# uTrak APRS-Implementierung

Da die Aussendung von APRS schon ein eher komplexes Thema mit der minimalistischen Hardware ist, soll die Implementierung hier dokumentiert werden.

## Beschreibung

APRS wird im Amateurfunk in Form von Packet Radio ausgesandt. Ein AFSK-Signal wird frequenzmoduliert und auf regional koordinierten Frequenzen ausgesendet. Um die Abdeckung der Verfolgbarkeit des Trackers zu erhöhen, sollen Positionsaussendungen im APRS-Format ausgesendet werden. Da der Si4060 eigentlich keine frequenzmodulierten Aussendungen unterstützt, wird die FM in Software erledigt. Die Implementierung folgt ansonsten der AX.25-Spezifikation. Konkret werden folgende Parameter der Aussendung implementiert:

- Physical Layer
  - Mark-Frequenz: 1200 Hz
  - Space-Frequenz: 2200 Hz
  - Übertragungsrate: 1200 Baud
  - NRZI encoding
  - Bit-Stuffing (nach jeder fünften „1“ eine „0“ einfügen)
- Data-Link Layer
  - AX.25-Framing (HDLC)
  - Frame Check Sequence nach CRC16-CCITT

## Implementierung

### AFSK-FM

Im MSP430 wird ein NCO implementiert, der mit möglichst geringem Fehler die nötigen Frequenzen erzeugt. Die Samplerate wird dabei aus dem vorhandenen Grundtakt so gewählt, dass der Fehler für Baudrate, Mark- und Spacefrequenz möglichst klein ist. Um dieses Optimum zu finden, wurde ein Matlab-Skript geschrieben, was nach Brute-Force-Methode den prozentualen Fehler für alle Einstellungen herausfindet (msp_fm.m). Ein Timer stellt die Zeitbasis für den NCO zur Verfügung, dessen Samplerate so gewählt werden sollte, dass ein üblicher FM-Demodulator die Samplefrequenz schon nicht mehr in den NF-Zweig durchlässt. Ein Wert um die 10kHz sollte dafür ausreichen. Der Phasenakkumulator wird bei jedem Takt auf Grundlage des aktuell zu übertragenden Tons erhöht (FCW_MARK bzw. FCW_SPACE). Aus einer Sinus-Tabelle, erstellt mit einem kleinen Skript, wird der nächste Sinus-Wert ausgelesen und daraufhin der Si4060 auf den entsprechenden Kanal gewechselt.

### Bit-Stream

Aus einem ganzzahligen Teil des Sampletaktes wird der Bit-Takt erzeugt, mit welchem die Funktion zum holen des nächsten Bits ausgeführt wird. Es wird sich einerseits um die NRZI-Kodierung gekümmert, als auch um die Vorgabe des AX.25-Protokolls, Bit-Stuffing zu betreiben. Nach der fünften

„1“ wird automatisch eine „0“, also ein Flankenwechsel eingefügt und das nächste Nutzdatenbit um eine Bitlänge verzögert.

### AX.25-Protokoll

Von der Software wird lediglich ein Puffer beschrieben, in welchem die Nutzdaten liegen. Das zugehörige Längenfeld muss mitbeschrieben werden. Die Sende-State-Machine kümmert sich um Aussendung der Flag-Sequenzen am Anfang und am Ende, der Header-Informationen, der Nutzdaten sowie der FCS (Frame Check Sequence).

## Links

- AX.25-Spezifikation, Version 2.2

From:
https://loetlabor-jena.de/ - **Lötlabor Jena**

Permanent link:
**https://loetlabor-jena.de/doku.php?id=projekte:utrak:aprs&rev=1413887378**

Last update: **2014/10/21 10:29**