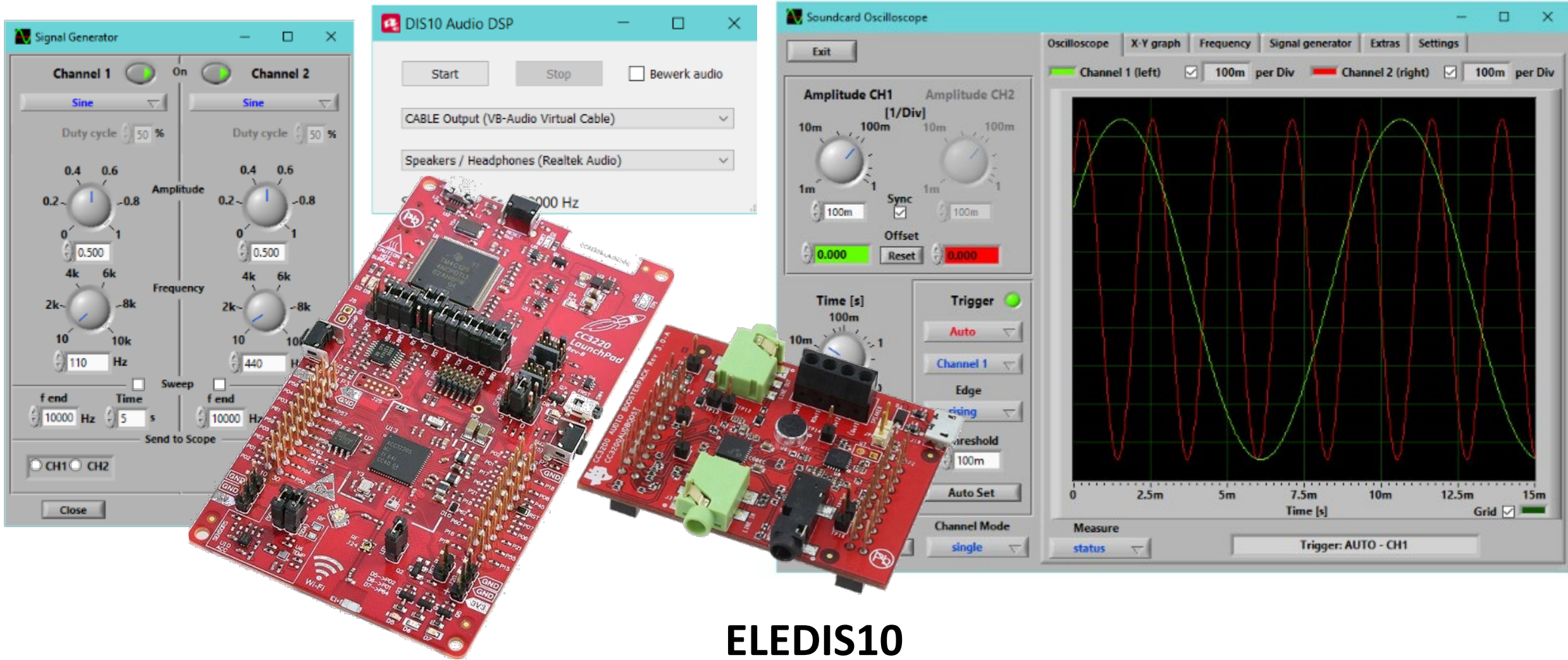




ELEDIS10

Digitale Systemen

Opleiding Elektrotechniek

BEREKENINGEN MET KOMMAGETALLEN

Voorbeeld FIR filter coëfficiënten

- Definitie laagdoorlaatfilter in frequentiedomein:

$$H(\Omega) = \begin{cases} 1, & -\Omega_c \leq \Omega \leq \Omega_c \\ 0, & \Omega < -\Omega_c \vee \Omega > \Omega_c \end{cases}$$

- Impulsresponsie in het tijddomein:

$$h[n] = \frac{1}{\pi n} \sin(\Omega_c n), \quad h[0] = \frac{\Omega_c}{\pi}$$

- Stel $\Omega_c = \frac{\pi}{4}$, 4e orde FIR filter:

- $h[n] = \frac{1}{\pi n} \sin\left(\frac{\pi}{4} n\right), \quad h[0] = 0,25; \quad h[-1] = h[1] = \frac{\sqrt{2}}{2\pi} \approx 0,2251;$

- $h[-2] = h[2] = \frac{1}{2\pi} \approx 0,1592$

- Hoe implementeren we (zonder na te denken) kommagetallen in C?

Floating-point

- Normaal gesproken gebruiken we in C het type **double** of **float** om kommagetallen weer te geven.

float: 8 bits

float: 23 bits

double: 11 bits

double: 52 bits

S	Exponent	Fraction
---	----------	----------

$$x = (-1)^S \times (1 + \text{Fraction}) \times 2^{(\text{Exponent} - \text{Bias})}$$

Bron: slides Computer Organization and Design, The Hardware/Software Interface by David A. Patterson, John L. Hennessy

Fixed-point versus floating-point

- **Floating-point** getallen:
 - IEEE Standaard voor Floating-Point Arithmetic (IEEE 754).
 - Ondersteund door alle gangbare programmeertalen.
 - Groot dynamisch bereik: float: $1.2\text{E}-38$ tot $3.4\text{E}+38$, double: $2.3\text{E}-308$ tot $1.7\text{E}+308$.
 - De precisie varieert met de waarde van het getal: een float heeft 24 significante bits (7 decimale digits) en een double heeft 53 significante bits (16 decimale digits), de precisie hangt af van de plek van de komma (en die varieert).
 - Berekeningen kosten meer tijd / chip oppervlak / energie dan berekeningen met gehele getallen (integers).
- **Fixed-point** getallen:
 - Niet gestandaardiseerd en (daarom) niet ondersteund door de gangbare programmeertalen.
 - Beperkt dynamisch bereik (hou rekening met under- en overflows).
 - Precisie staat vast.
 - Berekeningen zijn vrijwel gelijk aan berekeningen met gehele getallen (integers).

FIXED-POINT VOORBEELD 1

Integer getallen zijn binair vrijwel altijd beschreven in **two's complement**

$$x = 0100100000011000_2 = 18456_{10}$$

Fixed-point getallen zijn vaak beschreven in het **$Qm.n$** notatie

$Qm.n$ = een two's complement getal met m bits vóór en n bits na de binaire komma en een impliciet tekenbit (signbit)

$$\begin{aligned} Q0.15: x &= 0.100100000011000_2 \\ x &= 2^{-1} + 2^{-4} + 2^{-11} + 2^{-12} = 0.56323_{10} \end{aligned}$$

$$\begin{aligned} Q2.13: x &= 010.0100000011000_2 \\ x &= 2^1 + 2^{-2} + 2^{-9} + 2^{-10} = 2.25293_{10} \end{aligned}$$

$$\begin{aligned} Q5.10: x &= 010010.0000011000_2 \\ x &= 2^4 + 2^1 + 2^{-6} + 2^{-7} = 18.02344_{10} \end{aligned}$$

FIXED-POINT VOORBEELD 2

Two's complement integer:

$$y = 1100111111111000_2 = (-)0011000000001000_2 = -12296_{10}$$

$$Q0.15: y = (-)0.011000000001000_2$$

$$y = -(2^{-2} + 2^{-3} + 2^{-12}) = -0.37524414_{10}$$

$$Q2.13: y = (-)001.1000000001000_2$$

$$y = -(2^0 + 2^{-1} + 2^{-10}) = -1.500976563_{10}$$

$$Q5.10: y = (-)001100.0000001000_2$$

$$x = -(2^3 + 2^2 + 2^{-7}) = -12.0078125_{10}$$

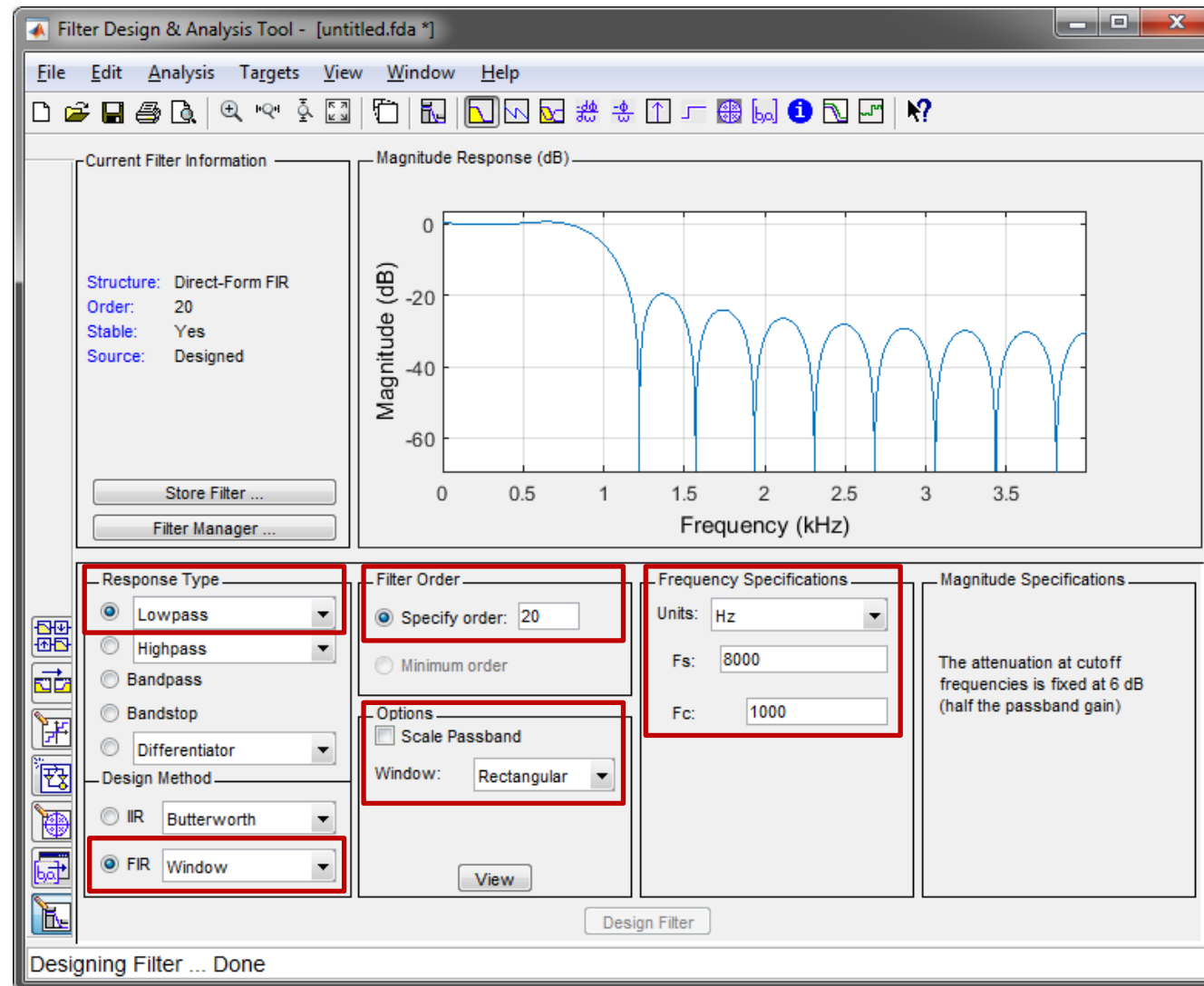
FIXED-POINT MATLAB NOTATIE

Standaard notatie	Matlab
$Q0.15$	$s16,15$
$Q2.13$	$s16,13$
$Q5.10$	$s16,10$

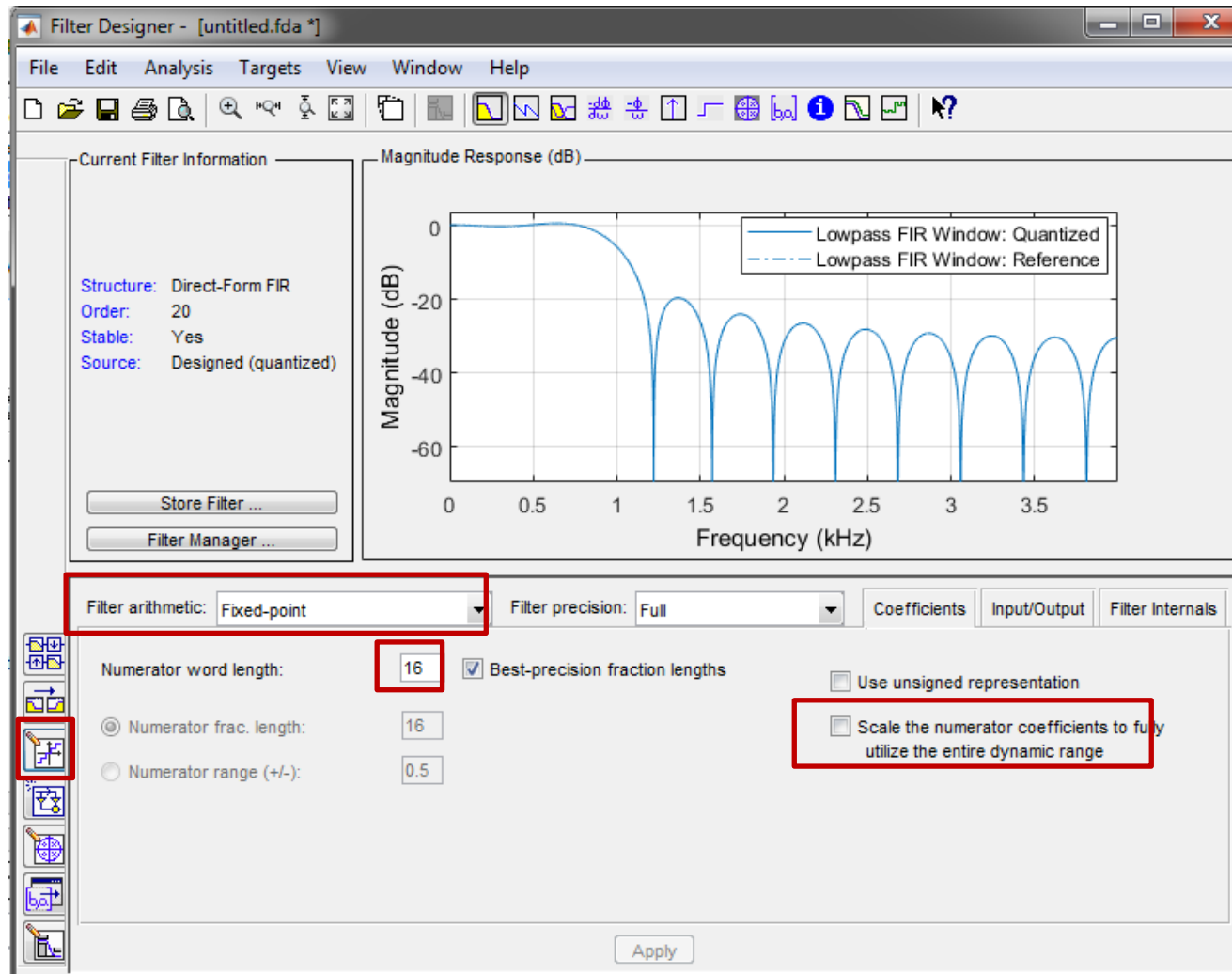
16 bits totaal waarvan 10
bits rechts van de komma

Dus 6 bits links van de
komma (inclusief tekenbit)

FIR FILTER COËFFICIENTEN 1

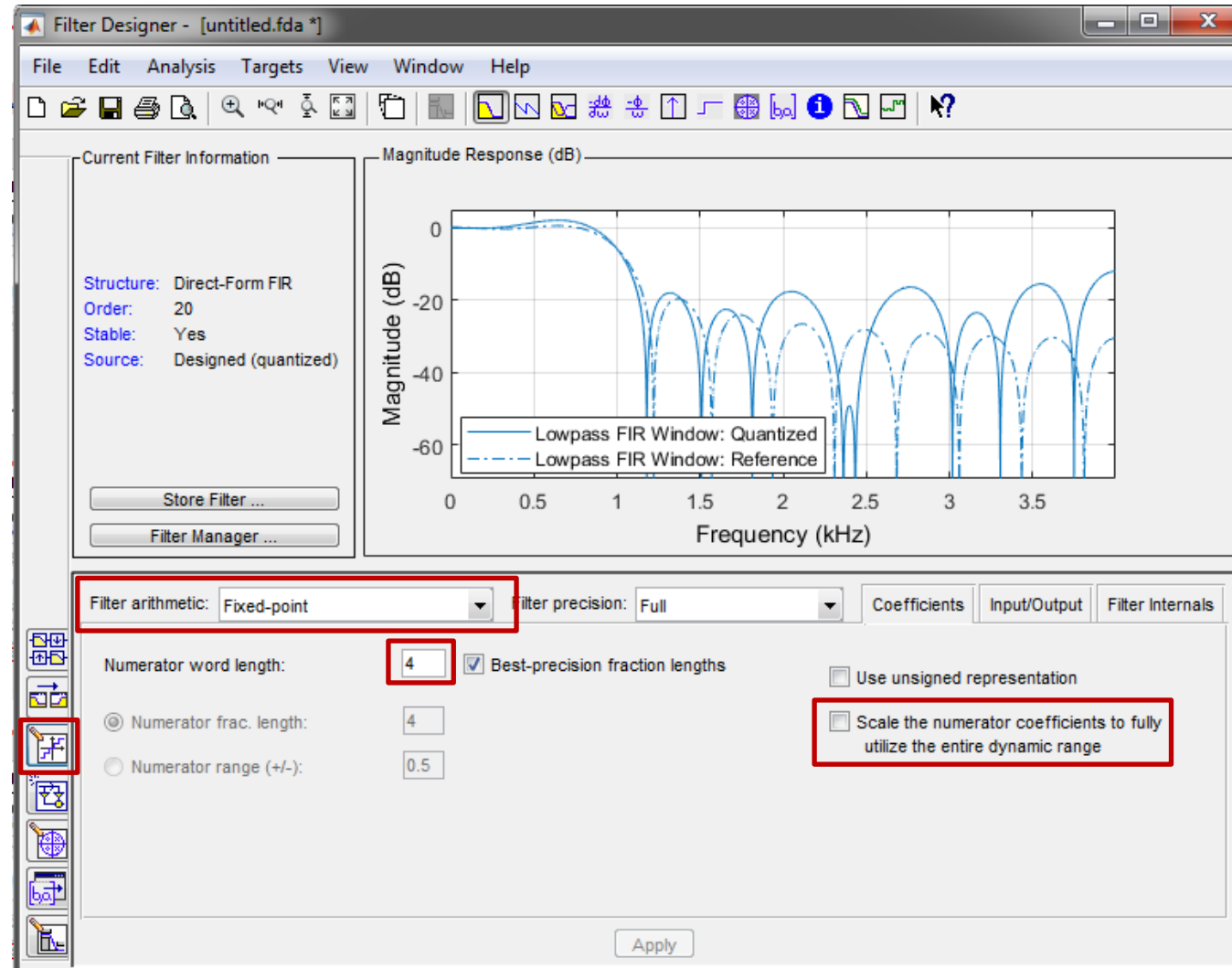


FIR FILTER COEFFICIENTS 2



Als je deze optie mist, dan heb je de **Fixed-Point Designer Toolbox** en/of de **DSP System Toolbox** niet geïnstalleerd:
<https://nl.mathworks.com/help/fixedpoint/index.html> en/of <https://nl.mathworks.com/products/dsp-system.html>

FIR FILTER COEFFICIENTS 3



4 bits precisie is hier duidelijk te weinig.

FIR FILTER COEFFICIENTS 4

Targets -> Generate C Header

Generate C Header

Variable names in C header file

Numerator: Numerator length:

Data type to use in export

☐ Export suggested: Signed 16-bit integer with 16-bit fractional length

☒ Export as: Signed 16-bit integer
Fractional length: 16

Generate

C-HEADER

```

/*
 * Filter Coefficients (C Source) generated by the Filter Design and Analysis Tool
 * Generated by MATLAB(R) 9.0 and the DSP System Toolbox 9.2.
 * Generated on: 15-Sep-2016 15:23:37
 */

/*
 * Discrete-Time FIR Filter (real)
 * -----
 * Filter Structure   : Direct-Form FIR
 * Filter Length      : 21
 * Stable             : Yes
 * Linear Phase       : Yes (Type 1)
 * Arithmetic         : fixed
 * Numerator          : s16,16 -> [-5.000000e-01 5.000000e-01]
 * Input              : s16,15 -> [-1 1)
 * Filter Internals   : Full Precision
 * Output             : s33,31 -> [-2 2) (auto determined)
 * Product            : s31,31 -> [-5.000000e-01 5.000000e-01) (auto determined)
 * Accumulator        : s33,31 -> [-2 2) (auto determined)
 * Round Mode         : No rounding
 * Overflow Mode      : No overflow
 */

/* General type conversion for MATLAB generated C-code */
#include "tmwtypes.h"

/*
 * Expected path to tmwtypes.h
 * C:\Program Files\MATLAB_2016a\extern\include\tmwtypes.h
 */

const int BL = 21;
const int16_T B[21] = {
    2086, 1639, 0, -2107, -3477, -2950, 0, 4917, 10430,
    14751, 16384, 14751, 10430, 4917, 0, -2950, -3477, -2107,
    0, 1639, 2086
};

```

16 bit fixed-point
getal met 16 bits
rechts van de komma.
(two's complement!)

Bijvoorbeeld:

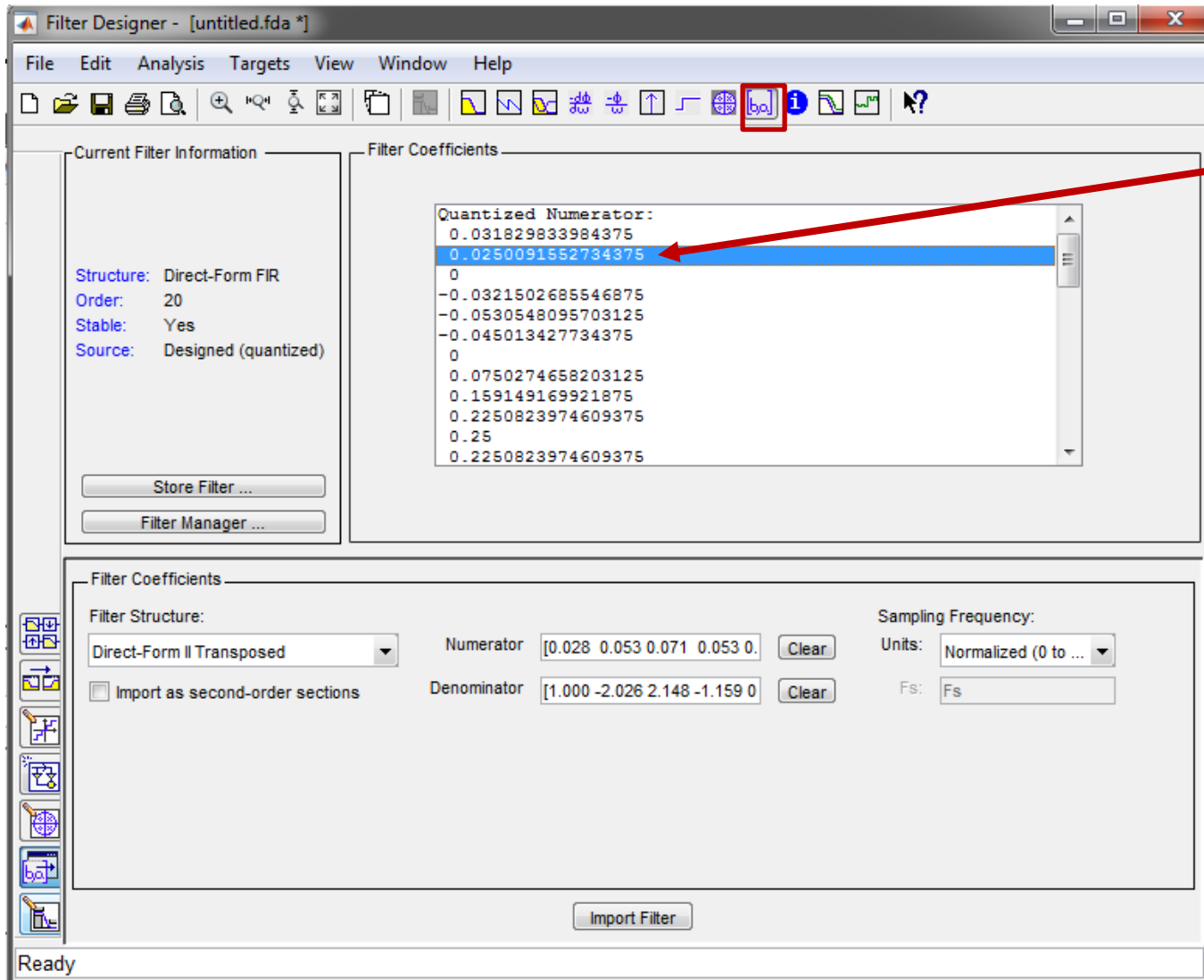
$$1639_{10} = 0000011001100111_2$$

naar fixed point:

$$0.0000011001100111_2$$

$$= +(2^{-6} + 2^{-7} + 2^{-10} + 2^{-11} + 2^{-14} + 2^{-15} + 2^{-16})$$
$$\approx 0.025009155_{10}$$

CONTROLLEREN



$\approx 0.025009155_{10}$
Correct!

Makkelijker:

$$\frac{1639}{2^{16}} = 0.025009155_{10}$$

OPTELLING

0011	=	0.375
0001	=	0.125
-----+		-----+
0100		0.500

Q0.3 notatie

==>

Zorg ervoor dat punten onder elkaar staan.

VERMENIGVULDIGING

0111	=	0.875
0001	=	0.125
-----*		-----*
0111		4375
0000		1750
0000		875
0000		
-----+		-----+
0000111		109375

Q0.3 notatie

==>

3+3 getallen rechts van de komma dus de komma moet worden geplaatst op:

0.000111 = 0.109375 (Q0.6 notatie)

Eventueel terug naar Q0.3 notatie (afronden):

0.001 = 0.125

Zie:

https://bitbucket.org/HR_ELEKTRO/dis10/wiki/Opdrachten/Hardware_Opdrachten/DIS10_Hardware_Fixed-point_berekeningen_ebook.pdf

Samenvatting

- **Fixed-point versus Floating-point:**

- Voordelen:

- Kost minder tijd / chip oppervlak / energie
 - Vaste precisie

- Nadelen:

- Klein dynamisch bereik

- **Fixed-point berekeningen:**

- Vermenigvuldiging: gebruik integer vermenigvuldiging en onthoudt de positie van de komma:

- $Q1.14 \cdot Q2.13 \rightarrow Q3.27$

- Optelling: gebruik integer optelling, vergeet niet om de komma's op dezelfde plek te plaatsen door de schuiven:

- $Q1.14 + Q2.13 \rightarrow Q1.14 \gg 1 + Q2.13 \rightarrow Q2.13 + Q2.13 \rightarrow Q3.13$