# SDR DAB Signal Synchronisation Bericht

## Einleitung

Im Rahmen des Moduls DSVB ist ein Versuch zu Software Defined Radio (SDR) durchgeführt worden. Der Unterschied zum normalen Empfänger liegt in der Signalverarbeitung. Der Hauptteil wird nicht auf der Hardwareseite, sondern auf der Softwareseite ausgeführt. Als Transceiver wird ein NESDR Smart verwendet. Damit kann ein Radiosignal einer gewünschten Frequenz umgewandelt und über USB ausgelesen werden. In dieser Übung wurde ein Teil des FM Radio Signals untersucht, den Digital Audio Broadcast(DAB).

## DAB Broadcast

Ein DAB Frame dauert 96 ms. Jeder Frame besitzt 77 Orthogonal Frequency Division Multiplex (OFDM) Symbole. Das Phase Ref Symbol wird für die Synchronisierung der Frames genutzt. Alle Symbole im Frame verwenden OFDM. Das bedeutet dass alle Trägersignale spektral gesehen dort eine Nullstelle haben wo die anderen Trägersignale Informationen senden.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Number of Samples in OFDM Main Part NU = TU \* FS = 1792

Number of samples in cyclic prefix Ncp = Tcp \* Fs =

## Frame Synchronisation

Für die Synchronisation wird das Phase Ref Symbol genutzt. Die Koeffizienten des Frequenzspektrums sind vorhanden. Mit einer Inverse Discrete Fourier Transform (IDFT) kann aus diesen Koeffizienten das Phase Ref Symbol in den Zeitbereich abgebildet werden. Wenn dieses mit dem Eingangssignal Korreliert wird erhält man ein Signal mit Peaks. Diese zeigen an welchen Stellen beim Eingangssignal das Phase Ref Symbol gesendet wird.

Es senden mehrere Sendestationen die sich unterschiedlich weit entfernt befinden. Deshalb besitzt das Empfangssignal mehrere überlagerte Frames die Zeitverzögert sind. Das Ziel ist es, das Signal mit der grössten Leistung auszuwerten. Deshalb soll das Phase Ref Symbol vor dem höchsten Peak angeschaut werden.

## Filter Entwurf

Das Ziel des Filters ist es, das Phase Ref Symbol am Eingang anzulegen, und dann das Empfangene OFDM Symbol vor dem höchsten Peak zu bekommen. Somit entfernt der Filter die Signale aller Sendestationen, ausser jener die am wenigsten weit entfernt ist. Somit kann der entworfene Filter für die Signal Rekonstruktion verwendet werden.

Als Ausgangssignal des Systems wird das OFDM Symbol vor einem Peak genommen. Dies entspricht einem Vektor mit 1792 Werten. Dieser wird Fourier transformiert, weil die Impulsantwort des Systems im Frequenzbereich über eine einfache Division gerechnet werden kann. Der Eingang des Systems ist das Phase Ref Symbol. Die Impulsantwort muss dann noch mit einer IDFT Funktion in den Zeitbereich transformiert werden.

## Vorgehen

Um die Vorgaben erreichen zu können wird folgendermassen vorgegangen.

1. Parameter werden definiert
2. Koeffizienten Vektor erstellen und in den Zeitbereich transformieren
3. Frame synchronisieren. Dazu gehört die Peaks zu lokalisieren. Um die Peaks zu erhalten muss das Signal mit dem Koeffizienten Vektor korreliert werden.
4. Kanal Einschätzung. Dazu wird die Frequenzantwort und die Impulsantwort berechnet.
5. Analyse der Ergebnissen

Die Implementation wurde mit Matlab durchgeführt.

### Parameter definieren

fc = 227.36e6; % center frequency [Hz]

rs = 1.792e6; % sampling rate [Hz]

framelen = 256\*20; % frame length [samples]

nframes = 1000; % number of frames to be received

Tcp = 246\*10^-6; % length of cycle prefix [s]

Tu = 1000\*10^-6; % length of one symbol [s]

N = rs\*Tu; % number of Sampels for baseband (N=1792)

### Koeffizienten Vektor erstellen

In Abbildung 1 ist zu sehen, wie das ursprüngliche Signal, welches bei 227 MHz empfangen wurde ins Basisband um 0 Hz verschoben wurde. Wie bereits berechnet ist das N = 1792 Samples. Der Koeffizienten Vektor A\_cof ist aber nur 1537 Werte Lang. Es wird auf beiden Seiten noch mit der entsprechenden Anzahl von Nullen aufgefüllt. Die Ränder des Basisbandes werden nicht verwendet um eine Sicherheitsmarge gegenüber den Nachbarbändern zu erhalten.



Abbildung 1: Basisband

Nachdem der Vektor A\_cof auf beiden Seiten mit Nullen aufgefüllt wurde, damit dieser die gleiche Länge wie N hat, kann dieser nun analysiert werden. Dies geschieht im Frequenzbereich.

Abbildung 2: Übertragungsfunktion von A

Abbildung 3: Pol/Nullstellen Plan von A

Auf der Abbildung 2 ist zu sehen, dass auf……. In der Abbildung 3 zeigt sich die Übertragungsfunktion. Es werden alle Frequenzen, mit Ausnahme der Frequenzen, wo die Koeffizienten 0 zugefügt wurden durchgelassen. Zudem wird DC nicht übertragen.

### Peaks Lokalisieren

Der Koeffizienten Vektor a (im Zeitbereich) kann nun mit dem empfangenen Signal im Vektor r korreliert werden. Das Ergebnis ist in der sichtbar.



Abbildung 4: Korrelation von Vektor a und r

Die einzelnen Peaks bestehen aus mehreren Werten. Es wurde deshalb ein Fenster definiert, welches die Länge von der Signallänge/(Anzahl Peaks) hat. Danach wurde vom obigen Signal nur immer ein Teil mit der Fensterlänge angeschaut. So wurde sichergestellt, dass nur ein Peak angeschaut. Von diesem Bereich konnte nun der maximale Wert genommen werden. Dies wurde für alle Peaks wiederholt. Das Ergebnis war ein Vektor, wo sämtliche Positionen der Peaks enthalten sind. Um das Resultat zu überprüfen, wurde der Vektor in den gleichen Plot wie oben eingezeichnet. Das Ergebnis ist in Abbildung 5 ersichtlich.

Abbildung 5: Korrelation mit Peaks

Es konnten so sämtliche Peaks im Signal detektiert werden.

### Kanaleinschätzung

Nun sind die Positionen der Peaks bekannt. Es wurde nun der höchste Peak im Signal verwendet. Um die Position im Signal zu finden, wurde die Position des Peaks verwendet und mit der Länge von N subtrahiert. Nun konnten die ersten 1792 Werte ab dieser Position extrahiert werden. Die Übertragungsfunktion des Systems ist gegeben durch:

Y(w) ist der Vektor, der gerade gebildet wurde und der zweite Term in der Formel entspricht dem konjugiert-komplexen Wert von dem Koeffizienten Vektor A.

Abbildung 7: Impulsantwort

Abbildung 6: Frequenzantwort

Die beiden Abbildungen zeigen die Frequenz- bzw. die Impulsantwort, synchronisiert auf die Stelle des höchsten Peaks.

### Analyse der Ergebnisse