

Ansteuerung des N2ADR/HiQSDR DDC/DUC Frontends (FW V1.1)

Mario Rößler, DH5YM

9. Februar 2012

Zusammenfassung

Dieses Dokument beschreibt die Ansteuerung des N2ADR DDC/DUC HF Frontend, auch bekannt als HiQSDR. Die Frame Formate der Samples für Empfang und Sendung, sowie die Ansteuerung werden beschrieben. Das N2ADR Frontend ist ein Direkt-Sampler (DDC, direct digital conversion) / Direkt-Synthesizer (DUC, direct up-conversion) für den Kurzwellenbereich zwischen nahezu Gleichspannung und 66MHz. Dafür wird ein A/D Wandler benutzt, der via LNA und Balun direkt mit der Antenne verbunden ist. Sendeseitig wird das Signal mit einem D/A Wandler erzeugt.

Inhaltsverzeichnis

1	Beschreibung der Ansteuer Strategie und der Schnittstellen	3
2	Ports, die für die Kommunikation benutzt werden	3
3	Ansteuerung des Empfängers	3
3.1	St Identifikation [0,1]	4
3.2	RX tune phase Einstellung [2,3,4,5]	4
3.3	TX tune phase Einstellung [6,7,8,9]	4
3.4	TX Ausgangspegel Einstellung [10]	4
3.5	TX Steuer Bits [11]	5
3.6	RX Control Einstellung [12]	5
3.7	FPGA Firmware Versionsnummer [13]	6
3.8	X1 Anschluss Pin Status [14]	6
3.9	Dämpfungs Einstellung [15]	6
3.10	Zusätzliche Steuerungseinstellungen [16]	7
3.11	RFU Bytes [17,18,19,20,21]	7
4	Empfangs Paketformat	7
5	Sende Paketformat	7
6	Beispiel für das Starten des Empfängers	8
6.1	Setzen der Zieladresse für die Samples	8
6.2	Konfiguration des Transceivers	8
6.2.1	FPGA FW Version 1.0 Steuerwort	8
6.2.2	FPGA FW Version 1.1 Steuerwort	8
6.3	Startsequenz	9
6.4	Empfang von Daten	9
6.5	Senden von Daten	9
7	Hardware Varianten	9
8	Dokument Änderungsliste	9
8.1	20120111	10

1 Beschreibung der Ansteuer Strategie und der Schnittstellen

Das Frontend stellt HF Samples über UDP über Ethernet zur Verfügung und wird ebenso über UDP Frames gesteuert. Als Audio Schnittstelle zum Benutzer fungiert die Soundkarte des PC, auf dem die SDR Software läuft. Die Steuerung der Sende/Empfangsumschaltung wird normalerweise durch das Frontend selbst durchgeführt. Dafür sind separate Eingänge für PTT und CW Taste vorhanden. Damit kann QSK Fähigkeit erreicht werden. Um das Frontend einzuschalten, wird etwas Austausch von Frames zwischen Software und Hardware benötigt. Die Abfolge wird im Abschnitt über das Aufsetzen des Empfängers näher beschrieben.

2 Ports, die für die Kommunikation benutzt werden

Die empfangenen Samples werden vom Frontend über Port 0xBC77 (dezimal 48247) gesendet. Die Steuerung erfolgt via den nächsten Port 0xBC78 (dezimal 48248). Die zu sendenden Samples werden über Port 0xBC79 (dezimal 48249) übertragen.

3 Ansteuerung des Empfängers

Die Ansteuerung des Empfängers und die generelle Steuerung der eingestellten Frequenzen und Modes erfolgt über ein 14Byte langes UDP Frame für Hardware Version 1.0. Ab Version 1.1 wird ein 22Byte langes UDP Frame benutzt. Dieses Frame muss jedes Mal gesendet werden, wenn eine Einstellung des Frontend geändert werden soll. Das Steuerwort besteht aus den folgenden Bytes:

Aus der Datei `quisk hardware.py` für die HiQSDR Hardware kann folgende Information entnommen werden:

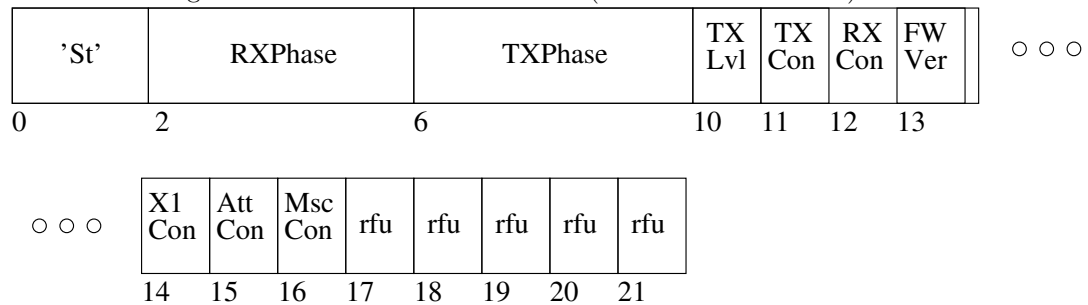
```
# want_udp_status is a 14-byte string with numbers in little-endian order:
#   [0:2]           'St'
#   [2:6]           Rx tune phase
#   [6:10]          Tx tune phase
#   [10]            Tx output level 0 to 255
#   [11]            Tx control bits:
#                   0x01   Enable CW transmit
#                   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.

```

Abbildung 1: N2ADR Frontend Steuerwort (Firmware Version 1.1)



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

3.1 St Identifikation [0,1]

Um die SDR Software beim Frontend anzumelden und versehentliches Ändern der Einstellungen zu vermeiden, müssen die beiden Buchstaben S und t als erste beide Zeichen innerhalb des Steuerwortes stehen.

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

Der Parameter RXPhase enthält 4 Bytes. Diese werden aus der gewünschten Empfangsfrequenz wie folgt berechnet:

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

Diese Bytes müssen mit dem niederwertigsten Byte zuerst angeordnet werden. Die Referenz Taktrate ist 122880000 Hz. Die Empfangsfrequenz ergibt sich aus der Frequenz des zu empfangenen Signals und der ausgewählten Betriebsart.

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

Die TXPhase Einstellung wird analog zur Einstellung des RX berechnet.

3.4 TX Ausgangspegel Einstellung [10]

Die Einstellung des Ausgangspegels beeinflusst die Ausgangsleistung des TX DAC über den TX Pegel DAC. Der Wert des TX Ausgangspegel ist ein 8bit Wert im Bereich von 0 bis 255.

3.5 TX Steuer Bits [11]

Die TX Steuer Bits einstellung ist ein 8Bit Wert, der im Moment folgende Inhalte haben kann:

0x00 = nicht gültig

0x01 = CW Betrieb

0x02 = alle anderen Betriebsarten, IQ Modulator (zum Beispiel SSB)

0x04 = Benutzung der HiQSDR erweiterten IO Pins (ab FPGA Version 1.1)

0x08 = Simulation des FPGA Key Down Signals (Software PTT)

CW Betrieb bedeutet, dass das TX Signal direkt durch den FPGA erzeugt wird, wenn der Tasteneingang betätigt wird. Der Mithörton muss dann am PC abgespielt werden. Für Hardware Version 1.0 wird der PTT Eingang für die Tastung von CW mitbenutzt. In SSB werden die Baseband Samples vom PC geschickt und digital moduliert, wenn die PTT betätigt wird. Der Transceiver kann zusätzliche Signale benutzen, die an Headern (zum Beispiel beim HiQSDR) verfügbar sind, wenn Option 0x04 benutzt wird. Diese ist ab FPGA Version 1.1 verfügbar. Ebenfalls ab 1.1 gibt es die Möglichkeit die PTT für alle Modes ausser CW aus der Software zu steuern. Das kann durch das Setzen von bit 0x08 aktiviert werden.

3.6 RX Control Einstellung [12]

Die RX Control Einstellung ist ein einzelnes Byte, das die Dezimation im Frontend steuert. Alle geraden Dezimationsschritte zwischen 2 und 39 sind erlaubt. Auch die Einstellung Dezimation = 1 kann benutzt werden, wurde aber nicht getestet. Dabei könnte es Probleme mit den Lautzeiten im FPGA und der begrenzten Bandbreite des Ethernet geben. Das Control Byte enthält die Einstellung der Dezimationsstufe verringert um Eins. Das bedeutet, wenn die Dezimation auf 2 gestellt werden soll muss 2-1=1 gesendet werden. Die Sample Raten werden wie folgt berechnet:

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

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

Das ergibt die nachfolgend genannten Sampleraten. Natuerlich machen nur Sampleraten Sinn, die ein Vielfaches von 48kHz sind, da das die Abspielrate des demodulierten Audio ist.

Dezimation dezimal	dezimation hexadezimal	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 Versionsnummer [13]

Für HW Version 1.0 ist dieses Byte 0. Ab Hardware Version 1.1 enthält dieses Byte die Firmware Versionsnummer (nur die Nachkommastelle). Die FPGA Versionsnummer ist eine Zahl > 0 ab Version 1.1.

3.8 X1 Anschluss Pin Status [14]

Die 8 Bit von Byte 14 repräsentieren die Preselektor Pins 69,68,65,64 und das Vorverstärker Pin 63, sowie das TX LED Pin 57. Die übrigen Bits müssen auf 0 belassen werden. Diese Pins sind an Header X1 der HiQSDR Platine verfügbar. Diese können benutzt werden, um ein externes Preselect Filter entsprechend einer Multiplex Matrix einzustellen, die in der Software definiert wird und der verwendeten Hardware entspricht. Damit können ein externer Vorverstärker, die TX LED und auch eine Sendeempfangsumschaltung gesteuert werden. Alle Signale sind High-Aktiv.

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, Nur aktiv während TX , Hier keine Steuerung !
0xC0	-	unbenutzt

3.9 Dämpfung Einstellung [15]

Byte 15 steuert die Pins für das Dämpfungsglied an den Pins 84,83,82,81,80. Die übrigen Pins bleiben 0. Das Dämpfungsglied RF2420 (RF Micro Devices) ist in die HiQSDR Platine integriert und kann benutzt werden, um den Eingangspegel in den integrierten LNA vor dem ADC zu verringern. Die Schrittweite ist 2dB im Bereich von 0..44dB + 4dB Einfügedämpfung. Die Stufen sind Überbrückung, 2dB, 4dB, 8dB, 10dB, 20dB. Die Verteilung auf Byte 15 ist wie folgt:

Bit	Dämpfungsstufe	Pin/Name
0x01	2dB	P84/ATT2dB
0x02	4dB	P83/ATT4dB
0x04	8dB	P82/ATT8dB
0x08	10dB	P81/ATT10dB
0x10	20dB	P80/ATT20dB
0xE0	-	unbenutzt

Bitte beachten, dass sich der steuerbare Vorverstärker nicht auf der Frontend Platine befindet. Ein externer Verstärker kann über X1 gesteuert werden.

3.10 Zusätzliche Steuerungseinstellungen [16]

Byte 16 enthält zusätzliche Bits für Steuerfunktionen. Im Moment ist nur Bit 0x01 definiert. Es wird benutzt, um die Antenneneingänge auf der HiQSDR Platine umzuschalten. Wenn das Bit gesetzt ist wird Antenne 1 benutzt. Wenn es gelöscht ist wird Antenne 2 benutzt. Die übrigen Bits müssen 0 gesetzt werden.

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

Bytes 17 bis 21 sind im Moment unbenutzt und RFU (reserved for future usage). Diese Bytes müssen auf 0 gesetzt werden.

4 Empfangs Paketformat

Die empfangenen Samples werden mittels UDP Paketen vom Frontend geschickt. Diese Frames haben eine feste Inhaltlänge von 1442 Byte. Jedes Sample besteht aus 3 Bytes I und 3 Bytes Q Daten. Diese sind jeweils abwechselnd angeordnet. Das erste Byte jedes UDP Paketes enthält eine Sequenznummer, die mit jedem Paket um 1 erhöht wird. Diese kann benutzt werden, um den Verlust von UDP Paketen während des Empfangs zu erkennen. Das zweite Byte wird benutzt, um den aktuellen Zustand des Frontends zu übertragen. Bit 0 wird als PTT Indikator benutzt. Wenn es auf 0 gesetzt ist wurde die PTT ausgelöst, wenn es 1 ist wurde die PTT nicht aktiviert. Bit 1 signalisiert eine Übersteuerung des ADC (Clipping). Das Datenformat ist Little Endian (niederwertigstes Byte zuerst) mit dem realwertigen Teil an der ungeraden Stelle (zuerst) und den komplexwertigen Worten an den gerade (zweiten) Stellen.

5 Sende Paketformat

Audio Samples werden in einem Paket bestehend aus 2 führenden Nullbytes gefolgt von 300 Samples zum Frontend gesendet. Jedes Sample ist ein 16 Bit I Wert gefolgt von einem 16 Bit Q Wert in Little Endian Anordnung. Die Sample Rate ist Fix 48000 Samples pro Sekunde. Dieser Mechanismus wird nicht für CW benutzt. Hier generiert das FPGA selbst die CW Signale anhand der Tastung des Key Pins.

6 Beispiel für das Starten des Empfängers

6.1 Setzen der Zieladresse für die Samples

Um die IP Adresse des PCs für die weitere Verarbeitung der Samples am Frontend zu setzen ist es notwendig, ein UDP Paket mit dem Inhalt 0x72,0x72 (2 mal 0x72) zu senden. Wenn das Frontend diese Nachricht empfängt wird es nachfolgend alle UDP Pakete mit empfangenen Samples an die IP Adresse schicken, die das Initialisierungspaket geschickt hat. Wenn 0x73,0x73 gesendet wird, so wird die Hardware die Übertragung der Samples stoppen. In beiden Fällen wird als Zielport der RX Sample Port benutzt.

Die IP und Ethernet (MAC) Adresse der Hardware wird in das FPGA Programm eincompiliert. Die IP Adresse sollte entsprechend dem eigenen Netzwerk gewählt werden. Die Ethernet Adresse muss im erreichbaren Netzwerk eindeutig sein. Die Standard Ethernetadresse sollte funktionieren, solange nicht 2 Frontends gleichzeitig im Netzwerk aktiv sind.

6.2 Konfiguration des Transceivers

6.2.1 FPGA FW Version 1.0 Steuerwort

Ein Beispiel, wie der Transceiver gestartet wird (FPGA Version 1.0):

Steuerwort Hexadezimal 0x 5374—ABAAAA07—52B8A807—78—02—27—00
St RXPhase TXPhase Level Mode RXC RFU
0x 07 AA AA AB bedeutet 3679.995kHz RX Frequenz
0x 07 A8 B8 52 bedeutet 3676.350kHz TX Frequenz
0x 78 bedeutet Ausgangspegel 120
0x 02 bedeutet TX Mode ist auf nicht-CW gestellt (z.B. SSB)
0x 27 bedeutet Dezimation ist $40 \cdot 8 \cdot 8$, was eine Samplerate von 48kSample/s ergibt
0x 00 RFU

Für andere Versionen müssen die weiteren Bytes entsprechend der Beschreibung des Steuerwortes gesetzt werden. Zum Beispiel sind das die X1 Einstellungen, die Einstellung des Dämpfungsgliedes, weitere Konfigurationen und 5 0x00 am Ende für FW Version 1.1.

6.2.2 FPGA FW Version 1.1 Steuerwort

Die folgenden Zeilen beschreiben ein Beispiel für das Steuerwort in Zusammenhang mit der FPGA FW Version 1.1

Hexadezimal 0x 5374—5555e503—bc05e303—78—0E—27—01—10—08—01—000000000
St RXPhase TXPhase Level Mode RXC RFU
0x 5374 ist die Kennung des Steuerwortes
0x 55 55 e5 03 ist die Empfangsfrequenz nach vorher beschriebener Berechnungsvorschrift
0x bc 05 e3 03 ist die Sendefrequenz nach vorher beschriebener Berechnungsvorschrift
0x 78 bedeutet der relative Ausgangspegel ist 120 von 255
0x 0E bedeutet der TX Mode ist auf nicht-CW gesetzt. Die Steuerung der erweiterten IO Pins ist aktiviert und die Software PTT ist eingeschaltet (Frontend

ist auf TX geschaltet)

0x 27 bedeutet die Dezimation ist $40 \cdot 8 \cdot 8$, was eine Samplerate von 48kSample/s ergibt

0x 01 bedeutet die Firmware hat Version 1.1

0x 10 bedeutet der LNA Steuerpin ist auf High gesetzt. Der Preselektor Pin ist auf 0 gesetzt. Die TX-LED wird über den Zustand des Senders gesteuert und kann hier nicht gesetzt werden.

0x 08 bedeutet das Dämpfungsglied ist auf -10dB gesetzt

0x 01 bedeutet Antenneneingang 2 ist gewählt

0x 000000000 sind RFU Bytes

Jedes Steuerwort, welches zum Frontend gesendet wird, wird mit einem Paket beantwortet, das den Status des Frontend im gleichen Format enthält.

6.3 Startsequenz

1. Senden von 0x72 0x72, um sich selbst beim Frontend zu registrieren
2. Senden des Steuerwortes entsprechend der Firmware Version 1.0 (für Rückwärtskompatibilität)
3. Auswertung der Antwort, um die aktuelle Firmware Version zu bestimmen
4. Senden eines Steuerwortes entsprechend der Firmware Version, die detektiert wurde
5. Senden von 0x72 0x72, um den Start der Übertragung von Samples zu veranlassen

6.4 Empfang von Daten

tbd

6.5 Senden von Daten

tbd

7 Hardware Varianten

Es kann verschiedene Ausprägungen der FPGA Hardware von unterschiedlichen Nutzern geben. Zum Beispiel hat DL2STG das Design verändert, um seine HiQScope Funktionalität zu implementieren. Diese alternativen Designs können veränderte Ansteuerung oder unterschiedliches Frame Format haben. Diese sind aktuell nicht in diesem Dokument beschrieben.

8 Dokument Änderungsliste

Deutsche Übersetzung

FPGA FW 1.1 Beispiel hinzugefügt

Startup Sequenz hinzugefügt

Inhaltsverzeichnis hinzugefügt

8.1 20120111

Details über die HF Dämpfungsglied Einstellung hinzugefügt
X1 Steuerung Details hinzugefügt