# DSP-Based GFSK Demodulator for AAU Ground Station

Group 627

June 18, 2014


Jacob Velling Dueholm
Jacob Møller Hjerrild Hansen
Miklas Strøm Kristoffersen
Søren Bøgeskov Nørgaard
14gr627@es.aau.dk

Department of Electronic Systems
Aalborg University
Denmark


AALBORG UNIVERSITY
DENMARK

# Agenda

Modulation Scheme

Spacelink Format

Blackfin Architecture

Program Flow

System Block Division

- Sampler
- Channel Filter
- Packet Detection
- Mid-Frequency Estimation
- Time Synchronization
- Symbol Decision

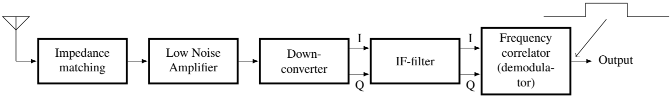
Acceptance Test

Conclusion

Demonstration

# Motivation

- Make something needed
- Satlab – Improve receiver for ground station



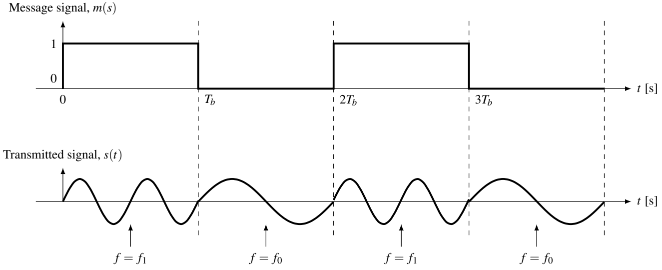

- Better reception
- More Flexible
- Soft-bit decision






Dept. of Electronic Systems
Aalborg University
Denmark

# Modulation Scheme – FSK




- FSK – Frequency Shift Keying

$$s(t) = \sqrt{\frac{2E_b}{T_b}} \cos\left[2\pi f_c t + \phi(t)\right] \tag{1}$$

$$\phi(t) = \phi(0) \pm h\pi t / T_b \tag{2}$$

$$\Delta f = |f_1 - f_0| = \frac{h}{T_b} = \frac{1}{\frac{1}{9600}} = 9600 \text{ Hz} \tag{3}$$

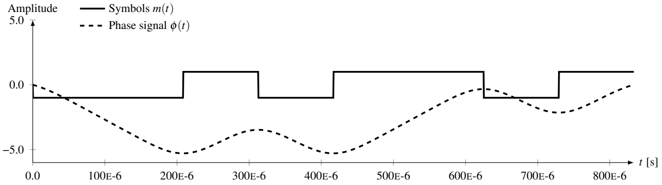






Dept. of Electronic Systems
Aalborg University
Denmark

# Modulation Scheme – GFSK



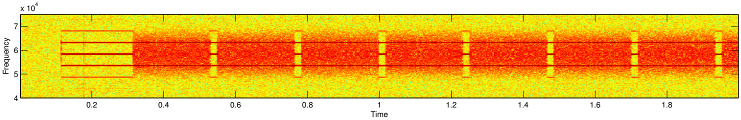

- $\phi(t) = \phi(0) + \frac{h\pi}{T_b} \int_0^{T_b} m_{\text{NRZ}}(t)dt$
- Basis for demodulating by phase

# Spacelink Format

| Training 240 B | Sync Word 6 B | FSM 1 B | Length 2 B | CSP Header 4 B | Data 23/84 B | HMAC 2 B |
|---|---|---|---|---|---|---|

Preamble | FEC Data

- Recording from AAUSAT III




- Signal generator
- Short package $= 8 \cdot \left(24B + 6B + 1B + 128B\right) = 1272$ Symbols

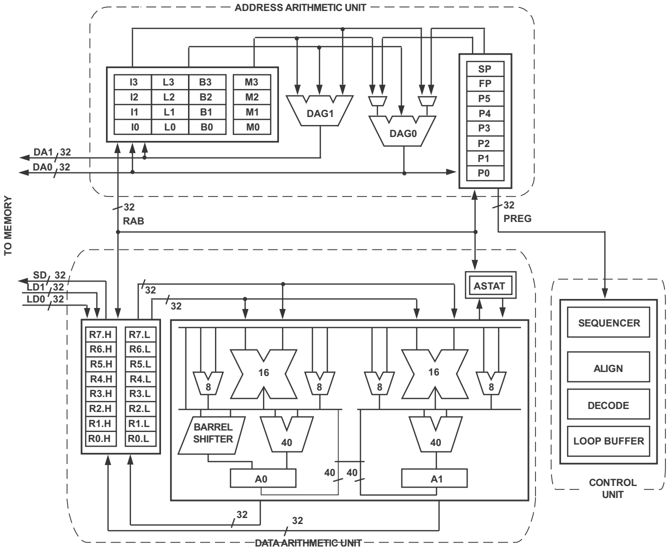




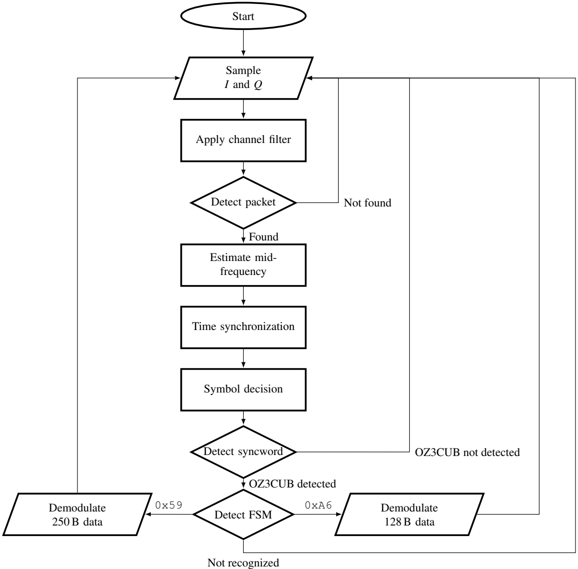



Dept. of Electronic Systems
Aalborg University
Denmark

# Blackfin Architecture

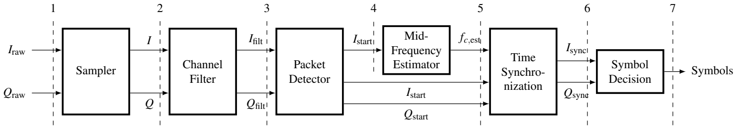


# Division Into Sub Modules – Program Flow



# System Block Division

The modules which defines the defines the demodulator.







Dept. of Electronic Systems
Aalborg University
Denmark

# Sampler

## Sorting Data Input

Sorting the input data.

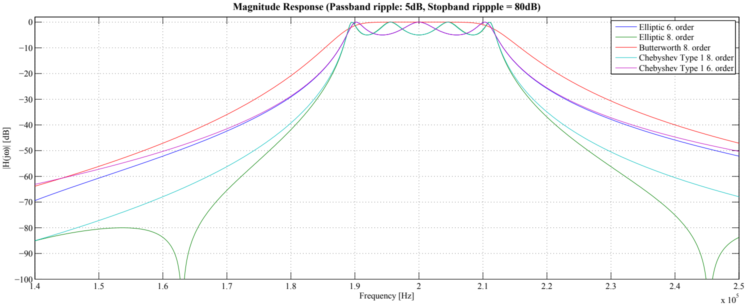
| 16 | 4 | 12 | 16 | 4 | 12 | bits |
|---|---|---|---|---|---|---|
| 0 | 0 | $I$ | 0 | 0 | $Q$ | $\cdots$ |

2 MiB

```
uint32_t *buf32 = (uint32_t *)buffer;
fract16  *buf16I;
fract16  *buf16Q;

while (n < SAMBLOCK/2) {
    buf16I[i] = (fract16)((buf32[n] << 4)&0xFFFF); n++;
    buf16Q[i] = (fract16)((buf32[n] << 4)&0xFFFF); n++;
    i++;
}
```

- Implementation appropriate inside the channel filter.




- Bandwidth is defined from Doppler range.
- Frequency characteristic is chosen from existing filter
- Group Delay of Butterworth is constant.

- 4th order LP $\rightarrow$ 8th order BP.

$$H(s) = \left. \frac{1}{s^4 + 2.61s^3 + 3.41s^2 + 2.61s + 1} \right|_{s=\frac{S^2+\Omega_0^2}{BS}}$$

$$H(z) = \left. \frac{B^4 S^4}{aS^8 + bS^7 + cS^6 + dS^5 + eS^4 + fS^3 + gS^2 + hS + i} \right|_{S=\frac{2}{T_S}\cdot\frac{z-1}{z+1}}$$

$$\begin{aligned}
\mathrm{a} &= 1 \\
\mathrm{b} &= 2.61B \\
\mathrm{c} &= 3.41B^2 + 4\Omega_0^2 \\
\mathrm{d} &= 2.61B(B^2 + 3\Omega_0^2) \\
\mathrm{e} &= B^4 + 6.82B^2\Omega_0^2 + 6\Omega_0^4 \\
\mathrm{f} &= 2.61B(B^2 + 3\Omega_0^2)\Omega_0^2 \\
\mathrm{g} &= 3.41B^2\Omega_0^4 + 4\Omega_0^6 \\
\mathrm{h} &= 2.61B\Omega_0^6 \\
\mathrm{i} &= \Omega_0^8
\end{aligned}$$




Dept. of Electronic Systems
Aalborg University
Denmark

# Channel Filter

## Implementation

$$H(z) = 67.263\text{E-}6 \cdot \frac{1-4z^{-2}+6z^{-4}-4z^{-6}+z^{-8}}{1+0.65z^{-1}+3.65z^{-2}+1.73z^{-3}+4.89z^{-4}+1.52z^{-5}+2.84z^{-6}+0.44z^{-7}+0.60z^{-8}}$$

- 2nd order cascades by pairing poles closest to unit circle with the closest zero.

$$H(z) = k \cdot \frac{(z-z_1)(z-z_2)}{(z-p_n)(z-p_n^*)} = k \cdot \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{a_0 + a_1 z^{-1} + a_2 z^{-2}}$$

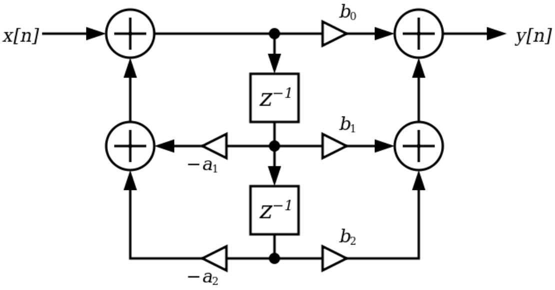
- Discrete Time Domain

$$y(n) = b_0 x(n) + b_1 x(n-1) + b_2 x(n-2) - a_1 y(n-1) - a_2 y(n-2)$$



Dept. of Electronic Systems
Aalborg University
Denmark

## Direct Form II




$$w(n) = x(n) - a_1 w(n-1) - a_2 w(n-2)$$
$$y(n) = b_0 w(n) + b_1 w(n-1) + b_2 w(n-2)$$

# Channel Filter

## Implementation

```
void biquad_df2(fract16 *input, fract16 *output, fract16 *coeff, uint32 N) {
    fract16 xn=0;
    fract16 y1=0; fract16 y2=0; fract16 y3=0; fract16 y4=0;
    fract16 w1n=0; fract16  w1n1=0; fract16  w1n2 = 0;
    fract16 w2n=0; fract16  w2n1=0; fract16  w2n2 = 0;
    fract16 w3n=0; fract16  w3n1=0; fract16  w3n2 = 0;
    fract16 w4n=0; fract16  w4n1=0; fract16  w4n2 = 0;

    int i;
    for (i = 0; i < N; i++) {

        xn  = input[i];

        w1n = ((xn<<15) - coeff[3]*w1n1 - coeff[4]*w1n2)       >> 15;
        y1  = (coeff[0]*w1n + coeff[1]*w1n1 + coeff[2]*w1n2)    >> 15;

        w2n = ((y1<<15) - coeff[9]*w2n1 - coeff[5]*w2n2)       >> 15;
        y2  = (coeff[6]*w2n + coeff[7]*w2n1 + coeff[8]*w2n2)    >> 15;

        w3n = ((y2<<15) - coeff[13]*w3n1 - coeff[14]*w3n2)     >> 15;
        y3  = (coeff[10]*w3n + coeff[11]*w3n1 + coeff[12]*w3n2) >> 15;

        w4n = ((y3<<15) - coeff[19]*w4n1 - coeff[15]*w4n2)     >> 15;
        y4  = (coeff[16]*w4n + coeff[17]*w4n1 + coeff[18]*w4n2) >> 15;

        output[i] = y4;

        w4n2 = w4n1;
        w4n1 = w4n;
        w3n2 = w3n1;
        w3n1 = w3n;
        w2n2 = w2n1;
        w2n1 = w2n;
        w1n2 = w1n1;
        w1n1 = w1n;
    }
```

# Channel Filter

## Implementation

- C-Code

```
w1n = ((xn<<15) - coeff[3]*w1n1 - coeff[4]*w1n2)      >> 15;
y1  = (coeff[0]*w1n + coeff[1]*w1n1 + coeff[2]*w1n2)  >> 15;
```

- Assembly

```
A1   = R0.L * R1.L,  A0 = R0.L * R1.H         || R3 = [P0+4];
R0.H = (A1 -= R2.L * R7.L), A0 -= R2.H * R7.L || R4 = [P0+8];

A1   = R0.L * R3.L,  R2.L = (A0 -= R0.H * R3.H);
R2.H = (A1 -= R0.H * R4.L) || R1 = [P0+12];
M0 = R2;
```

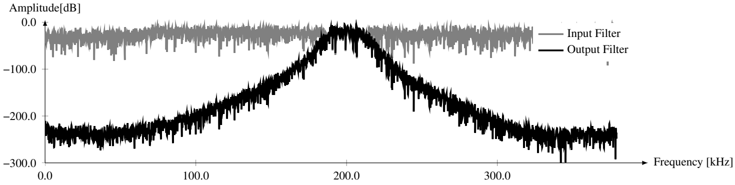





Dept. of Electronic Systems
Aalborg University
Denmark

# Channel Filter
## Test on Blackfin DSP

DSP test with Q15 frequency sweep and C-filter



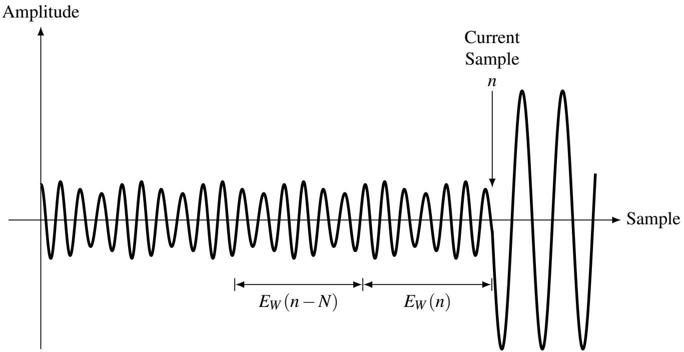

- Frequency Characteristic as specified.
- Theoretical cycle usage of 1.8% (Optimal)




Dept. of Electronic Systems
Aalborg University
Denmark

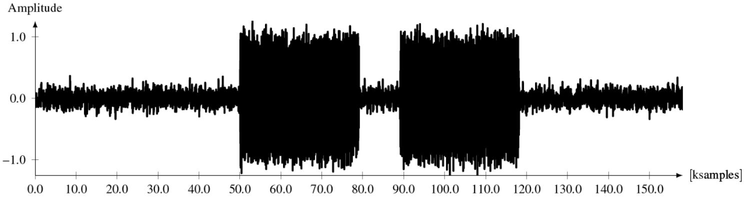
$$E_w(n) = \frac{1}{N} \sum_{i=n-N+1}^{n} x_i^2$$

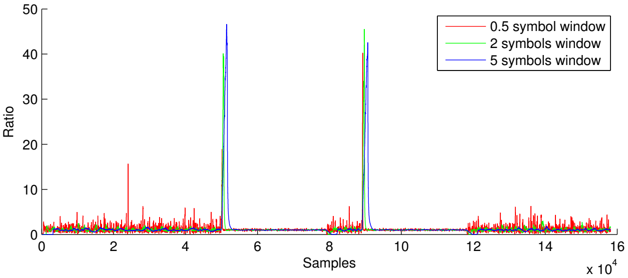
$$Ratio(n) = \frac{E_w(n)}{E_w(n-N)}$$



# Packet Detection

## Test of Window Size at 5 dB SNR





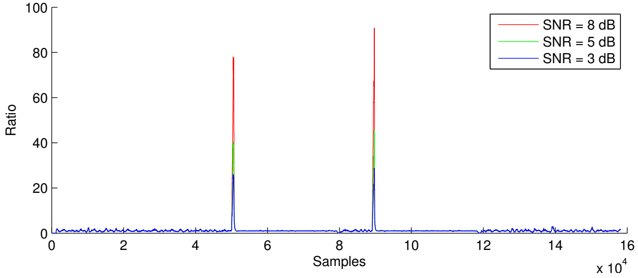


Dept. of Electronic Systems
Aalborg University
Denmark

# Packet Detection
## Test of SNR Influence with Window Size of 2 Symbols

$$W_{model} \approx \mathcal{N}(\mu, \sigma^2), \mu = 0$$

$$P_n = \sigma^2 = \frac{P_s}{SNR}$$

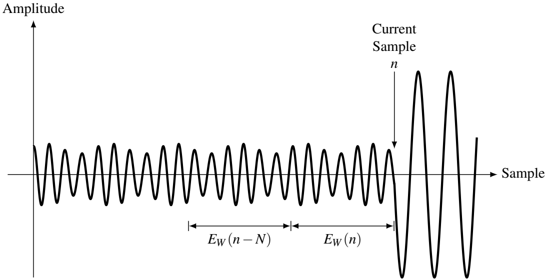






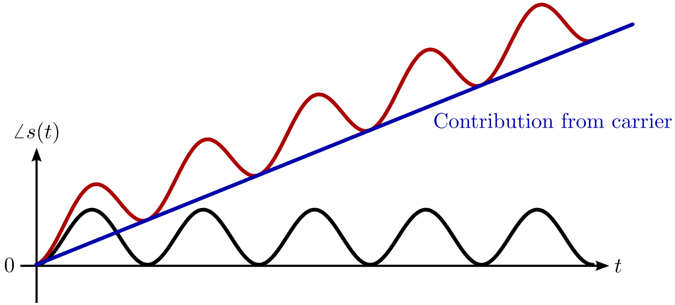
Dept. of Electronic Systems
Aalborg University
Denmark

$$E_w(n) = E_w(n) - \frac{x_{n-N}^2}{N} + \frac{x_n^2}{N}$$




- Window size power of two.
- Store old windows as a constant vector
- 2 mult, 1 sub, 1 add, 1 division

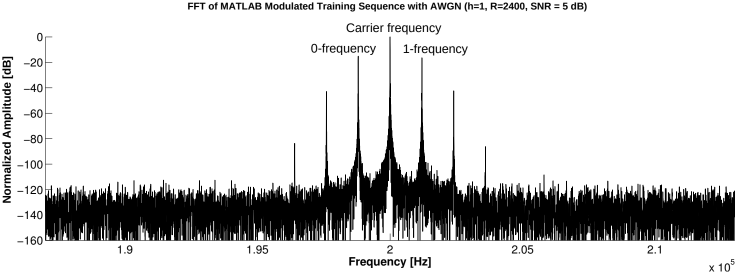
- Specify contribution from carrier
  - $\angle s(t) = \phi(t) + 2\pi f_c t$







Dept. of Electronic Systems
Aalborg University
Denmark




- Performing an 8192-point FFT on the DSP
- 8192 samples are approximately 13 bytes of training sequence
  - $f_s = 758\,272\,\mathrm{Hz}$, $R = 9600$, $\mathrm{SPS} \approx 79$

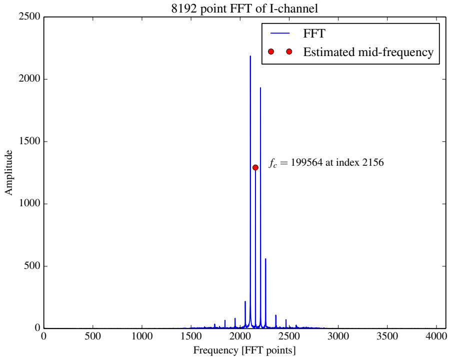
| $N$ [samples] | $f_{\mathrm{res}}$ [Hz] | FFT cycles [kcycles] | max cycles [kcycles] | FFT usage [%] |
|---|---|---|---|---|
| 8192 | 92.6 | 213 | 142.433 | 0.150 |





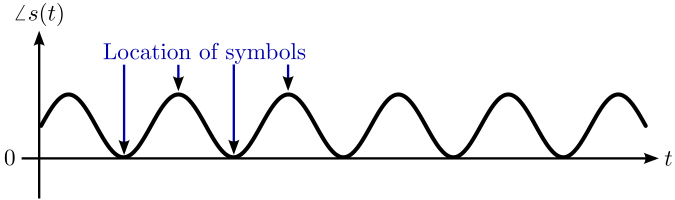

Dept. of Electronic Systems
Aalborg University
Denmark

Tested with signal generator.




- Actual mid-frequency
  - 199 567 Hz
- Estimated mid-frequency
  - 199 564 Hz

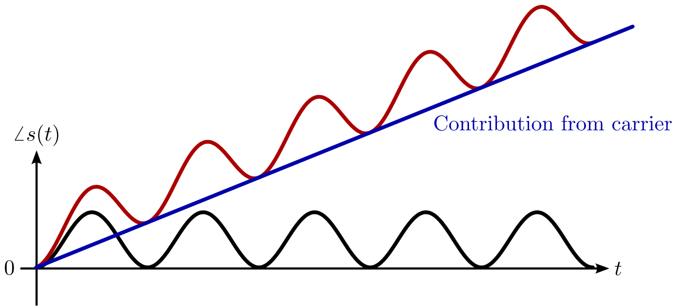
- Timing is needed to decide the value of each symbol






Dept. of Electronic Systems
Aalborg University
Denmark

- The downconverted phase-signal is used for time synchronization
- The phase-signal is computed
- The slope added by the carrier and Doppler shift is subtracted
  - $\angle s(t) = \phi(t) + 2\pi f_{c,est} t$

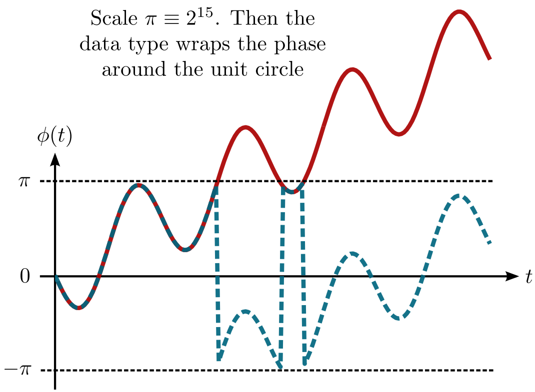








Dept. of Electronic Systems
Aalborg University
Denmark



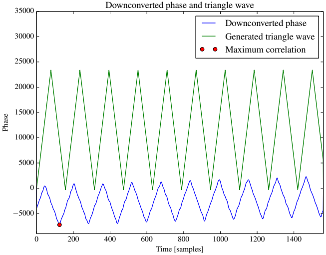

- Wrapping: $\texttt{arg\_fr16(phi)} \in [-\pi, \pi)$, $\texttt{fract16} \in [-2^{15}, 2^{15})$.
- Scale $\pi \equiv 2^{15}$ – wraps around unit circle.

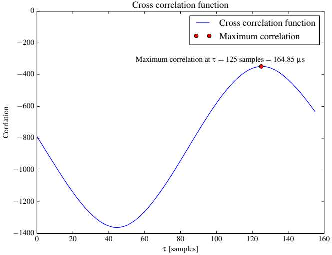



Dept. of Electronic Systems
Aalborg University
Denmark

- Cross-correlation of downconverted phase-signal, $y_2$, and pre-computed triangle wave, $y_1$

$$\psi_{y_1 y_2}[\tau] = \frac{1}{N} \sum_{n=0}^{N-2T_b-1} y_1[n] y_2[n+\tau], \quad \tau \in [0,\ 2T_b]$$



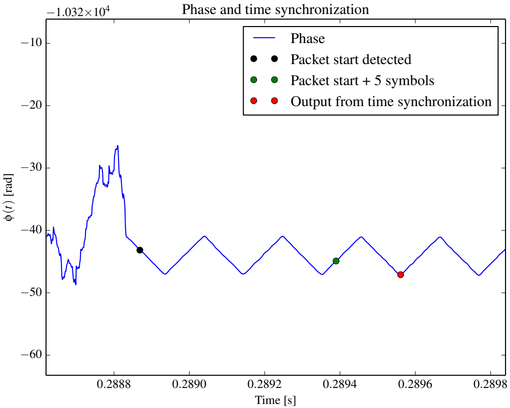




Dept. of Electronic Systems
Aalborg University
Denmark

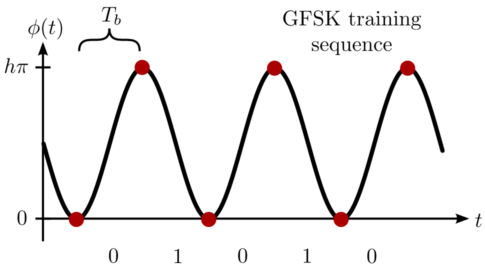
# Time Synchronization

## Test Results



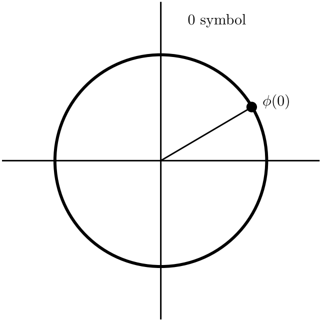
# Symbol Decision

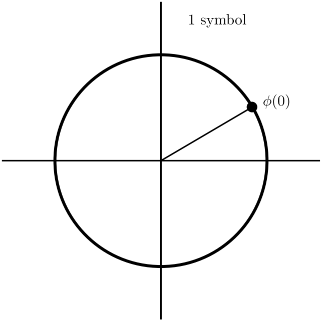
Phase ⟶ Symbols




- Positive/negative phase change ⇔ 1/0 symbol.
- $T_b = f_s / R = 758\,272 / 9600 = 78.987$ samples/s.
- From measurement: $T_b = 78.63$ samples/s.
- Fractional number of samples:

$$T_b = [78\,78\,80\,78\,78\,80\,78\,78\,80\,78\,78\,80\,78\,78\,80\,78\,78\,80\,78] \sim 78.63$$

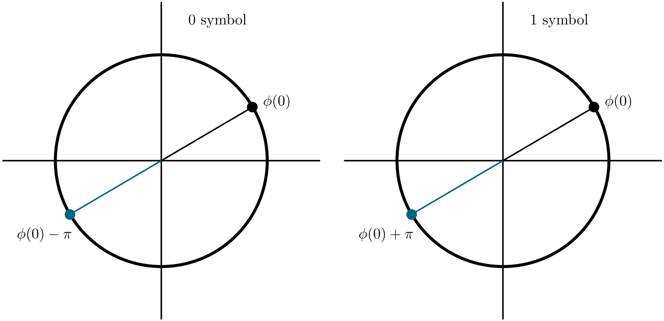



Wrapped phase at the beginning of a symbol.

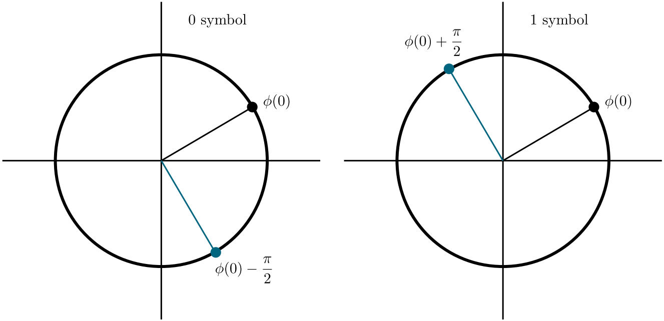
# Symbol Decision

## Wrapping Phase $\longrightarrow$ Symbols




Next symbol (for $h = 1$): $\phi(0) + \pi = \phi(0) - \pi$.

- The same for 0- and 1-symbol.




Half way to next symbol: Here is a difference!

$$\hat{m}_n = \begin{cases} 1 & \text{if } \hat{\phi}_{n+0.5} - \hat{\phi}_n > 0 \\ 0 & \text{if } \hat{\phi}_{n+0.5} - \hat{\phi}_n < 0 \\ \text{undefined} & \text{otherwise} \end{cases}$$

Sync word: Symbol-wise comparison.

Tested with signal generator.

## Drifting

- Little drifting with $T_b \equiv 79\,\mathrm{samples}/\mathrm{s}$.
- Correct demodulation of small packet.

## With Slope

- $f_{c,\mathrm{err}} \approx 4000\,\mathrm{Hz}$.
- Correct demodulation of small packet.

## Finding Sync Word and FSM

- Sync word $=$ `OZ4CUB`, FSM $=$ `0xA6` (small packet).

## Cycle Count

Total number of clock cycles available: 142 433 491.

| Module | Cycles Used | DSP Usage |
|---|---|---|
| Sampler | 109 263 467 | 77 % |
| Channel Filter | 295 467 922 | 207 % |
| Packet Detector | 26 435 831 | 19 % |
| Mid-Frequency Estimator | 14 322 454 | 10 % |
| Time Synchronization | 26 638 224 | 19 % |
| Symbol Decision | 4 587 500 | 3 % |
| Total | 476 715 398 | 335 % |

- C filter $\rightarrow$ Assembly: 86 913 124 clock cycles $\sim$ 61 %.
- Downsample complex signal.
- Move data when it is processed.

# Acceptance Test

Packet From AAUSAT3

## Signals

- Strong, medium, weak.

## Criteria

- Correct sync word and FSM.
- Cannot check symbol-by-symbol.

## Results

| | |
|---|---|
| Strong | Found sync word and FSM. |
| Medium | Found sync word and FSM. |
| Weak | Wrong sync word – found later in buffer. |

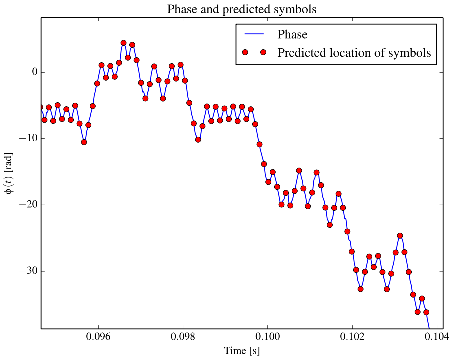




Dept. of Electronic Systems
Aalborg University
Denmark

# Acceptance Test

## Packet From AAUSAT3

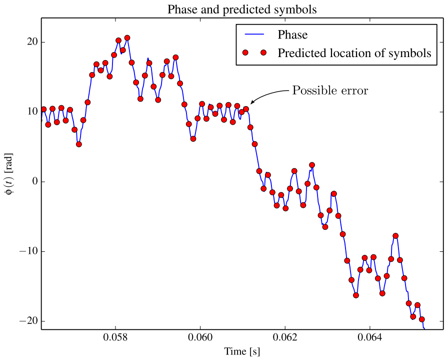



- Medium signal.
- Sampled $I$ and $Q$ + estimated $f_c \rightarrow$ Phase.
- Time synchronized + variable $T_b \rightarrow$ Predicted location of symbols.
- No drifting – correct sync word and FSM.




Dept. of Electronic Systems
Aalborg University
Denmark

# Acceptance Test
## Packet From AAUSAT3




- Weak signal.
- Sampled $I$ and $Q$ + estimated $f_c \rightarrow$ Phase.
- Time synchronized + variable $T_b \rightarrow$ Predicted location of symbols.
- No drifting – wrong sync word.




Dept. of Electronic Systems
Aalborg University
Denmark

# Conclusion

- Detect packet from signal generator.
- Correctly demodulate recording from AAUSAT3.
- Prepared for real-time optimization.



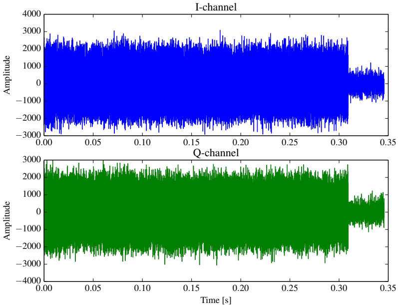


Dept. of Electronic Systems
Aalborg University
Denmark

# Demonstration

## What is Shown?

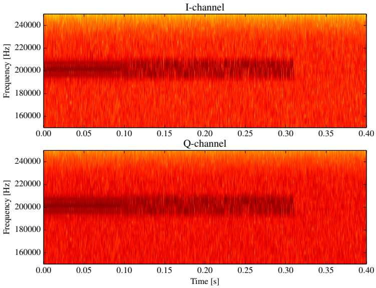
- Medium signal strength.
- Demodulation on DSP.
- Amplitude, spectrogram, FFT (training).
- Interactive phase + decisions.



# Demonstration

## What is Shown?

- Medium signal strength.
- Demodulation on DSP.
- Amplitude, spectrogram, FFT (training).
- Interactive phase + decisions.







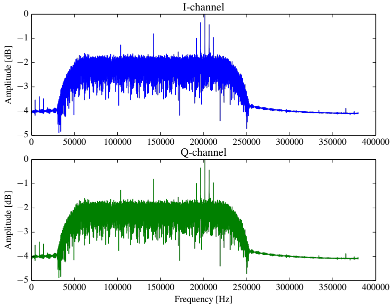


Dept. of Electronic Systems
Aalborg University
Denmark

# Demonstration

## What is Shown?

- Medium signal strength.
- Demodulation on DSP.
- Amplitude, spectrogram, FFT (training).
- Interactive phase + decisions.







Dept. of Electronic Systems
Aalborg University
Denmark