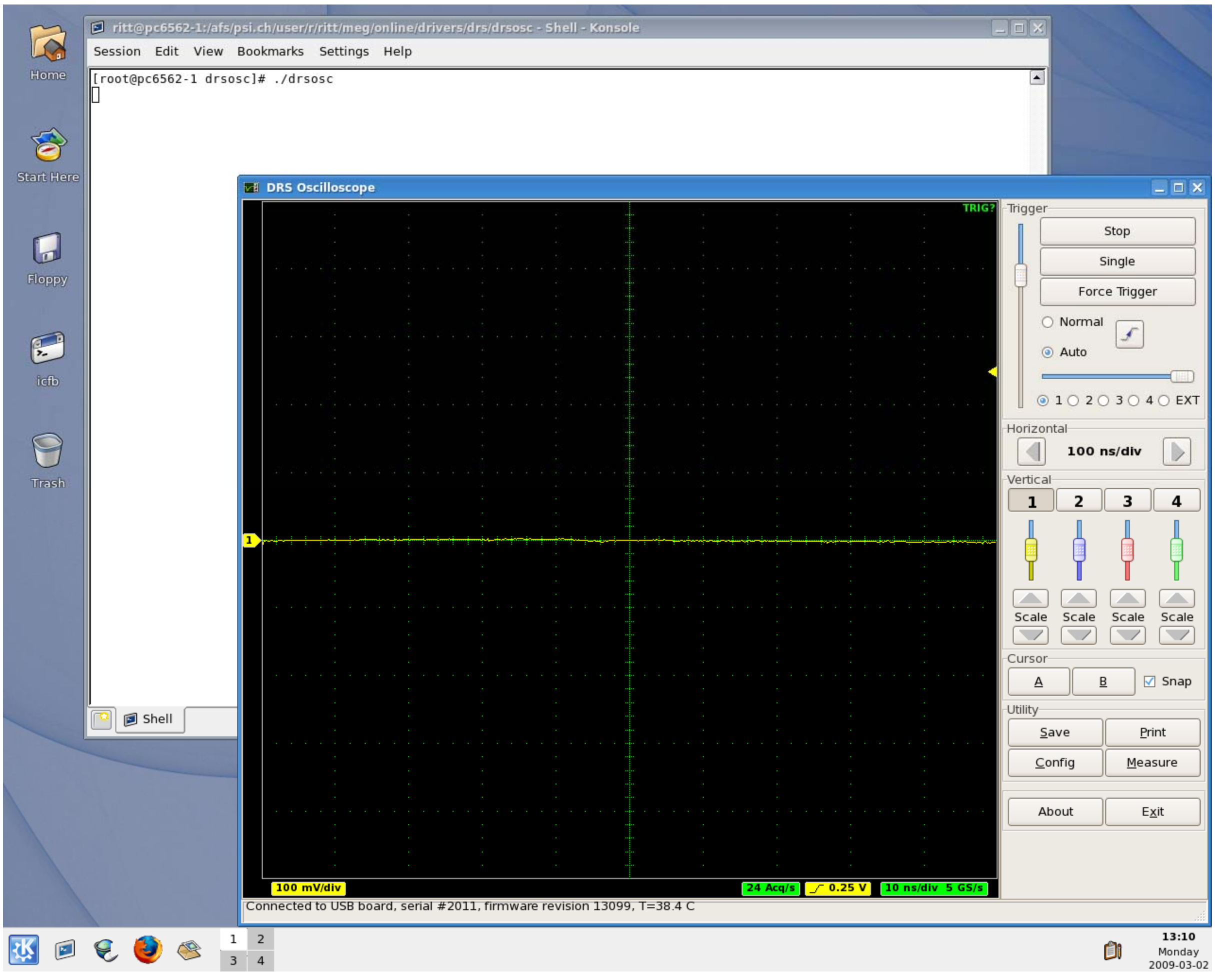
Now you can connect the DRS4 board to the PC. On systems where the "lsusb" tool is installed, one should be able to find the DRS4 evaluation board after connecting it with following command:

```
[/usr/local/drs-1.0]$ /sbin/lsusb -d 04b4:1175 –v

Bus 005 Device 005: ID 04b4:1175 Cypress Semiconductor Corp.
Device Descriptor:
  bLength                18
  bDescriptorType         1
  bcdUSB               2.00
  bDeviceClass            0 (Defined at Interface level)
  bDeviceSubClass         0
  bDeviceProtocol         1
  bMaxPacketSize0        64
  idVendor           0x04b4 Cypress Semiconductor Corp.
  idProduct          0x1175
  bcdDevice            0.01
  iManufacturer           1 S. Ritt PSI
  iProduct                2 DRS4 Evaluation Board
  iSerial                 3 REV1
  bNumConfigurations      1
  Configuration Descriptor:
    bLength                 9
    bDescriptorType         2
    wTotalLength           46
    bNumInterfaces          1
    bConfigurationValue     1
    iConfiguration          0
    bmAttributes         0x80
    MaxPower              500mA
...
```

If the board is correctly recognized, one can access it with the command line program. Under most Linux distributions however, only the "root" user can directly access USB devices. Some systems can be configured to allow non-root access via the "udev" system, but the exact instructions vary from distribution to distribution and can therefore not be given here.

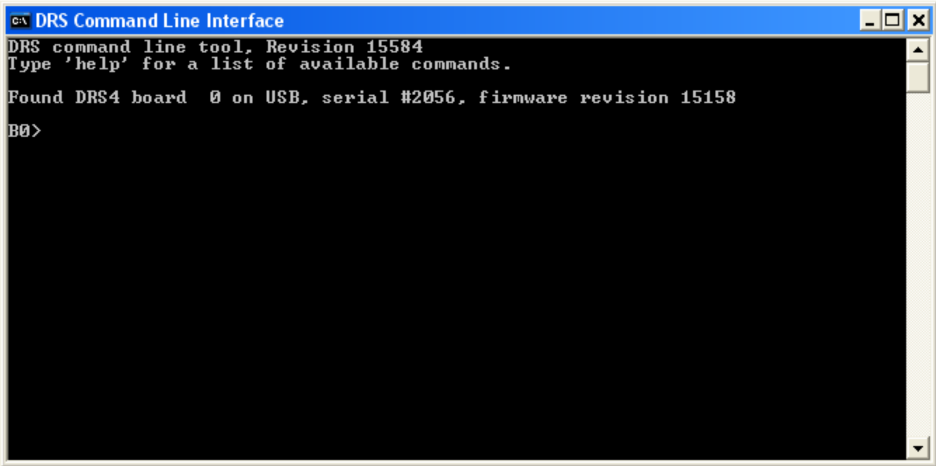
If the command line program works, the oscilloscope application "drsosc" can be started. It will open a X window and show exactly the same functionality as its Windows counterpart:



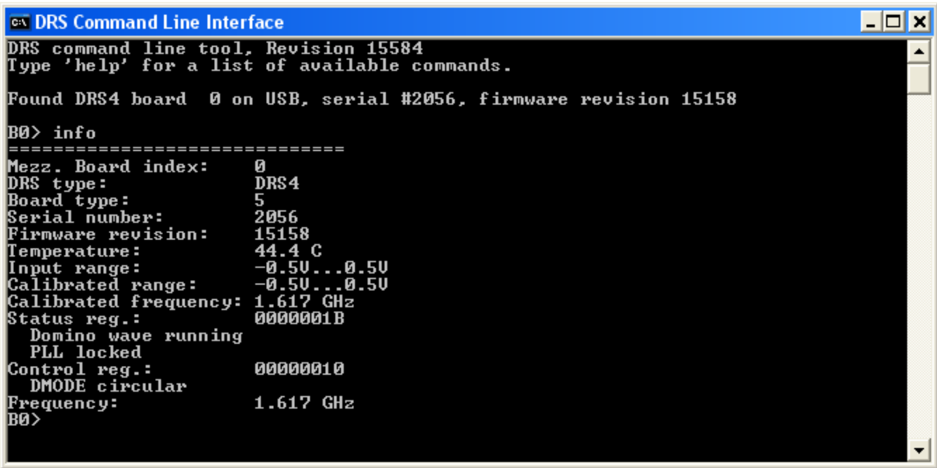
3. Running the Board

3.1. Command line Interface "drscl"

Clicking on „DRS Command Line Interface" (Windows) or entering "drscl" (Linux) will start a simple application which connects to the DRS4 Evaluation Board. If it finds the board, it displays the board serial number and the firmware revision as on the following screen shot:

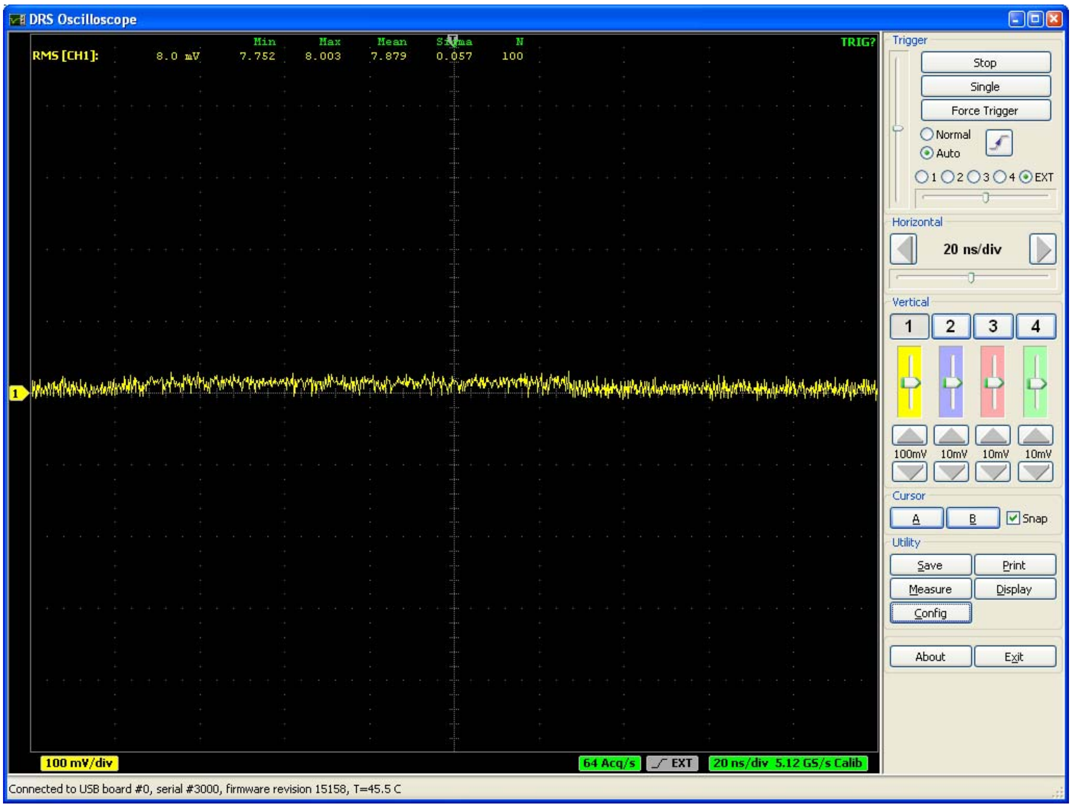


Now you are ready to issue your first command “info” which shows some more information, like the current board temperature. The temperature sensor is on the bottom side just below the DRS4 chip. If you keep issuing “info” commands and touch that sensor with your finger, you should see the temperature increase.



3.2. Oscilloscope application “drsosc”

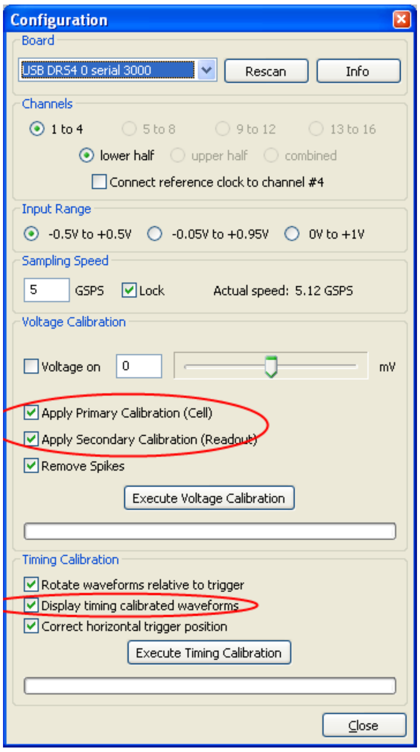
The second application is an oscilloscope-like program, which connects to the DRS4 board and works pretty much like a normal oscilloscope. You can select the trigger mode, trigger level and trigger source. On Rev. 1.1 of the DRS4 evaluation board, only CH1 can be selected as trigger source. You enable a channel by clicking on the number “1” to “4”. There are two cursors and a few utilities.



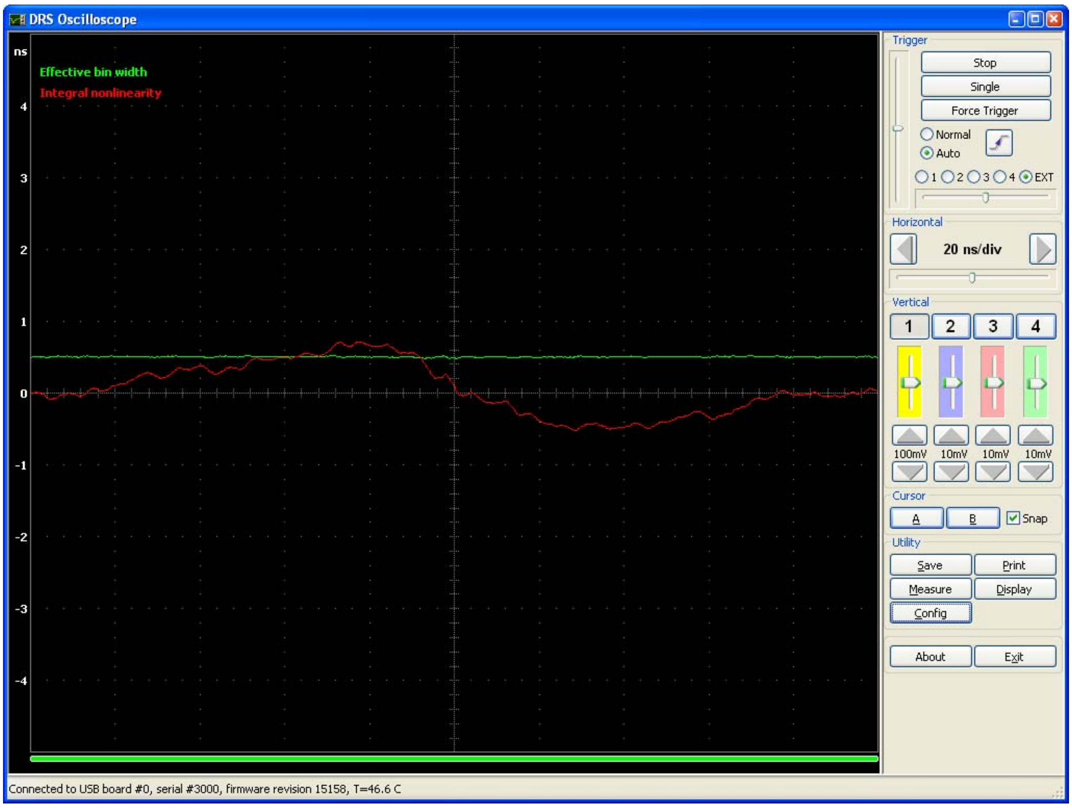
The picture above shows an un-calibrated evaluation board, which shows a noise level of about 8 mV RMS. After offset and gain calibrations, the noise level is reduced significantly:

The evaluation board Rev. 3 still shows some small random spikes. It is expected that future versions will improve this and reduce the noise level further.

The DRS4 evaluation board is shipped pre-calibrated in amplitude and time. This calibration can be turned on or off using the check boxes “Display calibrated waveforms” and “Display timing calibrated waveforms” in the “Config” Dialog:

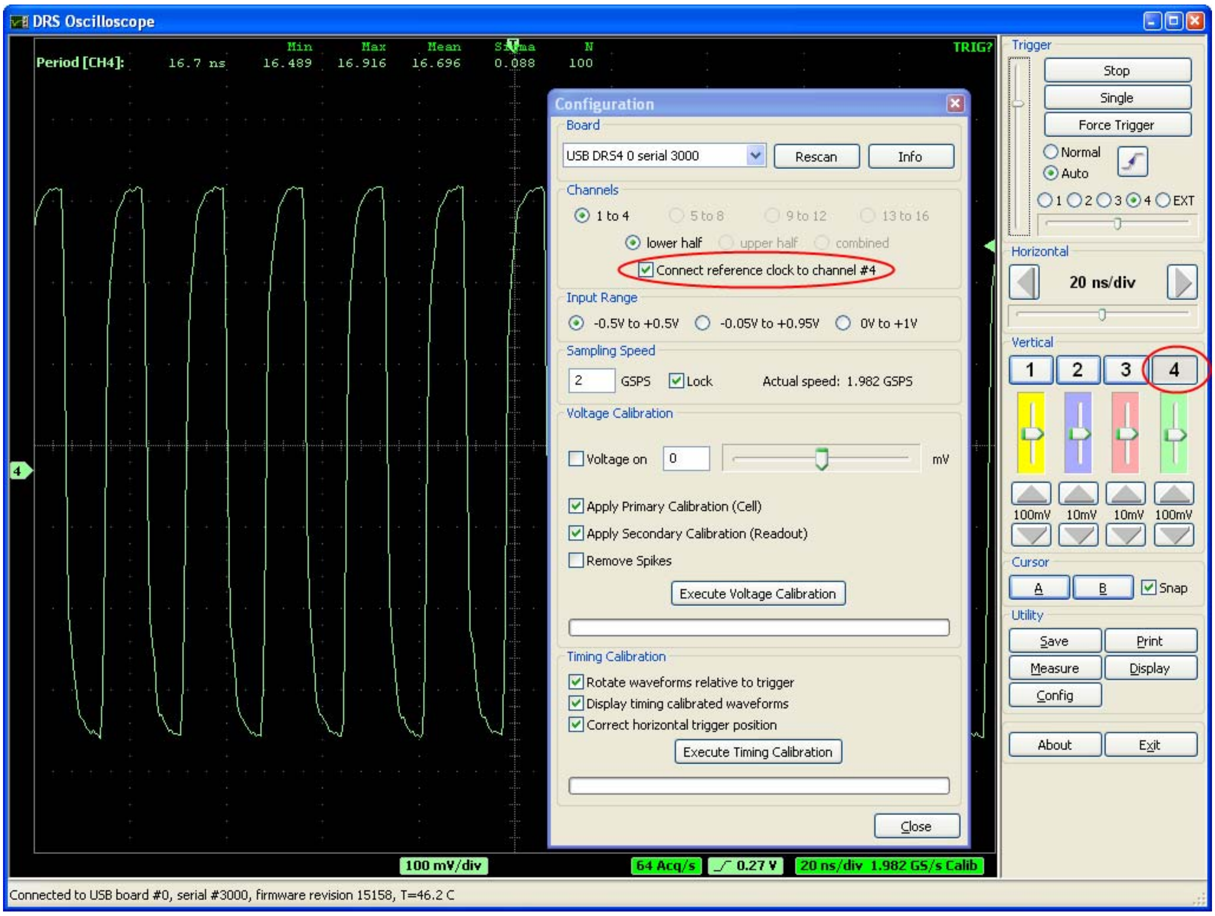


The calibration can be re-done any time by clicking on the “Execute Voltage Calibration” and “Execute Timing Calibration” buttons. For the voltage calibration, the inputs are switched to a calibration voltage generated by a DAC. Three calibration points (-0.4V, 0V, +0.4V) are taken and an offset and gain is evaluation. For the timing calibration, an internal 240 MHz clock is sampled in one channel and the deviation from the expected period to the measured period is used to determine the effective width of each cell. Following picture shows the result of such a timing calibration done at 2 GSPS. The effective bin width deviates only slightly from the nominal value of 0.5 ns, but the integral timing nonlinearity adds up to almost 1 ns, which is typical for the DRS4 chip. Since transistor parameters have normally gradients over the chip wafer, SCA chips are usually “faster” on one side compared to the other.



This calibration data both for voltage and timing is then stored in the EEPROM on the evaluation board, from where it is obtained each time the oscilloscope gets started. This assures that a board is calibrated even when used on different computers. Note however that the timing calibration is only valid for some sampling speed. So if you want to run the board at a different speed, you have to redo the timing calibration at that speed. The voltage calibration is a bit less dependent on the sampling speed, there is however some dependence on temperature. It is advised to keep the board running for a few minutes until the temperature shown in the status bar stabilizes before doing a voltage calibration.

For test purposes, an internal 60 MHz reference clock signal can be connected to channel #4 via the "Config" menu. To do so, activate channel #4, then select the "Config" menu and click on "Connect reference clock to channel #4":



The effect of the timing calibration can be tested by turning the timing calibration on and off via the "Display timing calibrated waveforms" check box.

You can save a waveform in an ASCII and a binary format by pressing the "Save" button. After you open a file, each trigger will write the waveform of the active channel(s) to that file. When you are continuously running, the file will grow very quickly. If the file has the extension ".xml" it will be written in ASCII form using XML encoding, otherwise a raw binary file will be written. Here is a snipped of such an XML file:


```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- created by MXML on Tue Feb 15 13:05:04 2011 -->
<DRSOSC>
  <Event>
    <Serial>1</Serial>
    <Time>2011/02/15 13:05:04.758</Time>
    <HUnit>ns</HUnit>
    <VUnit>mV</VUnit>
    <CHN1>
      <Data>20.7,-63.7</Data>
      <Data>21.7,-62.3</Data>
      ...
      <Data>1033.4,424.9</Data>
      <Data>1034.4,423.3</Data>
    </CHN1>
    <CHN2>
      <Data>20.7,-8.5</Data>
      <Data>21.7,-7.0</Data>
      ...
      <Data>1033.4,-8.3</Data>
      <Data>1034.4,-8.2</Data>
    </CHN2>
  </Event>
  <Event>
    <Serial>2</Serial>
    <Time>2011/02/15 13:05:04.883</Time>
    <HUnit>ns</HUnit>
    <VUnit>mV</VUnit>
    <CHN1>
      <Data>20.6,-63.0</Data>
      <Data>21.6,-63.8</Data>
      ...
```


Each individual event contains a header with the serial number of that event (starting with 1), and the date/time it was recorded. Then there is the channel data with pairs of time (in ns) and voltage (in mV). The number of channels depend on which channel was on when the data was recorded. It might be a single channel (CHN1) or all four channels (CHN1-CHN4). Please note that the XML format requires more space on your storage and takes also more CPU power to be written, so the maximum data rate is limited.

The binary format requires less space and can be written faster, but it requires a special program to read and analyze the data.

Word	Byte 0	Byte 1	Byte 2	Byte 3	Contents
0	'E'	'H'	'D'	'R'	Event Header
1	Serial number				Serial number starting with 1
2	Year		Month		Event date/time 16-bit values
3	Day		Hour		
4	Minute		Second		
5	Millisecond		*reserved*		
6	Time Bin #0				Time of sample bins in ns encoded in 4-Byte floating point format
7	Time Bin #1				
…	…				
1029	Time Bin 1023				
1030	'C'	'0'	'0'	'1'	Channel 1 header
1031	Voltage Bin #0		Voltage Bin #1		Channel 1 waveform data encoded in 2-Byte integers. 0=-0.5V and 65535=+0.5V
1032	Voltage Bin #2		Voltage Bin #3		
…	…		…		
1542	Voltage Bin #1022		Voltage Bin #1023		
1543	'C'	'0'	'0'	'2'	Channel 2 header
1544	Voltage Bin #0		Voltage Bin #1		Channel 2 waveform data encoded in 2-Byte integers. 0=-0.5V and 65535=+0.5V
1545	Voltage Bin #2		Voltage Bin #3		
…	…		…		
2055	Voltage Bin #1022		Voltage Bin #1023		
2056	'E'	'H'	'D'	'R'	Next Event Header
…					

Depending on the number of channels which are "on" during data acquisition, the file contains up to four channels, which can be identified by their channel headers. Then the next event follows, which can be identified by the event header 'EHDR'. All multi-byte data is encoded with LSB first, as in all Intel PC systems.

4. Development Hints

The idea behind the evaluation board is to make first steps in using the DRS4 chip, but then develop your own custom electronics around the chip. The first thing to do there is to study carefully the DRS4 data sheet, which can be obtained from http://drs.web.psi.ch/datasheets. Then have a look at the DRS4 Evaluation Board Reference Design, which schematics is supplied at the end of this document. When you start to design your own electronics, there are however some important points, which are not necessarily obvious from the data sheet or from the reference design. These points together with some design tips are explained in this section.

4.1. Power Supply

As with any analog design, the quality of the power supply is very important, since it has an influence of the noise level measured by the DRS4 chip. Low noise linear regulators together with the usual decoupling capacitors are recommended for all power supplies. The analog power supply AV_{DD} powers only the domino circuit of the DRS4 chip and directly influences the jitter of the sampling frequency. Long term variations in this power supply (seconds…) are regulated by the on-chip PLL, but high frequency noise in the MHz region leads directly to an increase of the PLL jitter. Therefore the evaluation board contains two separate 2.5V linear regulators for the DRS4 chip, one for the AV_{DD} power and one for the DV_{DD} power. Although the DV_{DD} power is called "digital power", it powers also the analog output buffers of the DRS4 chip and needs the same good quality than the AV_{DD} power in order to minimize the noise of the board.

The DRS4 chip also contains two grounds AGND and DGND. They can be either routed separately on the board and be connected at the power source, or they can be directly connected to an overall dedicated ground plane of the PCB. Tests have been shown that the latter choice gives slightly less noise.

The bottom of the QFN76 package of the DRS4 has an exposed paddle connected to the internal DGND. It is recommended that this paddle is matched by a PCB pad of similar size connected to analog ground to achieve the best electrical and thermal performance of the DRS4. The copper plane should have several vias to achieve a good heat dissipation to flow through the PCB as shown in Figure 2:

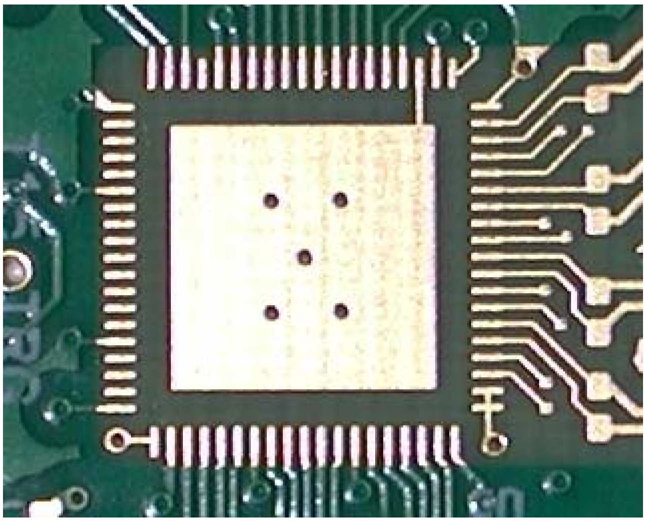


Figure 2: PCB pad under the DRS4 chip

These vias should be solder-filled or plugged. The maximum power dissipation of the DRS4 chip is not critical (350 mW), but an improved thermal stability helps the performance of any analog chip. To maximize the coverage and adhesion between the DRS4 and the PCB, the copper plane could be partitioned into several uniform sections, providing several tie points during the reflow process.

4.2. Analog Input

If non-differential signals should be digitized with the DRS4 chip, they must be converted into differential signals for the DRS4 inputs. The simplest solution is to connect the IN- inputs to AGND and to connect the signals directly to the IN+ inputs. This method has however the disadvantage that the crosstalk and noise immunity of the DRS4 chip are worsened. The evaluation board V3 uses differential drivers THS4508 from Texas Instruments® for this purpose. These drivers were carefully selected since they have to drive the relatively high DRS4 input current of almost 1 mA and capacitive load without compromising the analog bandwidth. The current design gives about 700 MHz (-3 dB) with moderate power consumption, so that the evaluation board can still be powered from the USB power (500 mA @ 5V). The linear regulator of the evaluation board V2 could however not be used, since the efficiency would be too low. Instead, a switching regulator LMZ10503 from National Semiconductor® is used on the board. It has an efficiency of more than 95% and a low output ripple. The output voltage of 3.3V is then converted using traditional linear regulators into two 2.5V low noise power supplies, which power the analog part of the board.

Other designs are possible which push the analog bandwidth to 800 or 900 MHz, close to the theoretical limit of 950 MHz of the DRS4 chip, but they require more power.

The usual design rules like proper termination, matched impedance PCB traces and separate power supply PCB planes apply as in any high frequency analog design.

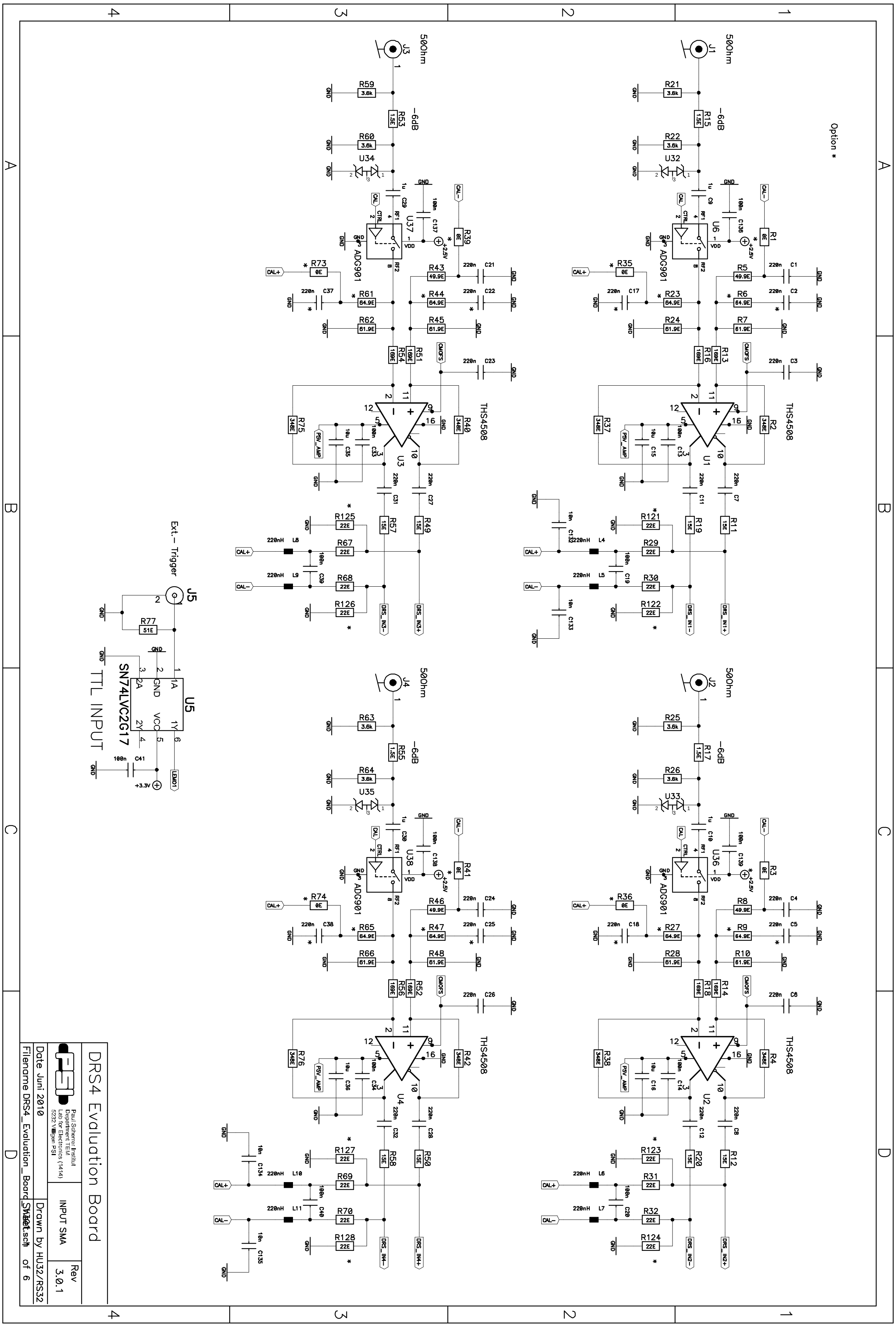
4.3. Control Voltages

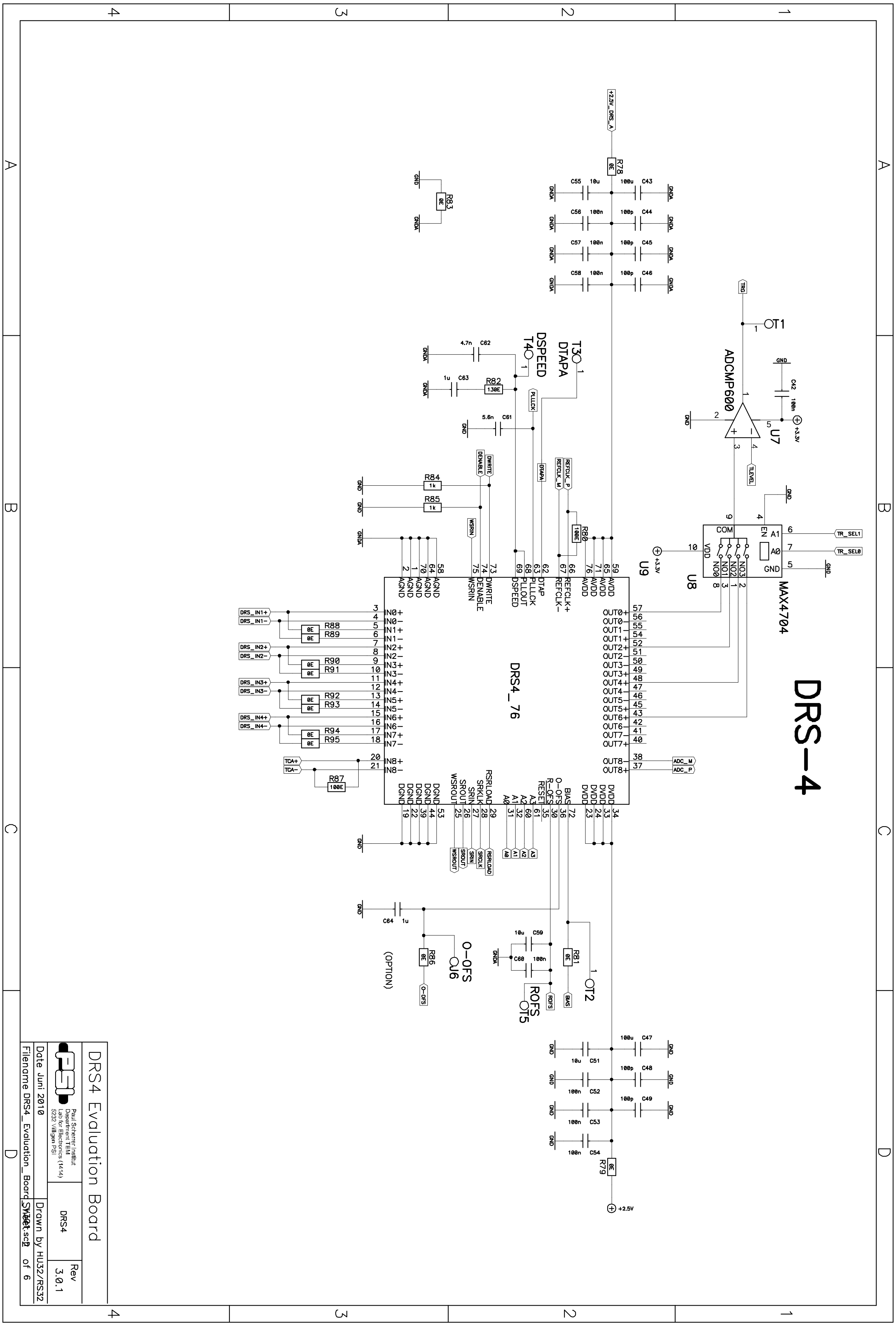
The DRS4 chip requires certain control voltages: ROFS, O-OFS and BIAS. The latter two are generated internally with some default voltage, but can be "overwritten" by an external low impedance source. It is recommended to connect these lines to an external 16-bit DAC, so that the DRS4 input range can be fine-tuned on a board-by-board basis, to compensate for chip variations. The ROFS signal should be driven by a high speed low noise buffer. If this signal would be directly connected to the DAC output, the signal height would change slightly during the chip readout and the measurement would show a varying baseline level.

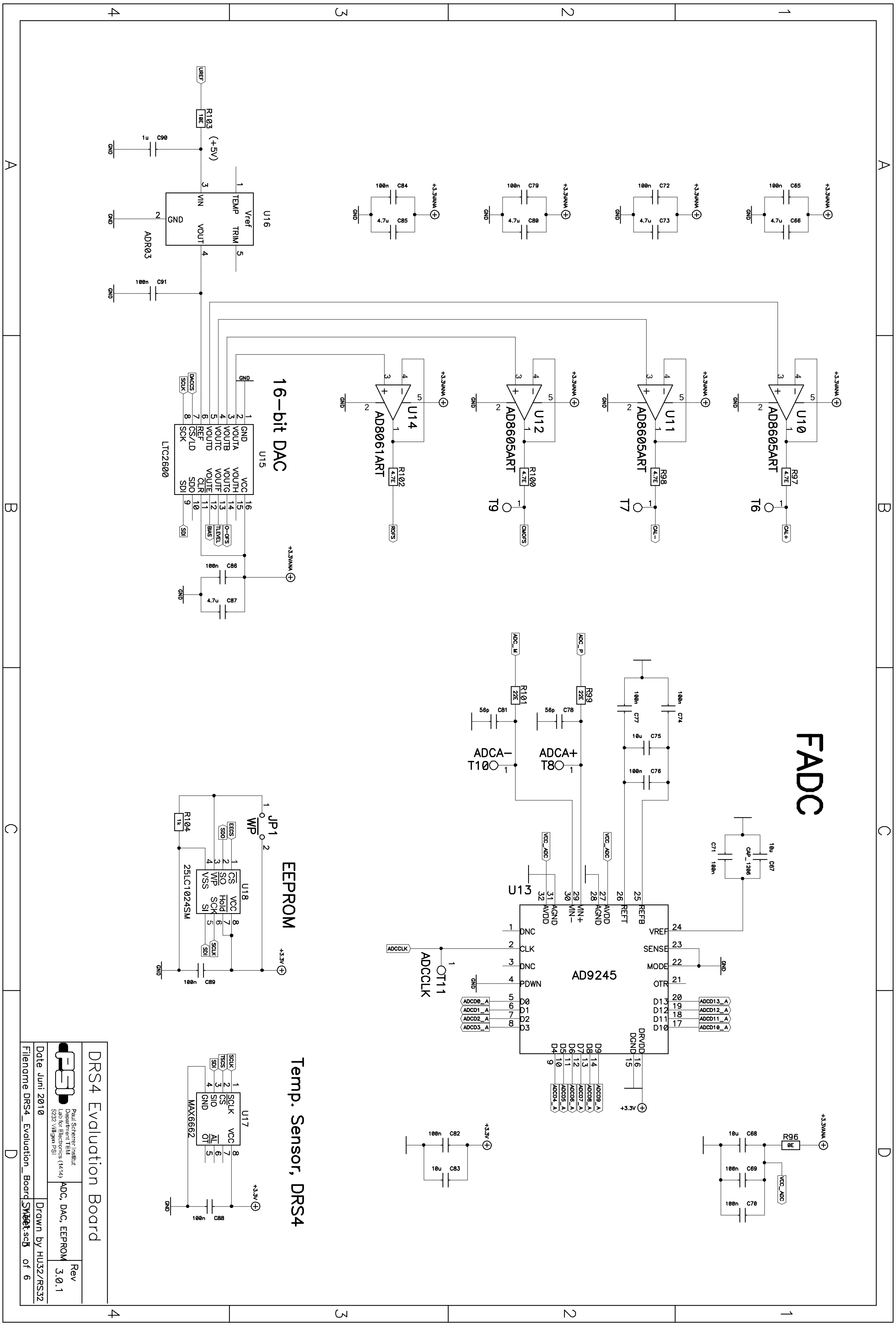
4.4. ADC Clock

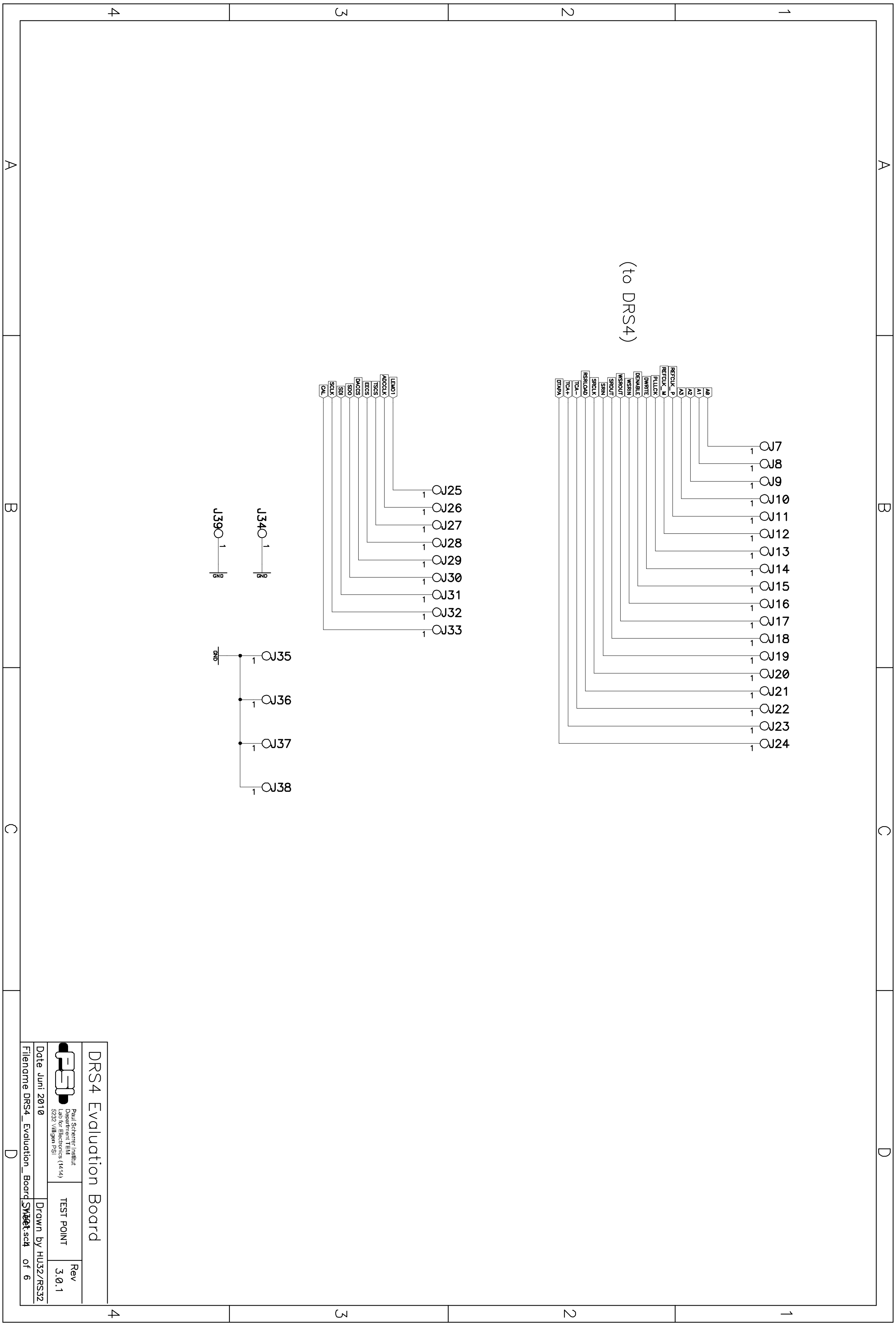
There is a very strict relation between the DRS4 output shift register clock SRCLK and the ADC clock (see DRS4 data sheet WAVEFORM READOUT). In order to reduce the noise due to aperture jitter, the phase shift between these two clocks must be fixed and contain very small jitter (~10ps). The easiest way to generate this phase shift is to use the digital clock managers (DCM) in the FPGA, as it is done on the evaluation board. Since the DCMs have however an inherent phase jitter of up to 150ps, this introduces some noise in form of a baseline variation when sampling a DC signal in the order of up to a few mV. If this becomes a problem, it is recommended to generate the phase shift between these two clocks with a low jitter delay circuit.

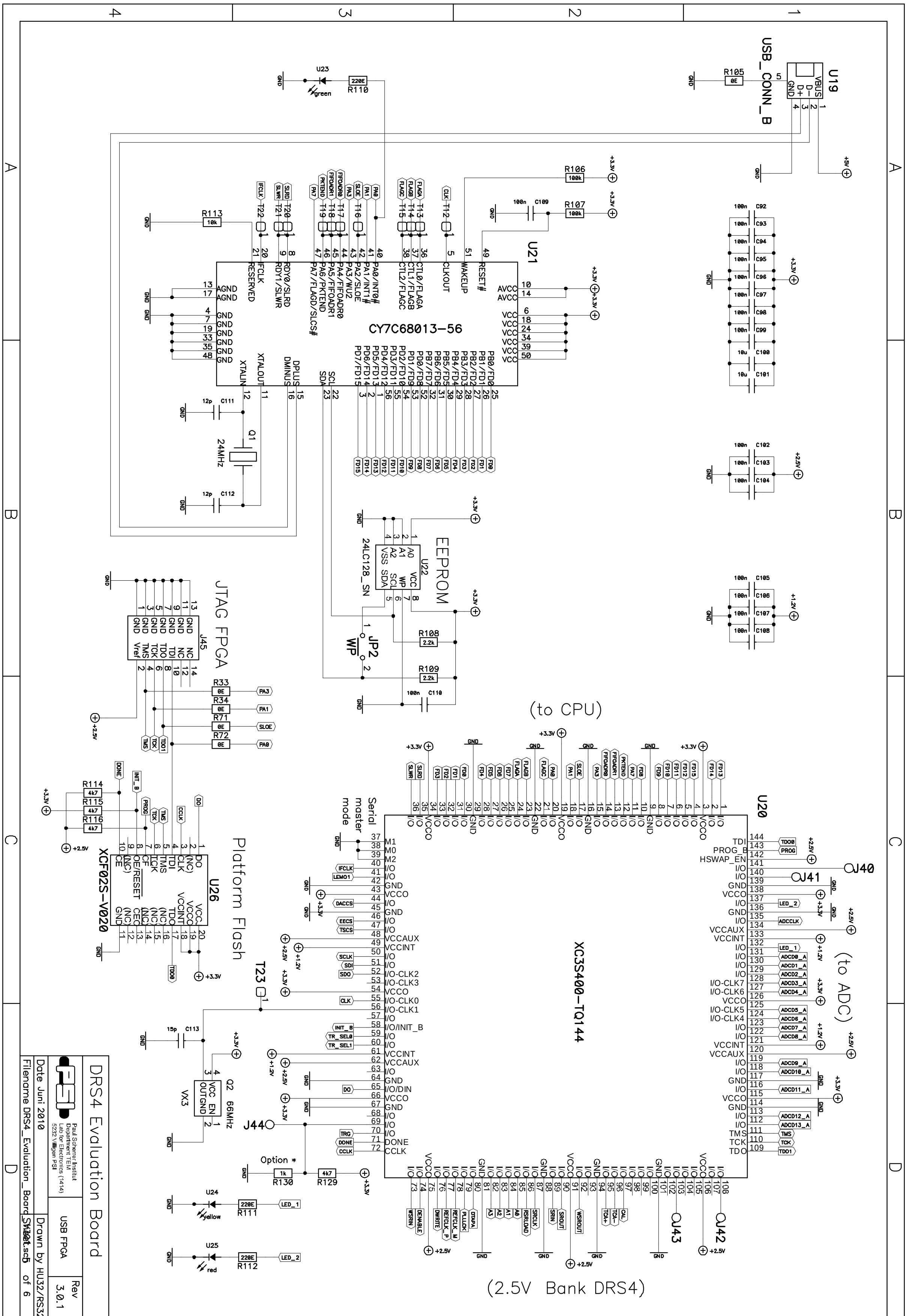
5. DRS4 Evaluation Board V3 Schematics

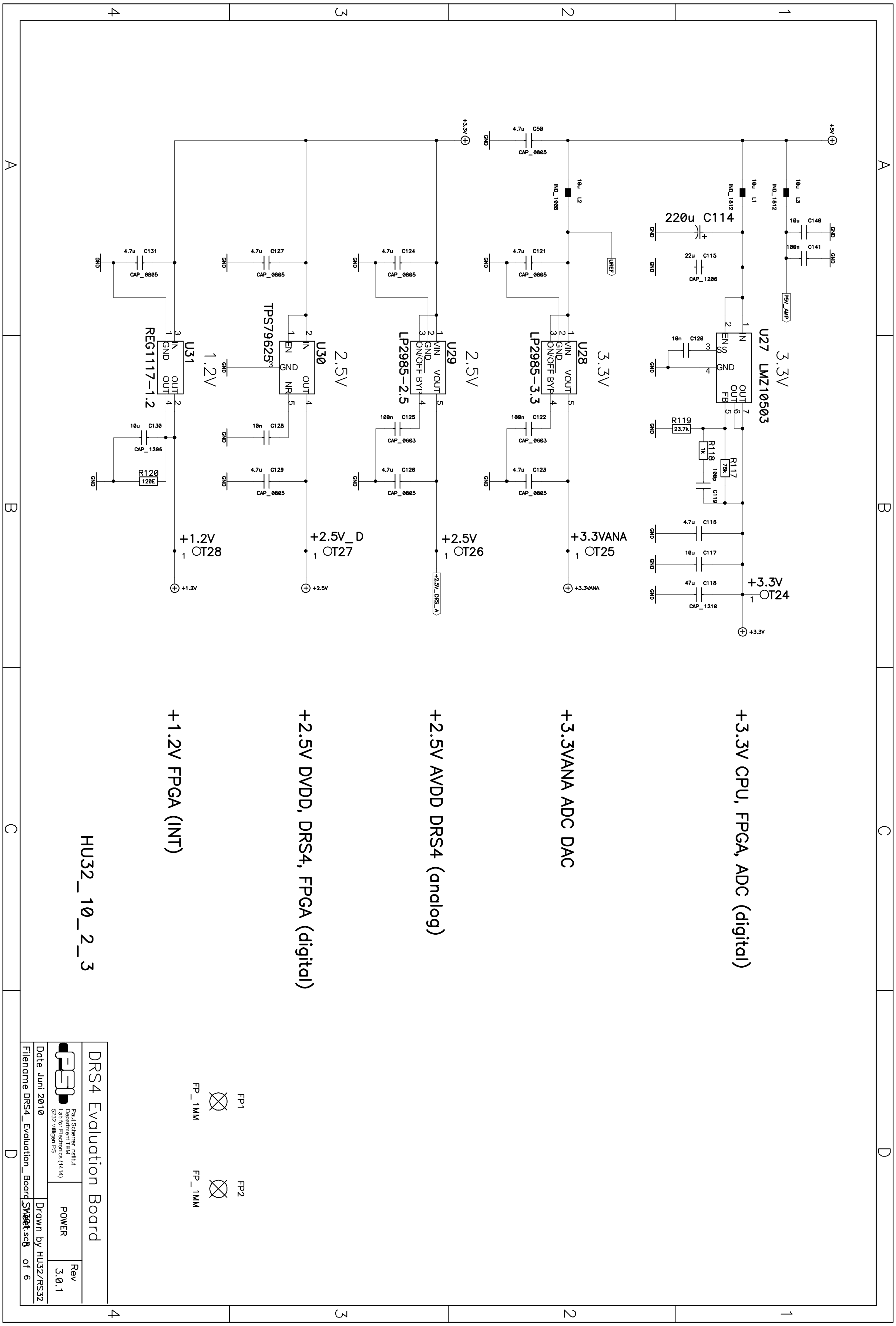



DRS-4
DRS4_76
ADCMP600
U7
MAX4704
U8
U9
DSPEED
DTAPA
ROFS
O-OFS
(OPTION)
DRS4 Evaluation Board
Paul Scherrer Institut
Date Juni 2010
Drawn by HU32/RS32
Rev 3.0.1
of 6


FADC
AD9245
U13
16-bit DAC
LTC2600
U15
ADR03
U16
AD8061ART
U14
AD8605ART
U12
U11
U10
EEPROM
25LC1024SM
U18
JP1
WP
Temp. Sensor, DRS4
MAX6662
U17
ADCCLK
DRS4 Evaluation Board
ADC, DAC, EEPROM
Rev
3.0.1
Date Juni 2010
Drawn by HU32/RS32
of 6


(to DRS4)
J7
J8
J9
J10
J11
J12
J13
J14
J15
J16
J17
J18
J19
J20
J21
J22
J23
J24
J25
J26
J27
J28
J29
J30
J31
J32
J33
J34
J39
GND
J35
J36
J37
J38
DRS4 Evaluation Board
TEST POINT
Rev
3.0.1
Date Juni 2010
Drawn by HU32/RS32
of 6


DRS4 Evaluation Board
USB FPGA
Rev 3.0.1
Date Juni 2010
Drawn by HU32/RS32
CY7C68013-56
XC3S400-TQ144
EEPROM
24LC128_SN
JTAG FPGA
Platform Flash
XCF02S-V020
USB_CONN_B
(to CPU)
(to ADC)
(2.5V Bank DRS4)
Serial master mode
24MHz
66MHz


+3.3V CPU, FPGA, ADC (digital)
+3.3VANA ADC DAC
+2.5V AVDD DRS4 (analog)
+2.5V DVDD, DRS4, FPGA (digital)
+1.2V FPGA (INT)
HU32_10_2_3
U27 LMZ10503
3.3V
U28
LP2985-3.3
3.3V
U29
LP2985-2.5
2.5V
U30
TPS79625
2.5V
U31
REG1117-1.2
1.2V
220u C114
+3.3V
T24
+3.3VANA
T25
+2.5V
T26
+2.5V_D
T27
+1.2V
T28
FP1
FP2
FP_1MM
DRS4 Evaluation Board
POWER
Rev
3.0.1
Date Juni 2010
Drawn by HU32/RS32

6. DRS4 Evaluation Board V3 Bill of Materials

Count	ComponentName	RefDes	PatternName	Value	Description
1	24LC128_SN	U22	SO-G8		EEPROM 16k x 8
1	25LC1024SM	U18	SO-8-SM		EEPROM
1	44.021.0547	J5	CONN_PSI2	Trigger	Lemo 00 90° 50Ohm
1	AD8061ART	U14	SOT23-5		Amplifier
3	AD8605ART	U10	SOT23-5		Amplifier
	AD8605ART	U11	SOT23-5		Amplifier
	AD8605ART	U12	SOT23-5		Amplifier
1	AD9245	U13	LFCSP_VQ_32		ADC
1	ADCMP600	U7	SOT23-5		Comparator
4	ADG901	U6	CP-8-2		Wideband SPST Switches
	ADG901	U36	CP-8-2		Wideband SPST Switches
	ADG901	U37	CP-8-2		Wideband SPST Switches
	ADG901	U38	CP-8-2		Wideband SPST Switches
1	ADR03	U16	SC70-5	2.5V	Ref.
1	CAPP	C114	CAPP	220u	Tantal, >=10V
12	CAP_0402	C13	0402	100n	Capacitor >=10V
	CAP_0402	C14	0402	100n	Capacitor >=10V
	CAP_0402	C19	0402	100n	Capacitor >=10V
	CAP_0402	C20	0402	100n	Capacitor >=10V
	CAP_0402	C33	0402	100n	Capacitor >=10V
	CAP_0402	C34	0402	100n	Capacitor >=10V
	CAP_0402	C39	0402	100n	Capacitor >=10V
	CAP_0402	C40	0402	100n	Capacitor >=10V
	CAP_0402	C136	0402	100n	Capacitor >=10V
	CAP_0402	C137	0402	100n	Capacitor >=10V
	CAP_0402	C138	0402	100n	Capacitor >=10V
	CAP_0402	C139	0402	100n	Capacitor >=10V
5	CAP_0402	C44	0402	100p	Capacitor >=10V
	CAP_0402	C45	0402	100p	Capacitor >=10V
	CAP_0402	C46	0402	100p	Capacitor >=10V
	CAP_0402	C48	0402	100p	Capacitor >=10V
	CAP_0402	C49	0402	100p	Capacitor >=10V
16	CAP_0402	C1	0402	220n	Capacitor >=10V

	CAP_0402	C2	0402	220n	Capacitor	>=10V
	CAP_0402	C3	0402	220n	Capacitor	>=10V
	CAP_0402	C4	0402	220n	Capacitor	>=10V
	CAP_0402	C5	0402	220n	Capacitor	>=10V
	CAP_0402	C6	0402	220n	Capacitor	>=10V
	CAP_0402	C7	0402	220n	Capacitor	>=10V
	CAP_0402	C8	0402	220n	Capacitor	>=10V
	CAP_0402	C11	0402	220n	Capacitor	>=10V
	CAP_0402	C12	0402	220n	Capacitor	>=10V
	CAP_0402	C17	0402	220n	Capacitor	>=10V
	CAP_0402	C18	0402	220n	Capacitor	>=10V
	CAP_0402	C21	0402	220n	Capacitor	>=10V
	CAP_0402	C22	0402	220n	Capacitor	>=10V
	CAP_0402	C23	0402	220n	Capacitor	>=10V
	CAP_0402	C24	0402	220n	Capacitor	>=10V
	CAP_0402	C25	0402	220n	Capacitor	>=10V
	CAP_0402	C26	0402	220n	Capacitor	>=10V
	CAP_0402	C27	0402	220n	Capacitor	>=10V
	CAP_0402	C28	0402	220n	Capacitor	>=10V
	CAP_0402	C31	0402	220n	Capacitor	>=10V
	CAP_0402	C32	0402	220n	Capacitor	>=10V
	CAP_0402	C37	0402	220n	Capacitor	>=10V
	CAP_0402	C38	0402	220n	Capacitor	>=10V
2	CAP_0603	C64	0603	1u	Capacitor	>=10V
	CAP_0603	C90	0603	1u	Capacitor	>=10V
4	CAP_0603	C9	0603	1u	Capacitor	>=10V
	CAP_0603	C10	0603	1u	Capacitor	>=10V
	CAP_0603	C29	0603	1u	Capacitor	>=10V
	CAP_0603	C30	0603	1u	Capacitor	>=10V
1	CAP_0603	C62	0603	4.7n	Capacitor	>=10V
1	CAP_0603	C61	0603	5.6n	Capacitor	>=10V
6	CAP_0603	C120	0603	10n	Capacitor	>=10V
	CAP_0603	C128	0603	10n	Capacitor	>=10V
	CAP_0603	C132	0603	10n	Capacitor	>=10V
	CAP_0603	C133	0603	10n	Capacitor	>=10V
	CAP_0603	C134	0603	10n	Capacitor	>=10V
	CAP_0603	C135	0603	10n	Capacitor	>=10V
2	CAP_0603	C111	0603	12p	Capacitor	>=10V
	CAP_0603	C112	0603	12p	Capacitor	>=10V
1	CAP_0603	C113	0603	15p	Capacitor	>=10V

1	CAP_0603	C63	0603	1u	Capacitor	>=10V
2	CAP_0603	C78	0603	56p	Capacitor	>=10V
	CAP_0603	C81	0603	56p	Capacitor	>=10V
42	CAP_0603	C41	0603	100n	Capacitor	>=10V
	CAP_0603	C42	0603	100n	Capacitor	>=10V
	CAP_0603	C52	0603	100n	Capacitor	>=10V
	CAP_0603	C53	0603	100n	Capacitor	>=10V
	CAP_0603	C54	0603	100n	Capacitor	>=10V
	CAP_0603	C56	0603	100n	Capacitor	>=10V
	CAP_0603	C57	0603	100n	Capacitor	>=10V
	CAP_0603	C58	0603	100n	Capacitor	>=10V
	CAP_0603	C60	0603	100n	Capacitor	>=10V
	CAP_0603	C65	0603	100n	Capacitor	>=10V
	CAP_0603	C69	0603	100n	Capacitor	>=10V
	CAP_0603	C70	0603	100n	Capacitor	>=10V
	CAP_0603	C71	0603	100n	Capacitor	>=10V
	CAP_0603	C72	0603	100n	Capacitor	>=10V
	CAP_0603	C74	0603	100n	Capacitor	>=10V
	CAP_0603	C76	0603	100n	Capacitor	>=10V
	CAP_0603	C77	0603	100n	Capacitor	>=10V
	CAP_0603	C79	0603	100n	Capacitor	>=10V
	CAP_0603	C82	0603	100n	Capacitor	>=10V
	CAP_0603	C84	0603	100n	Capacitor	>=10V
	CAP_0603	C86	0603	100n	Capacitor	>=10V
	CAP_0603	C88	0603	100n	Capacitor	>=10V
	CAP_0603	C89	0603	100n	Capacitor	>=10V
	CAP_0603	C91	0603	100n	Capacitor	>=10V
	CAP_0603	C92	0603	100n	Capacitor	>=10V
	CAP_0603	C93	0603	100n	Capacitor	>=10V
	CAP_0603	C94	0603	100n	Capacitor	>=10V
	CAP_0603	C95	0603	100n	Capacitor	>=10V
	CAP_0603	C96	0603	100n	Capacitor	>=10V
	CAP_0603	C97	0603	100n	Capacitor	>=10V
	CAP_0603	C98	0603	100n	Capacitor	>=10V
	CAP_0603	C99	0603	100n	Capacitor	>=10V
	CAP_0603	C102	0603	100n	Capacitor	>=10V
	CAP_0603	C103	0603	100n	Capacitor	>=10V
	CAP_0603	C104	0603	100n	Capacitor	>=10V
	CAP_0603	C105	0603	100n	Capacitor	>=10V
	CAP_0603	C106	0603	100n	Capacitor	>=10V

	CAP_0603	C107	0603	100n	Capacitor	>=10V
	CAP_0603	C108	0603	100n	Capacitor	>=10V
	CAP_0603	C110	0603	100n	Capacitor	>=10V
	CAP_0603	C122	0603	100n	Capacitor	>=10V
	CAP_0603	C125	0603	100n	Capacitor	>=10V
	CAP_0603	C125	0603	100n	Capacitor	>=10V
	CAP_0603	C125	0603	100n	Capacitor	>=10V
	CAP_0603	C141	0603	100n	Capacitor	>=10V
1	CAP_0603	C119	0603	100p	Capacitor	>=10V
14	CAP_0805	C50	0805	4.7u	Capacitor	>=10V
	CAP_0805	C66	0805	4.7u	Capacitor	>=10V
	CAP_0805	C73	0805	4.7u	Capacitor	>=10V
	CAP_0805	C80	0805	4.7u	Capacitor	>=10V
	CAP_0805	C85	0805	4.7u	Capacitor	>=10V
	CAP_0805	C87	0805	4.7u	Capacitor	>=10V
	CAP_0805	C116	0805	4.7u	Capacitor	>=10V
	CAP_0805	C121	0805	4.7u	Capacitor	>=10V
	CAP_0805	C123	0805	4.7u	Capacitor	>=10V
	CAP_0805	C124	0805	4.7u	Capacitor	>=10V
	CAP_0805	C126	0805	4.7u	Capacitor	>=10V
	CAP_0805	C127	0805	4.7u	Capacitor	>=10V
	CAP_0805	C129	0805	4.7u	Capacitor	>=10V
	CAP_0805	C131	0805	4.7u	Capacitor	>=10V
15	CAP_1206	C15	1206	10u	Capacitor	>=10V
	CAP_1206	C16	1206	10u	Capacitor	>=10V
	CAP_1206	C35	1206	10u	Capacitor	>=10V
	CAP_1206	C36	1206	10u	Capacitor	>=10V
	CAP_1206	C51	1206	10u	Capacitor	>=10V
	CAP_1206	C55	1206	10u	Capacitor	>=10V
	CAP_1206	C59	1206	10u	Capacitor	>=10V
	CAP_1206	C67	1206	10u	Capacitor	>=10V
	CAP_1206	C68	1206	10u	Capacitor	>=10V
	CAP_1206	C75	1206	10u	Capacitor	>=10V
	CAP_1206	C83	1206	10u	Capacitor	>=10V
	CAP_1206	C100	1206	10u	Capacitor	>=10V
	CAP_1206	C101	1206	10u	Capacitor	>=10V
	CAP_1206	C117	1206	10u	Capacitor	>=10V
	CAP_1206	C130	1206	10u	Capacitor	>=10V
1	CAP_1206	C115	1206	22u	Capacitor	>=10V
1	CAP_1206	C109	1206	100n	Capacitor	>=10V

1	CAP_1206	C140	1206	10u	Capacitor >=10V
1	CAP_1210	C118	1210	47u	Capacitor >=10V
2	CAP_1210	C43	1210	100u	Capacitor >=10V
	CAP_1210	C47	1210	100u	Capacitor **>=6.3V**
1	CONN_MOLEX_JTAG_FPGA	J45	DIL14P_2MM	JTAG	
1	CY7C68013-56	U21	SSO-G56		Microcontroller
1	DRS4_76	U6	QFN-76		DRS4
8	IND_0603	L4	0603	220nH	Inductor
	IND_0603	L5	0603	220nH	Inductor
	IND_0603	L6	0603	220nH	Inductor
	IND_0603	L7	0603	220nH	Inductor
	IND_0603	L8	0603	220nH	Inductor
	IND_0603	L9	0603	220nH	Inductor
	IND_0603	L10	0603	220nH	Inductor
	IND_0603	L11	0603	220nH	Inductor
1	IND_1008	L2	1008	10u	Inductor
2	IND_1812	L1	1812	10u	Inductor
	IND_1812	L3	1812	10u	Inductor
2	JMP2MM	JP1	TP50MIL	~WP	Jumper
	JMP2MM	JP2	TP50MIL	~WP	Jumper
1	LED_PLCC-4	U23	PLCC-4	green	
1	LED_PLCC-4	U25	PLCC-4	red	
1	LED_PLCC-4	U24	PLCC-4	yellow	
1	LMZ10503	U27	TO-PMOD-7		Power Module 3A
1	LP2985-2.5	U29	SOT23-5		150mA Low Dropout
1	LP2985-3.3	U28	SOT23-5		150mA Low Dropout
1	LTC2600	U15	SSOP16		DAC
1	MAX4704	U8	MSOP-10		MUX
1	MAX6662	U17	SO-G8		Temp. Sensor
1	QUARZ_NKS7	Q1	QUARZ_NKS7	24MHz	JXS75-12-30/30
4	RCLAMP0502B	U32	SOT523		ESD Protection
	RCLAMP0502B	U33	SOT523		ESD Protection
	RCLAMP0502B	U34	SOT523		ESD Protection
	RCLAMP0502B	U35	SOT523		ESD Protection
1	REG1117-1.2	U31	SOT223		REG 1117A Low Dropout
16	RES_0402	R1	0402	0E	Resistor
	RES_0402	R3	0402	0E	Resistor
	RES_0402	R35	0402	0E	Resistor
	RES_0402	R36	0402	0E	Resistor

	RES_0402	R39	0402	0E	Resistor
	RES_0402	R41	0402	0E	Resistor
	RES_0402	R73	0402	0E	Resistor
	RES_0402	R74	0402	0E	Resistor
	RES_0402	R88	0402	0E	Resistor
	RES_0402	R89	0402	0E	Resistor
	RES_0402	R90	0402	0E	Resistor
	RES_0402	R91	0402	0E	Resistor
	RES_0402	R92	0402	0E	Resistor
	RES_0402	R93	0402	0E	Resistor
	RES_0402	R94	0402	0E	Resistor
	RES_0402	R95	0402	0E	Resistor
8	RES_0402	R11	0402	15E	Resistor
	RES_0402	R12	0402	15E	Resistor
	RES_0402	R19	0402	15E	Resistor
	RES_0402	R20	0402	15E	Resistor
	RES_0402	R49	0402	15E	Resistor
	RES_0402	R50	0402	15E	Resistor
	RES_0402	R57	0402	15E	Resistor
	RES_0402	R58	0402	15E	Resistor
8	RES_0402	R121	0402	22E	Resistor
	RES_0402	R122	0402	22E	Resistor
	RES_0402	R123	0402	22E	Resistor
	RES_0402	R124	0402	22E	Resistor
	RES_0402	R125	0402	22E	Resistor
	RES_0402	R126	0402	22E	Resistor
	RES_0402	R127	0402	22E	Resistor
	RES_0402	R128	0402	22E	Resistor
8	RES_0402	R29	0402	22E	Resistor
	RES_0402	R30	0402	22E	Resistor
	RES_0402	R31	0402	22E	Resistor
	RES_0402	R32	0402	22E	Resistor
	RES_0402	R67	0402	22E	Resistor
	RES_0402	R68	0402	22E	Resistor
	RES_0402	R69	0402	22E	Resistor
	RES_0402	R70	0402	22E	Resistor
4	RES_0402	R5	0402	49.9E	Resistor
	RES_0402	R8	0402	49.9E	Resistor
	RES_0402	R43	0402	49.9E	Resistor
	RES_0402	R46	0402	49.9E	Resistor

8	RES_0402	R7	0402	61.9E	Resistor
	RES_0402	R10	0402	61.9E	Resistor
	RES_0402	R24	0402	61.9E	Resistor
	RES_0402	R28	0402	61.9E	Resistor
	RES_0402	R45	0402	61.9E	Resistor
	RES_0402	R48	0402	61.9E	Resistor
	RES_0402	R62	0402	61.9E	Resistor
	RES_0402	R66	0402	61.9E	Resistor
8	RES_0402	R6	0402	64.9E	Resistor
	RES_0402	R9	0402	64.9E	Resistor
	RES_0402	R23	0402	64.9E	Resistor
	RES_0402	R27	0402	64.9E	Resistor
	RES_0402	R44	0402	64.9E	Resistor
	RES_0402	R47	0402	64.9E	Resistor
	RES_0402	R61	0402	64.9E	Resistor
	RES_0402	R65	0402	64.9E	Resistor
8	RES_0402	R13	0402	169E	Resistor
	RES_0402	R14	0402	169E	Resistor
	RES_0402	R16	0402	169E	Resistor
	RES_0402	R18	0402	169E	Resistor
	RES_0402	R51	0402	169E	Resistor
	RES_0402	R52	0402	169E	Resistor
	RES_0402	R54	0402	169E	Resistor
	RES_0402	R56	0402	169E	Resistor
8	RES_0402	R2	0402	348E	Resistor
	RES_0402	R4	0402	348E	Resistor
	RES_0402	R37	0402	348E	Resistor
	RES_0402	R38	0402	348E	Resistor
	RES_0402	R40	0402	348E	Resistor
	RES_0402	R42	0402	348E	Resistor
	RES_0402	R75	0402	348E	Resistor
	RES_0402	R76	0402	348E	Resistor
1	RES_0603	R81	0603	0E	Resistor
5	RES_0603	R84	0603	1k	Resistor
	RES_0603	R85	0603	1k	Resistor
	RES_0603	R104	0603	1k	Resistor
	RES_0603	R118	0603	1k	Resistor
	RES_0603	R130	0603	1k	Resistor
2	RES_0603	R108	0603	2.2k	Resistor
	RES_0603	R109	0603	2.2k	Resistor

3	RES_0603	R114	0603	4k7	Resistor
	RES_0603	R115	0603	4k7	Resistor
	RES_0603	R116	0603	4k7	Resistor
1	RES_0603	R129	0603	4k7	Resistor
4	RES_0603	R97	0603	4.7E	Resistor
	RES_0603	R98	0603	4.7E	Resistor
	RES_0603	R100	0603	4.7E	Resistor
	RES_0603	R102	0603	4.7E	Resistor
1	RES_0603	R103	0603	10E	Resistor
1	RES_0603	R113	0603	10k	Resistor
2	RES_0603	R99	0603	22E	Resistor
	RES_0603	R101	0603	22E	Resistor
1	RES_0603	R119	0603	23.7k	Resistor
4	RES_0603	R15	0603	1.5E	Resistor
	RES_0603	R17	0603	1.5E	Resistor
	RES_0603	R53	0603	1.5E	Resistor
	RES_0603	R55	0603	1.5E	Resistor
1	RES_0603	R117	0603	75k	Resistor
2	RES_0603	R80	0603	100E	Resistor
	RES_0603	R87	0603	100E	Resistor
2	RES_0603	R106	0603	100k	Resistor
	RES_0603	R107	0603	100k	Resistor
8	RES_0603	R21	0603	1k	Resistor
	RES_0603	R22	0603	1k	Resistor
	RES_0603	R25	0603	1k	Resistor
	RES_0603	R26	0603	1k	Resistor
	RES_0603	R59	0603	1k	Resistor
	RES_0603	R60	0603	1k	Resistor
	RES_0603	R63	0603	1k	Resistor
	RES_0603	R64	0603	1k	Resistor
2	RES_0603	R111	0603	220E	Resistor
	RES_0603	R112	0603	220E	Resistor
1	RES_0603	R110	0603	220E	Resistor
1	RES_0603	R82	0603	130E	Resistor
2	RES_0805	R78	0805	0E	Resistor
	RES_0805	R79	0805	0E	Resistor
4	RES_0805	R83	0805	0E	Resistor
	RES_0805	R86	0805	0E	Resistor
	RES_0805	R96	0805	0E	Resistor
	RES_0805	R105	0805	0E	Resistor

1	RES_1206	R77	1206	51E	Resistor
1	RES_1206	R120	1206	120E	Resistor
4	RES_0603	R33	0603	0E	Resistor
	RES_0603	R34	0603	0E	Resistor
	RES_0603	R71	0603	0E	Resistor
	RES_0603	R72	0603	0E	Resistor
4	SMA_SMD_S	J1	SMA_SMD_S		SMA Connector
	SMA_SMD_S	J2	SMA_SMD_S		SMA Connector
	SMA_SMD_S	J3	SMA_SMD_S		SMA Connector
	SMA_SMD_S	J4	SMA_SMD_S		SMA Connector
1	SN74LVC2G17	U5	SC-70		Dual schmitt-trigger
4	THS4508	U1	QFN-16		Amplifier
	THS4508	U2	QFN-16		Amplifier
	THS4508	U3	QFN-16		Amplifier
	THS4508	U4	QFN-16		Amplifier
1	TPS79625	U30	SOT223-5		Low Dropout
1	USB_CONN_B	U19	USB_CONN_B		
1	VX3	Q2	VX3	66MHz	Oszillator VX3, 3.3V
1	XC3S400-TQ144	U20	TQFP144		FPGA
1	XCF02S-V020	U26	TSSOP20		EEPROM