Controlling the N2ADR full digital frontend (V1.1)

Mario Rößler, DH5YM

January 11, 2012

Abstract

This document describes the control of the N2ADR full digital RF frontend also known as HiQSDR. Also the frame format of the reception and transmission samples is described. The N2ADR frontend is an Direct Sampling (DDC) / Direct Synthesis (DUC) frontend for the shortwave range between DC and 60MHz. It uses an A/D converter coupled to the antenna via Balun and LNA for RX and D/A Converter for generating a TX signal.

1 Description of the Control strategy and interfaces

The frontend provides RF samples via UDP via Ethernet and is also controlled via UDP frames. The audio interface is the soundcard of the PC that runs the SDR software. RX/TX switching is done via the frontend itself. This is especially necessary for getting QSK capability for CW operation. Switching the frontend on needs some handshaking with frames that contain special bytes. This is described in the last section about setting up the receiver.

2 Ports used for communication

The rececceived samples are sent by the frontend via port 0xBC77 (dec 48247). Controlling the hardware is done via the next port 0xBC78 (dec 48248). Transmit samples are sent to the frontend via port 0xBC79 (dec 48249).

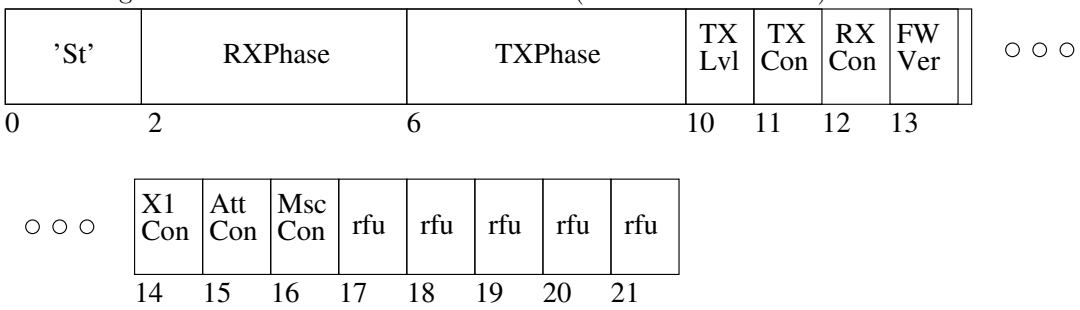
3 Receiver Control frame format

The receiver and general frequency setup is done via a 14Byte UDP frame for hardware version 1.0. From hardware version 1.1 22bytes are used. This frame needs to be send each time a setting of the frontend needs to be changed. The control word consists of the following bytes:

From the quisk_hardware.py for the HiQSDR hardware the following information about the control word can be obtained:

```
# want_udp_status is a 14-byte string with numbers in little-endian order:
#       [0:2]           'St'
#       [2:6]           Rx tune phase
#       [6:10]          Tx tune phase
#       [10]            Tx output level 0 to 255
#       [11]            Tx control bits:
#               0x01    Enable CW transmit
#               0x02    Enable all other transmit
#               0x04    Use the HiQSDR extended IO pins not present in
#                       the 2010 QEX ver 1.0
#               0x08    The key is down (software key)
#       [12]    Rx control bits
#                       Second stage decimation less one, 1-39, six bits
#       [13]    zero or firmware version number
# The above is used for firmware  version 1.0; add eight more
# bytes for version 1.1:
#       [14]    X1 connector:  Preselect pins 69, 68, 65, 64;
#                              Preamp pin 63, Tx LED pin 57
#       [15]    Attenuator pins 84, 83, 82, 81, 80
#       [16]    More bits: AntSwitch pin 41 is 0x01
#      Remaining five bytes are sent as zero.
# For version 1.2 include the VNA scan count:
#       [17] .. [21] Remaining five bytes are sent as zero.
```

Figure 1: N2ADR frontend control word (firmware version 1.1)



'St' = 0x5374 = fixed identification header
rfu = reserved for future usage, fill with Zero !

3.1 St identification [0,1]

The two characters S and t need to be send as the first two bytes inside the control word to identify the control software and avoid accidently changing the settings by random frames.

3.2 RX tune phase setting [2,3,4,5]

The RXPhase setting contains 4 bytes. These are calculated from the receiver frequency setting.

$$RXPhase = \frac{RXFrequency}{ReferenceClock} * 2^{32} + 0.5 \quad (1)$$

The bytes need to be aligned with least significant byte first. The clockrate is 122880000 Hz. The RX frequency results in the frequency of the signal that should be received and the mode that is selected.

3.3 TX tune phase setting [6,7,8,9]

The TXPhase setting is calculated according to the RX tuner phase setting.

3.4 TX output level setting [10]

The TX output level setting adjusts the output power of the TX DAC via the TX Level DAC. The TX output level is a 8 bit value that can be in the range of 0..255.

3.5 TX control bits [11]

The TX control bits setting is a 8 bit value that can currently have the values:
0x00 = not valid
0x01 = CW operation
0x02 = all other operation (e.g. SSB)
0x04 = use the HiQSDR extended IO pins (from FPGA version 1.1 on)
0x08 = simulate key down signal (software PTT)
CW operation means the TX sine signal is generated in the FPGA when the key input is keyed. The sidetone should be played at the PC. For hardware version 1.0 the PTT input is used for keying CW. In SSB mode the samples sent by the PC will be digitally modulated when the PTT is keyed. The hardware might use the additional signals available at header interfaces if the option 0x04 is used from FPGA version 1.1 on. From 1.1 on also a software PTT control is available for all modes except CW. This can be enabled by setting bit 0x08.

3.6 RX Control setting [12]

The RX control setting is a one byte value that controls the decimation setting of the frontend. All even decimation steps between 2 and 39 are allowed. Also decimation by one can be used but this is not tested. There may be problems with the FPGA and the limited ethernet bandwith with decimation 1. The decimation control byte is decimation setting reduced by one. This means if you want to set the decimation to two you need to send 2-1=1. The sample rates are calculated by:

$$Decimation = (RXControl + 1) * 8 * 8 \quad (2)$$

$$Samplerate = \frac{RXUDPClock}{Decimation} \quad (3)$$

This provides the following samplerates. Of cause only the rates that are multiple of 48kHz audio rate may make sense.

Decimation decimal	decimation hexadecimal	Samplerate
1	0x1	1920000
2	0x2	960000
4	0x4	480000
6	0x6	320000
8	0x8	240000
10	0xA	192000
12	0xC	160000
16	0xF	120000
20	0x14	96000
24	0x18	80000
30	0x1E	64000
32	0x20	60000
40	0x28	48000

3.7 FPGA firmware version number [13]

For hw version 1.0 this byte is left 0. From hardware version 1.1 this byte carries the FPGA firmware version number. The FPGA firmware version is a number > 0 from version 1.1 on.

3.8 X1 connector pin status [14]

The 8 bits of byte 14 represent the preselect pins 69, 68, 65, 64, the preamp pin 63 and the Tx LED pin 57. Remaining bits should be left 0. The pins are available at header X1 of the HiQSDR PCB. They can be used to control a external preselection filter according to a multiplexing matrix defined elswere, switch on and off a preamplifier and control a TX indication LED or a RX/TX relais. All signals are active-high.

Bit	Name	Location
0x01	Presel_1	P69/X1.1
0x02	Presel_2	P68/X1.2
0x04	Presel_4	P65/X1.3
0x08	Presel_8	P64/X1.4
0x10	Preamp-On	P63/X1.5
0x20	TX-LED	P57/X1.6, only set during TX, no control here !
0xC0	-	not used

3.9 Attenuator setting [15]

The byte 15 controls the attenuator pins 84, 83, 82, 81, 80. Other bits are left 0. The attenuator RF2420 (RF Micro Devices) is available at the HiQSDR PCB and can be used to reduce the input power level to the onboard LNA. The stepsize is 2dB in the range of 0..44dB plus 4dB insertion loss. The stages are Bypass, 2dB, 4dB, 8dB, 10dB, 20dB. Mapping of control byte 15:

Bit	attenuation stage	Location/Name
0x01	2dB	P84/ATT2dB
0x02	4dB	P83/ATT4dB
0x04	8dB	P82/ATT8dB
0x08	10dB	P81/ATT10dB
0x10	20dB	P80/ATT20dB
0xE0	-	not used

Please notice that a controlable preamplifier is not included on the frontend PCB. A external preamp can be controlled via X1.

3.10 Additional control setting [16]

Byte 16 contains additional bits for control functions. Currently only bit 0x01 is defined. It is used for switching between the to RX antenna inputs of the HiQSDR PCB. If the bit is unset antenna 1 is used. If the bit is set the antenna 2 is used. Other bits are unused and should be set to 0.

3.11 RFU bytes [17,18,19,20,21]

The bytes 17 to 21 are currently unused and RFU (reserved for future usage). These bytes need to be send as Zero.

4 Reception frame format

The received samples are carried by UDP frames with a total fixed payload size of 1442bytes. Each sample consists of 3 Bytes I and 3 Bytes Q data. They are interleaved. The first byte of each UDP frame carries a sequence number that is incremented from frame to frame. It can be used to identify a loss of UDP messages during reception. The second byte is used for signalling the current status of the reception. Bit 0 is used as Key indication. If it is 0 the PTT is issued and if it is 1 the PTT is not active. Bit 1 indicates clipping of the ADC (overrange). The data format of the samples is little endian (least significant byte first) with real words at the odd (first) positions and the imaginary words at the even (second) positions.

5 Transmission frame format

Audio samples are sent to the transmitter as two zero bytes followed by 300 sample frames. Each sample frame is a 16-bit I sample followed by a 16-bit Q sample in little-endian order. The sample rate is fixed at 48000 samples per second. This mechanism is not used for CW, as the FPGA generates its own shaped CW signal in response to a high level on the Key pin.

6 Derivatives

There may be FPGA designs changed by other users. For example DL2STG changed the design to support his HiQScope application. These alternative

designs may require different configuration and have different frame format. They are currently not described in this document.

7 Example for setting up the receiver

7.1 Setting the sample destination address

To register your PCś IP address at the frontend for receiving the samples it is necessary to send a UDP frame with the content of 0x72, 0x72 (two times 0x72). When the frontend receives this message it will afterwards automatically send all UDP messages with received samples to this destination. Sending 0x73, 0x73 will cause the hardware to stop sending samples. In both cases the port used is the same as the Rx sample port.

The IP and Ethernet (MAC) address of the hardware is compiled into the FPGA program. The IP address should be chosen to be valid on the network. The Ethernet address must be unique on the network. The default Ethernet address will work unless there are two frontend on the same network.

7.2 Configuration of the transceiver

One example how to setup the transceiver (FPGA version 1.0):

Control word hexadecimal 0x 5374—ABAAAA07—52B8A807—78—02—27—00
St RXPhase TXPhase Level Mode RXC RFU
0x 07 AA AA AB means 3679.995kHz RX frequency
0x 07 A8 B8 52 means 3676.350kHz TX frequency
0x 78 means output drive is 120
0x 02 means mode is set to non-CW (e.g.SSB)
0x 27 means decimation is 40*8*8 which results in a samplerate of 48kSpls
0x 00 is RFU

For further versions set the other bytes according to the control word description. e.g. X1 settings, attenuator settings, additional settings and 5 tailing 0x00 for FW version 1.1

7.3 Receiving data

tbd

7.4 Transmitting data

tbd

7.5 Document changelist

added RF attenuation setting details
added X1 control details