



# DSP ontwikkeling

10 Dec 2010

Jan Mennekens

Artesis 2010



# overzicht

## 1. know what you're doing



# overzicht

1. know what you're doing
2. think before you do it



# overzicht

1. know what you're doing
2. think before you do it
3. how will you do it?



# overzicht

1. know what you're doing
2. think before you do it
3. how will you do it?
4. do it

# overzicht

1. know what you're doing
2. think before you do it
3. how will you do it?
4. do it
5. now look what you've done



# stap 1

## know what you're doing

# algorithmes

het verschil tussen

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau$$

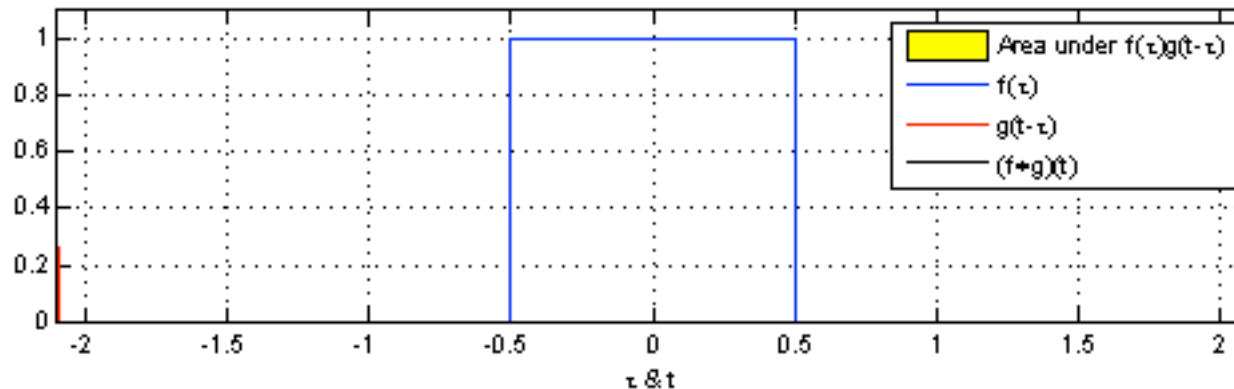


# algorithmes

het verschil tussen

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau$$

en

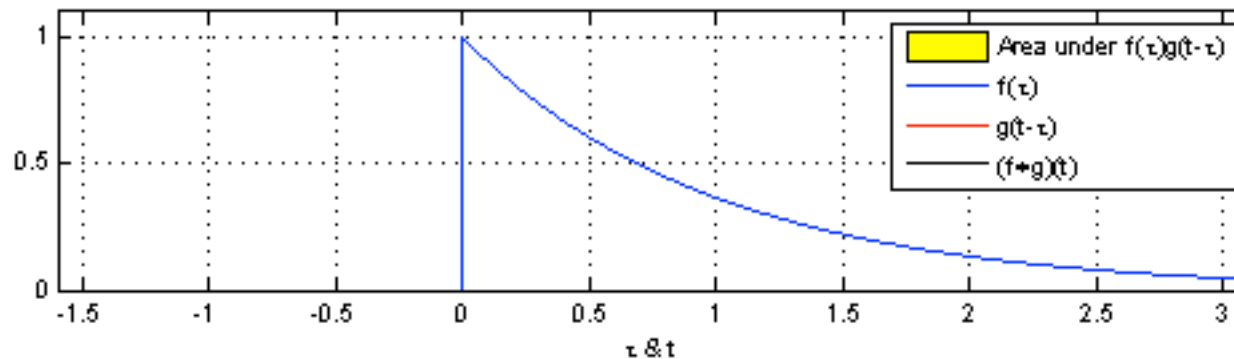


# algorithmes

het verschil tussen

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau$$

en



# algorithmes

het verschil tussen

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau$$

en

```
def conv(x, y):  
    P, Q, N = len(x), len(y), len(x)+len(y)-1  
    z = []  
    for k in range (N):  
        t, lower, upper = 0, max(0, k-(Q-1)), min(P-1, k)  
        for i in range(lower, upper+1):  
            t = t + x[i] * y[k-i]  
        z.append(t)  
    return z
```



# convolutie

- bepalen mate van gelijkheid

# convolutie

- bepalen mate van gelijkheid
- gebruikt voor
  - signaal/echo-detectie
  - beeld/rand-herkenning

# convolutie

- bepalen mate van gelijkheid
- gebruikt voor
  - signaal/echo-detectie
  - beeld/rand-herkenning
- gebruikt in
  - ontvangers / radar /sonar
  - medische apparatuur



# convolutie

- demo

# convolutie

- demo
- simpel te proberen
- werkt prima met ruis
- werkt nog beter met template



- transformatie frequentie-domein
  - welke frequenties zitten in dit signaal

- transformatie frequentie-domein
  - welke frequenties zitten in dit signaal
- gebruikt voor
  - draaggolf-detectie
  - harmonischen-detectie

- transformatie frequentie-domein
  - welke frequenties zitten in dit signaal
- gebruikt voor
  - draaggolf-detectie
  - harmonischen-detectie
- gebruikt in
  - ontvangers / radar /sonar
  - mechanische testbanken



FFT

- demo

- demo
- simpel te proberen
- werkt prima met ruis
- langere FFT's zijn beter
- recupereer energie indien mogelijk

# filters

- uitfilteren frequentie-componenten

# filters

- uitfilteren frequentie-componenten
- gebruikt voor
  - signaal-zuivering

# filters

- uitfilteren frequentie-componenten
- gebruikt voor
  - signaal-zuivering
- gebruikt in
  - ontvangers / radar /sonar
  - audio/video toepassingen
  - telefonie (echo-canceling)





# filter

- demo

- demo
- simpel te proberen
- eenvoudig te construeren
- is eigenlijk een convolutie



# conclusie stap 1

DSP is CONVOLUTIE



# stap 2

think before you do it



# architectuur



# architectuur

- simuleer eerst !!!!!



# architectuur



# architectuur

- simuleer eerst !!!!!





# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!



# architectuur



# architectuur

- simuleer eerst !!!!!



# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!



# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel



# architectuur



# architectuur

- simuleer eerst !!!!!



# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!





# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel

# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel
- hou dit proefmodel bij en up-to-date



# architectuur



# architectuur

- simuleer eerst !!!!!



# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!



# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel

# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel
- hou dit proefmodel bij en up-to-date



# architectuur





# architectuur

- simuleer eerst !!!!!



# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!



# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel

# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel
- hou dit proefmodel bij en up-to-date

# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel
- hou dit proefmodel bij en up-to-date
- hergebruik zoveel mogelijk



# architectuur



# architectuur

- simuleer eerst !!!!!



# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!





# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel

# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel
- hou dit proefmodel bij en up-to-date

# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel
- hou dit proefmodel bij en up-to-date
- hergebruik zoveel mogelijk

# architectuur

- simuleer eerst !!!!!
- simuleer nog eens !!!!!
- maak een proefmodel
- hou dit proefmodel bij en up-to-date
- hergebruik zoveel mogelijk
- wees niet bang om te herbeginnen



# conclusie stap 2

architectuur is belangrijk



# stap 3

## how will you do it?



DSP



# DSP

- DSP is convolutie



- DSP is convolutie
- DSP is heel veel MACs in korte tijd
  - meerdere bussen
  - meerdere cores
  - snelle interfaces

- DSP is convolutie
- DSP is heel veel MACs in korte tijd
  - meerdere bussen
  - meerdere cores
  - snelle interfaces
- en minder
  - veel memory
  - zware user-interfaces



# hardware



# hardware

- microcontroller



# hardware

- microcontroller
- standaard high-end PC



# hardware

- microcontroller
- standaard high-end PC
- DSP core



# hardware

- microcontroller
- standaard high-end PC
- DSP core
- DSP multi-core



# hardware

- microcontroller
- standaard high-end PC
- DSP core
- DSP multi-core
- GPU



# hardware

- microcontroller
- standaard high-end PC
- DSP core
- DSP multi-core
- GPU
- FPGA / discrete logica

# hardware

- microcontroller
- standaard high-end PC
- DSP core
- DSP multi-core
- GPU
- FPGA / discrete logica
- combinatie



# software



# software

- gesofisticeerde simulatie-paketten



# software

- gesofisticeerde simulatie-paketten
- IDE/dev-kit

# software

- gesofisticeerde simulatie-paketten
- IDE/dev-kit
- target language C / C++ / asm

# software

- gesofisticeerde simulatie-paketten
- IDE/dev-kit
- target language C / C++ / asm
- FPGA core design pakket

# software

- gesofisticeerde simulatie-paketten
- IDE/dev-kit
- target language C / C++ / asm
- FPGA core design pakket
- data-analyse tools





# conclusie stap 3

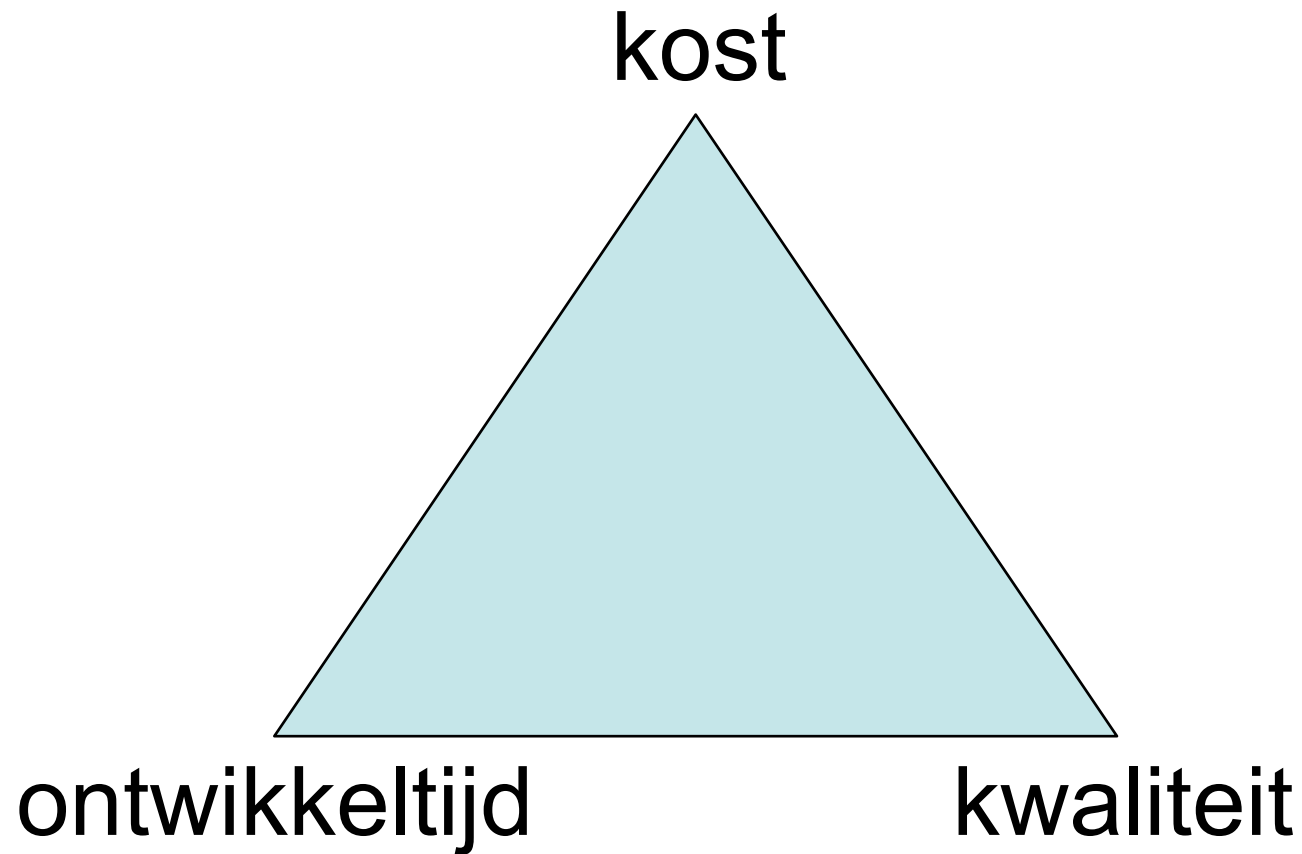
blijf op de hoogte van de  
technologieveranderingen



stap 4

do it

# decisions





# trade-offs



# trade-offs

- hardware of software



# trade-offs



# trade-offs

- hardware of software



# trade-offs

- hardware of software
- geheugen of tijd





# trade-offs



# trade-offs

- hardware of software



# trade-offs

- hardware of software
- geheugen of tijd



# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom



# trade-offs



# trade-offs

- hardware of software



# trade-offs

- hardware of software
- geheugen of tijd

# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom



# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom
- communicatie of rekenen



# trade-offs



# trade-offs

- hardware of software



# trade-offs

- hardware of software
- geheugen of tijd

# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom

# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom
- communicatie of rekenen

# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom
- communicatie of rekenen

# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom
- communicatie of rekenen

het beste algoritme wint  
**ALTIJD**





# trade-offs



# trade-offs

- hardware of software



# trade-offs

- hardware of software
- geheugen of tijd

# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom

# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom
- communicatie of rekenen

# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom
- communicatie of rekenen

# trade-offs

- hardware of software
- geheugen of tijd
- snelheid of stroom
- communicatie of rekenen

het beste algoritme wint  
**ALTIJD**



# conclusie stap 4

ken uw trade-offs





# stap 5

now look what you've done



# testing

- denk na over hoe je dat gaat testen

# testing

- denk na over hoe je dat gaat testen
- bouw je proefmodel als testbank

# testing

- denk na over hoe je dat gaat testen
- bouw je proefmodel als testbank
- voorzie testmogelijkheden op de HW

# testing

- denk na over hoe je dat gaat testen
- bouw je proefmodel als testbank
- voorzie testmogelijkheden op de HW
- automatiseer je testen

# testing

- denk na over hoe je dat gaat testen
- bouw je proefmodel als testbank
- voorzie testmogelijkheden op de HW
- automatiseer je testen
- loop je testen zo vaak mogelijk



# testing



# testing

- denk na over hoe je dat gaat testen



# testing

- denk na over hoe je dat gaat testen
- liefst zo vroeg mogelijk

# testing

- denk na over hoe je dat gaat testen
- liefst zo vroeg mogelijk



# testing



# testing

- denk na over hoe je dat gaat testen

# testing

- denk na over hoe je dat gaat testen
- liefst zo vroeg mogelijk

# testing

- denk na over hoe je dat gaat testen
- liefst zo vroeg mogelijk
- is deel van simulatie-model

# testing

- denk na over hoe je dat gaat testen
- liefst zo vroeg mogelijk
- is deel van simulatie-model

# voorbeeld setup

TCL  
simulation code

TCL  
function library

Python  
simulation code

Python  
libraries

SWIG  
wrapper

SWIG  
wrapper

SWIG  
wrapper

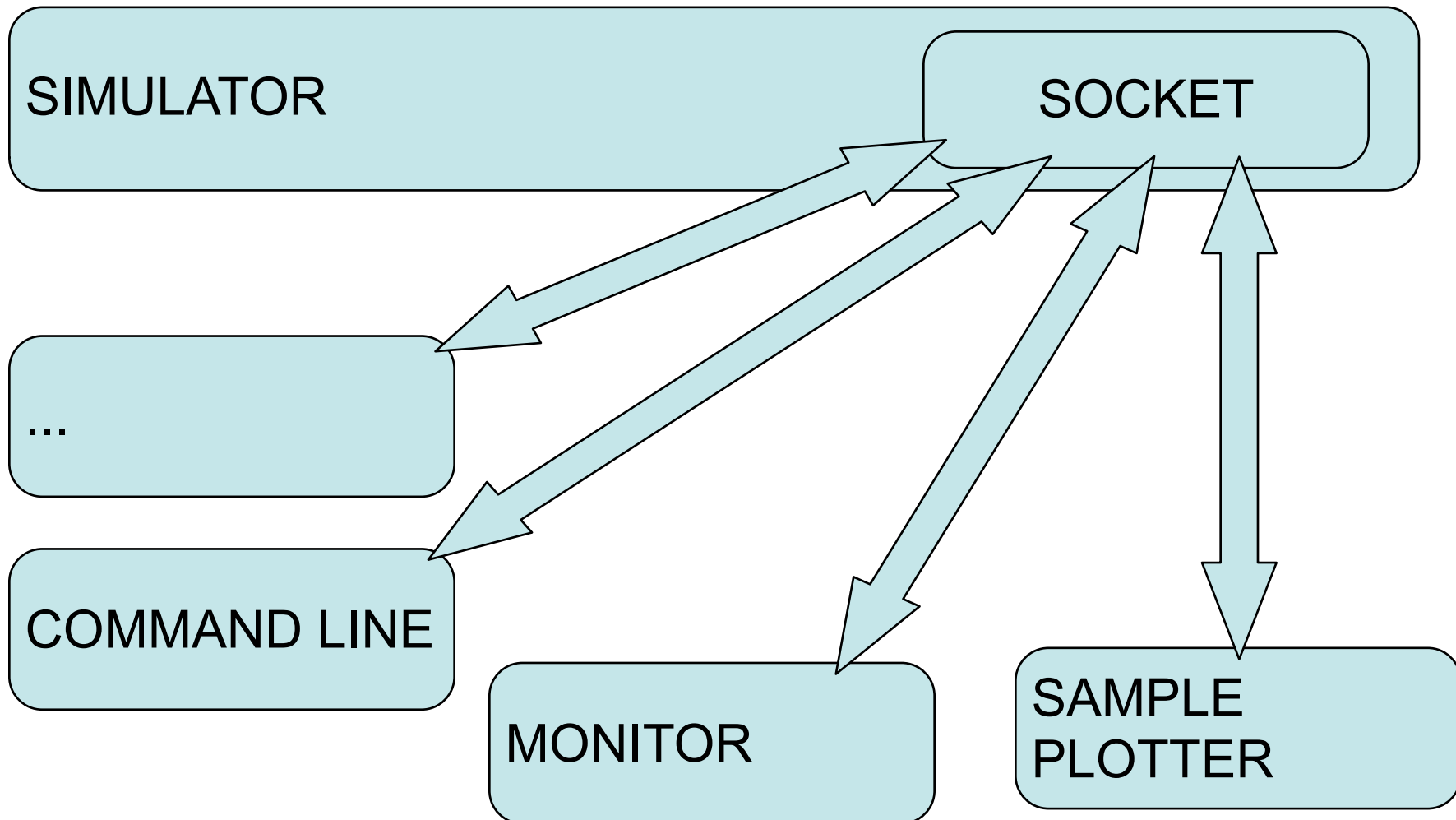
ANSI C DSP lib  
(fixed point)

ANSI C DSP lib  
(floating point)

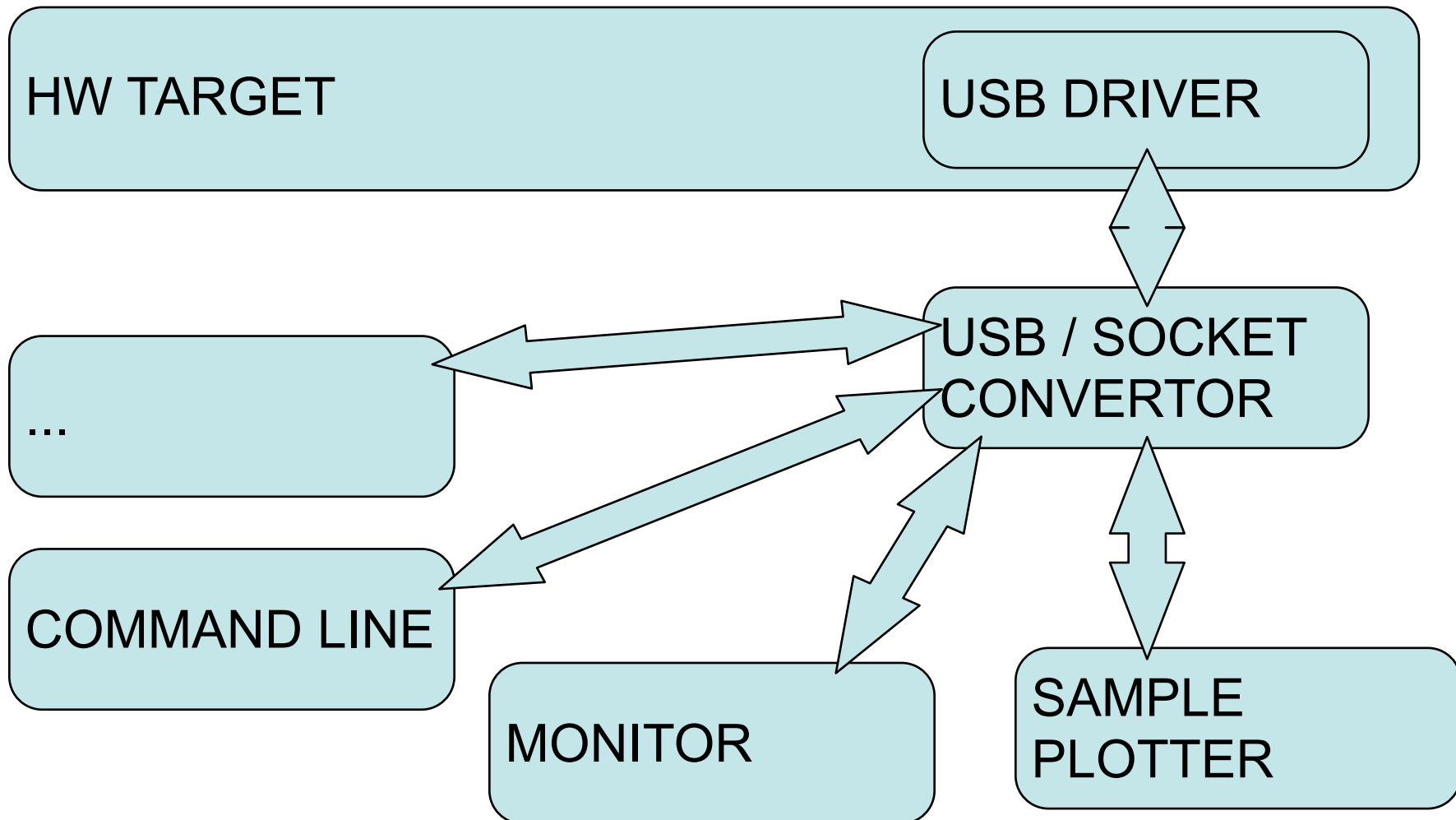
ANSI C FEC lib



# simulator setup



# target setup





# demo SWIG



# conclusie stap 5

the difference between a good  
design and a great design is the  
amount of testing done



vragen?

[www.dcim.be](http://www.dcim.be)

[jan@dcim.be](mailto:jan@dcim.be)



...anyone?....

[www.dcim.be](http://www.dcim.be)

[jan@dcim.be](mailto:jan@dcim.be)



# repository

```
hg clone https://  
JeanLeHacker@bitbucket.org/JeanLeHacker/  
artesis\_2010\_conv
```



...anyone?....

hg clone [https://  
JeanLeHacker@bitbucket.org/JeanLeHacker/  
artesis\\_2010\\_conv](https://JeanLeHacker@bitbucket.org/JeanLeHacker/artesis_2010_conv)





the end...

[www.dcim.be](http://www.dcim.be)

[jan@dcim.be](mailto:jan@dcim.be)



...anyone?....

[www.dcim.be](http://www.dcim.be)

[jan@dcim.be](mailto:jan@dcim.be)

# algorithmes

het verschil tussen

$$f\left(\frac{n}{2W}\right) = \frac{1}{2\pi} \int_{-2\pi W}^{2\pi W} F(\omega) e^{i\omega \frac{n}{2W}} d\omega.$$

# algorithmes

het verschil tussen

$$f\left(\frac{n}{2W}\right) = \frac{1}{2\pi} \int_{-2\pi W}^{2\pi W} F(\omega) e^{i\omega \frac{n}{2W}} d\omega.$$

en

