
Phantom Limb Pain Project
Create a smart Interface

Phantom Limb Pain Project

⇒ Reduce pain

- Create a realistic and user friendly system
- Moderate cost
- Adaptable to multiple users
- Self adaptable to a given user

3 months to :

1. Determine the usefull parameters
2. Choose the necessary hardware
3. Create a prototype
   and make it easily reusable

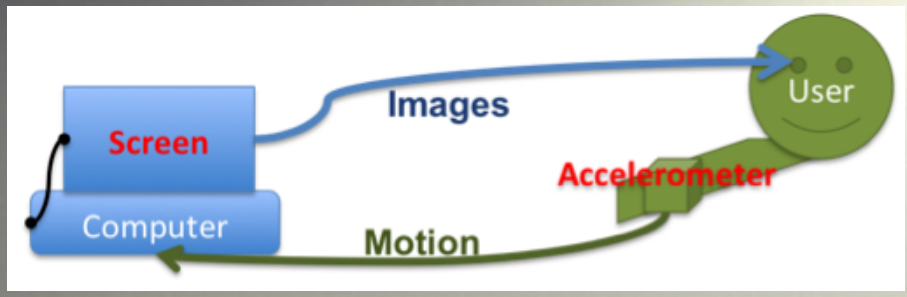
An appropriate model

Processing data to create a virtual limb

Make the system adaptable


An appropriate model
Processing data to create a virtual limb
Make the system adaptable

Context



- Impossible to use all sensory informations to correct or modify the motion path

- Determine the moves of the user's limb from the moves of an other part of his body
  - stump
  - symmetrical limb
  - eyes

- Use the accelerometer and get the maximum of workable data

Context

2 types of motion :

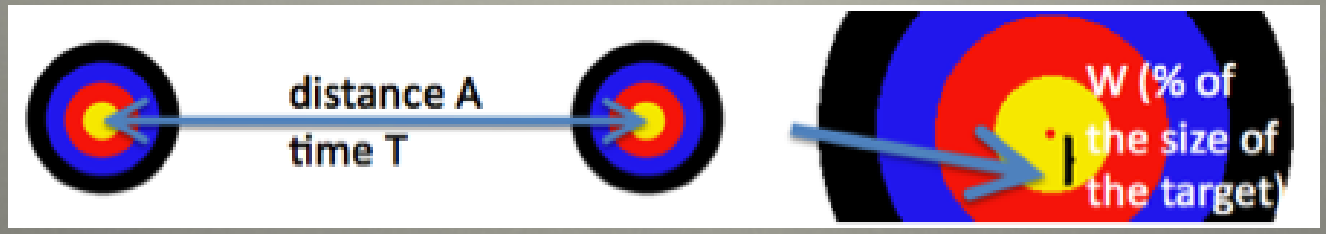
- Motion with user's feedback
- Motion without user's feedback

Task	Processing time of visual	Work
Alternate movements	500ms	[Woodworth 1999]
Target tracking movements	300~350ms	
Discrete movements of target attainment	190~260ms	[Keele *et al*. 1968]

Empirical knowledge

Woodworth (1899) and by Fitts (1954)

$$T = a + b.I_d = a + b.\log_2\left(\frac{2.A}{W}\right)$$



Empirical knowledge

MacKenzie (1992)

$$T = a + b.\log_2\left(\frac{A}{W} + c\right)$$

c=$^1/_2$ [Welford 1968] or c=1 [MacKenzie 1992]

Schmidt

$$W_e = a + b\frac{A}{T}$$

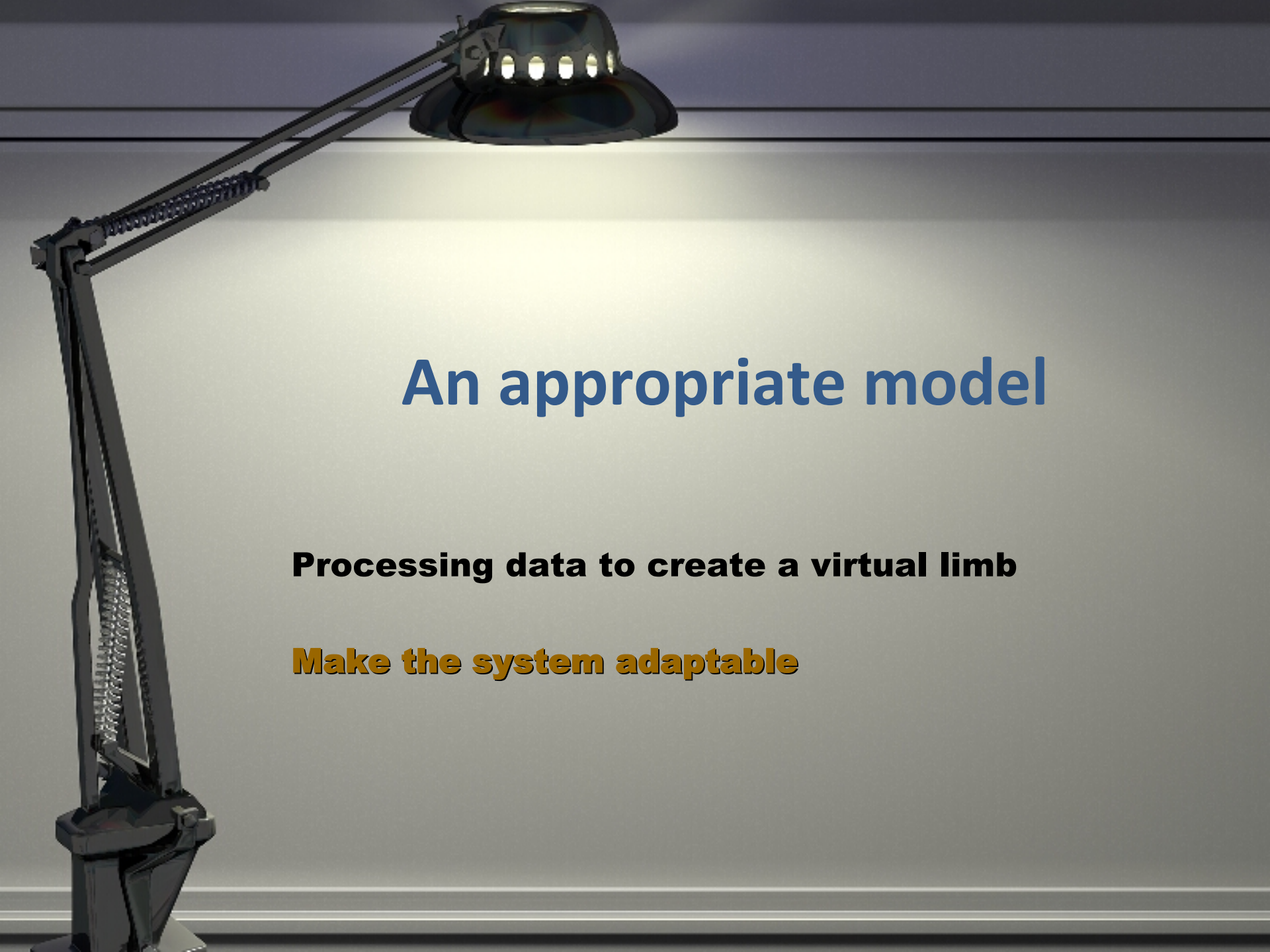
Empirical knowledge

The principle of isochronie established by Freeman (1914)

$$V = kA^{\alpha}$$

Meyer *et al.*

$$T = a + b.n\left(\frac{A}{W}\right)^{1/n}$$

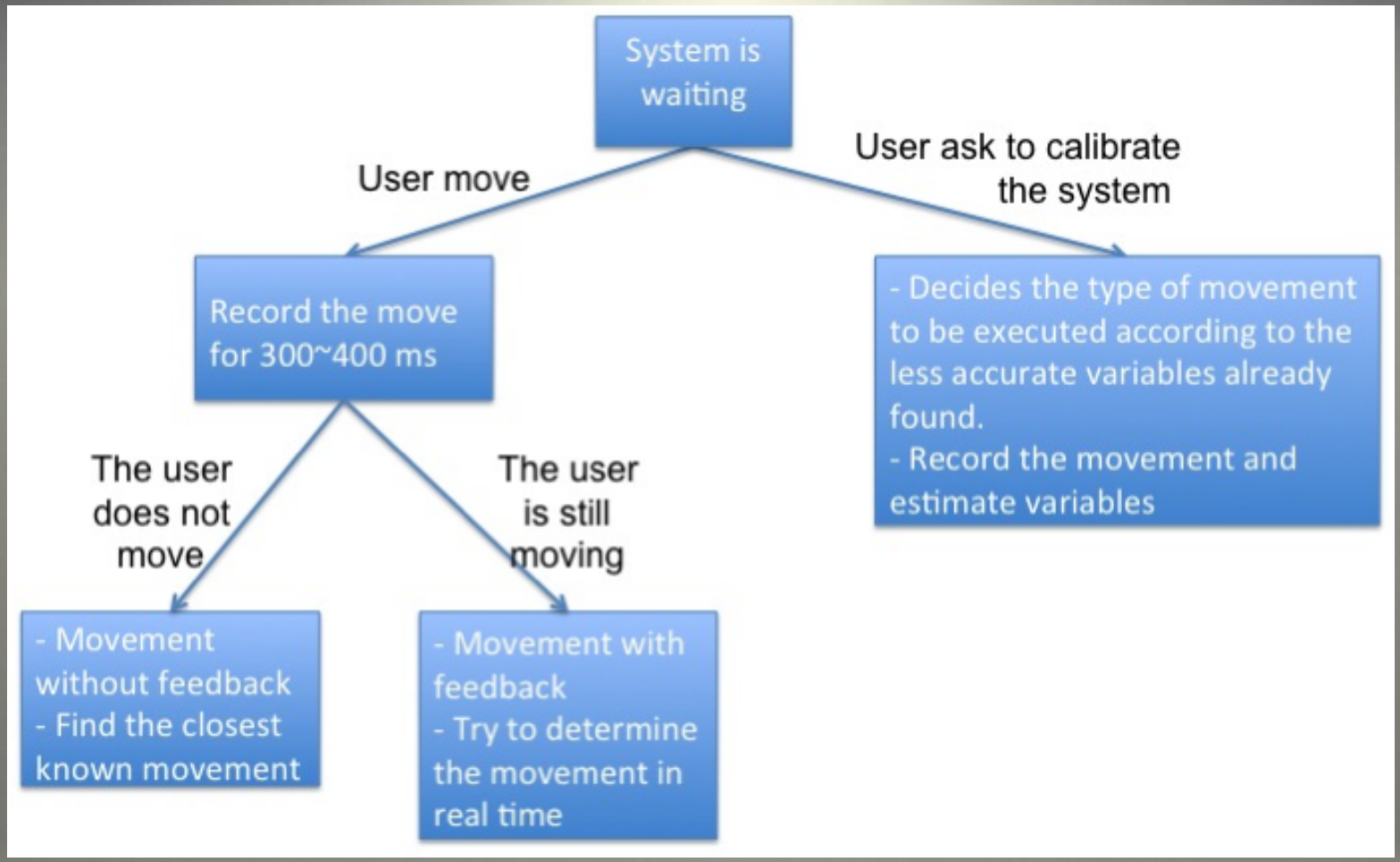

An appropriate model
Processing data to create a virtual limb
Make the system adaptable

For a specific user

2 types of motion :

- The user is asked to perform a specified move in order to calibrate the system
- The user is actually using the system

For a specific user

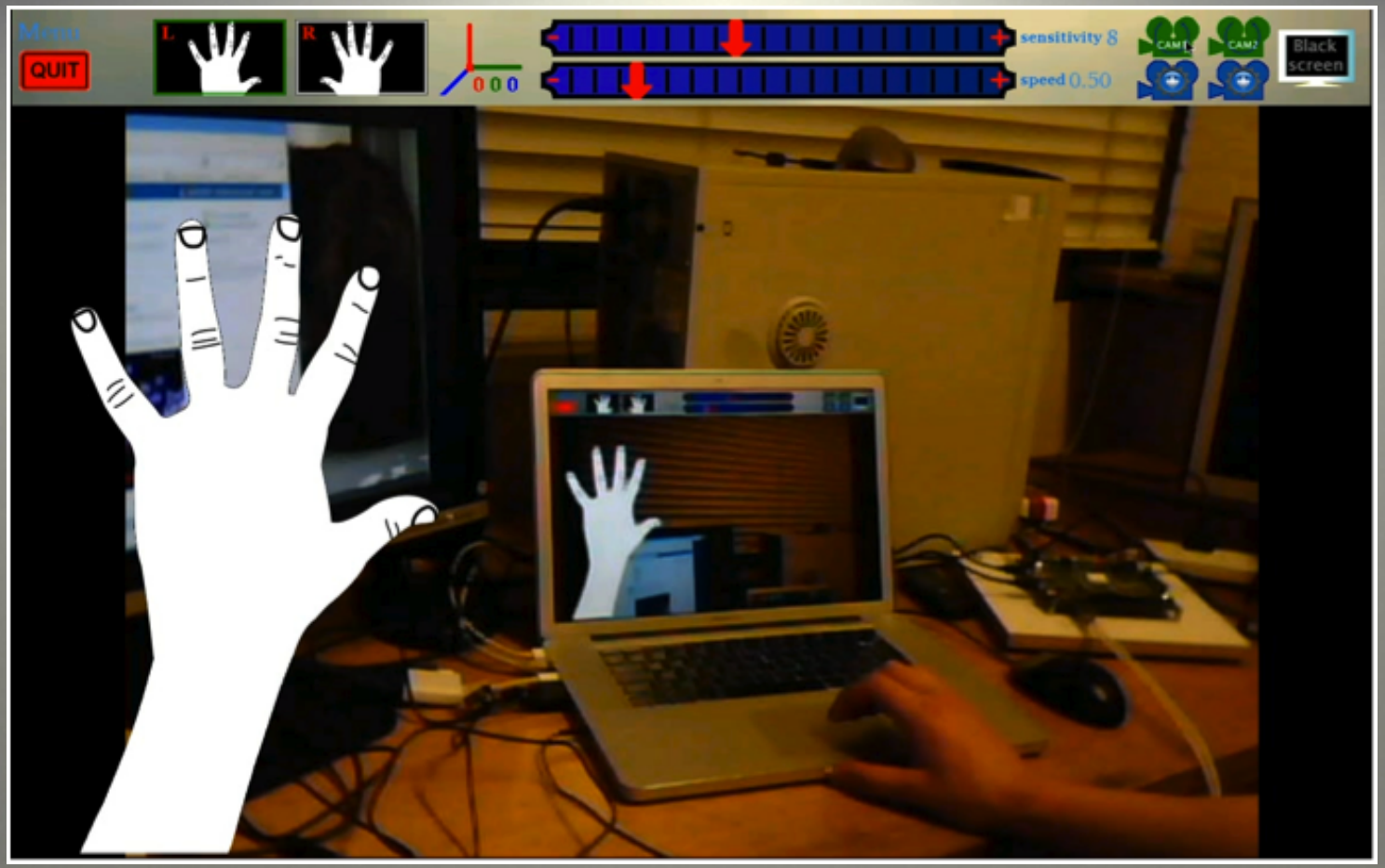


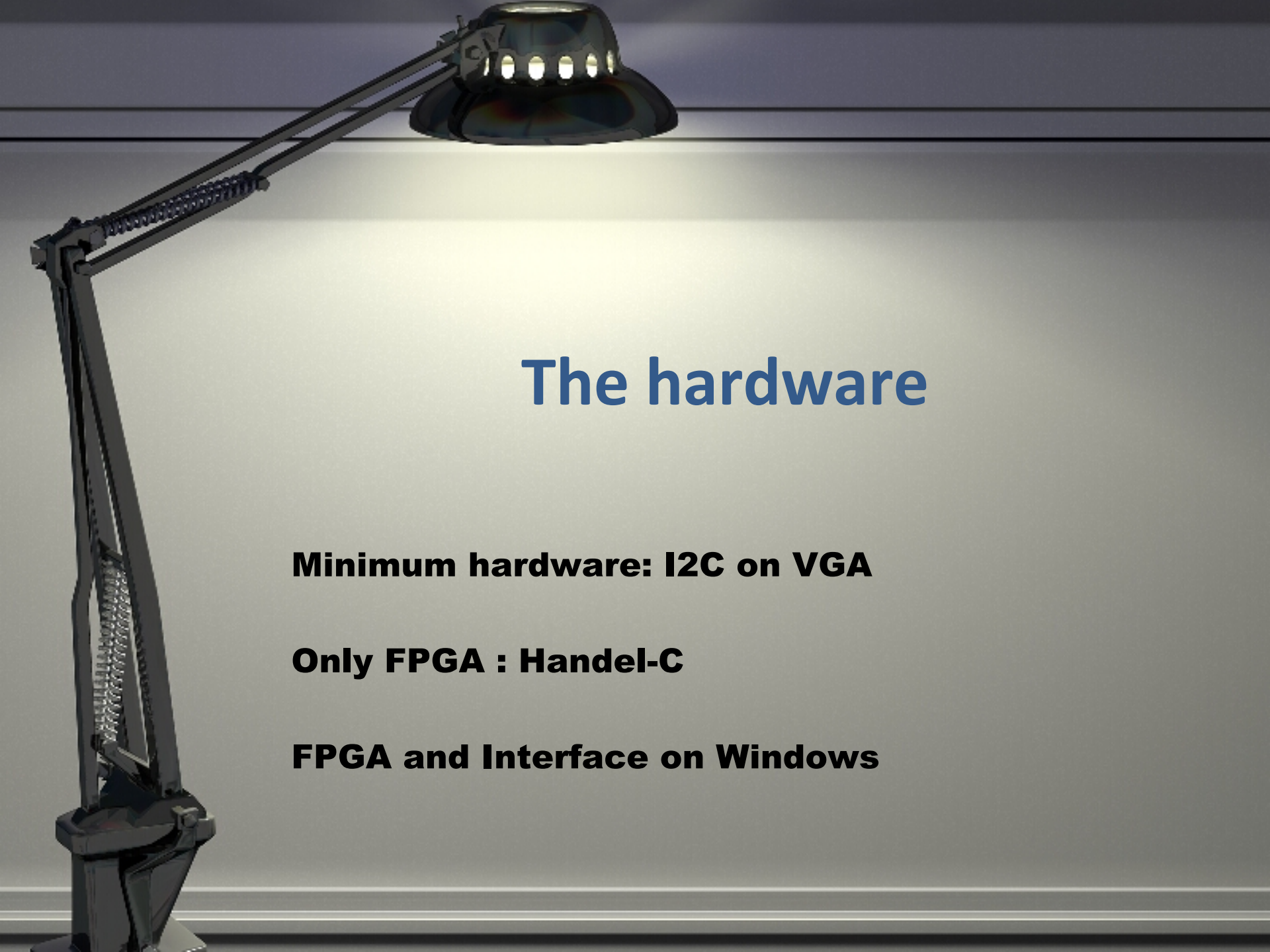
To different users and environments

The system must be able to adapt whatever the reference user

- Depending on the pathology
  - Which limb to model
- Depending on the environment
  - Take a picture of the background

Overview


Menu
QUIT
L
R
0 0 0
sensitivity 8
speed 0.50
CAM1
CAM2
Black screen

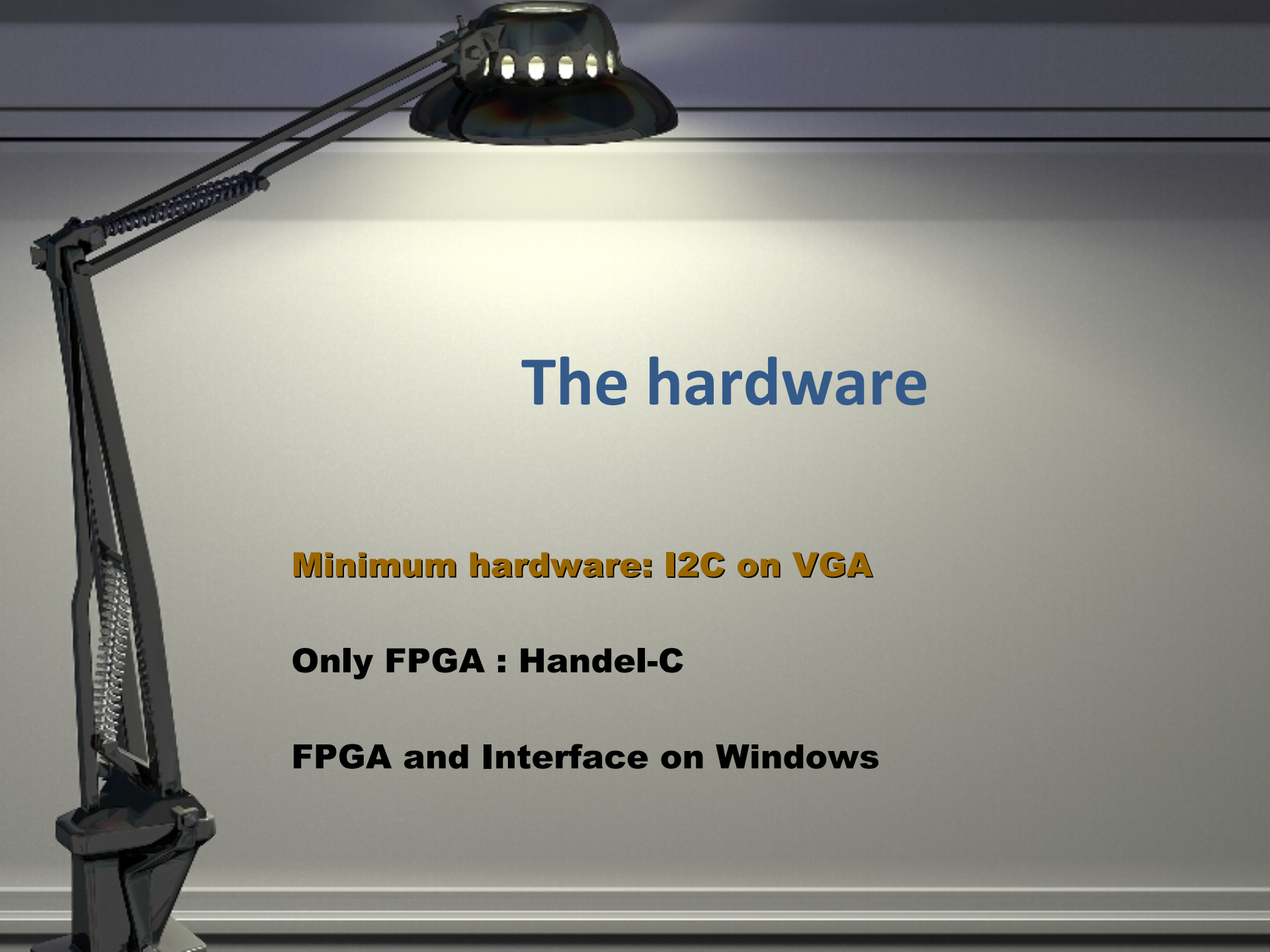


The hardware

Minimum hardware: I2C on VGA

Only FPGA : Handel-C

FPGA and Interface on Windows



The hardware

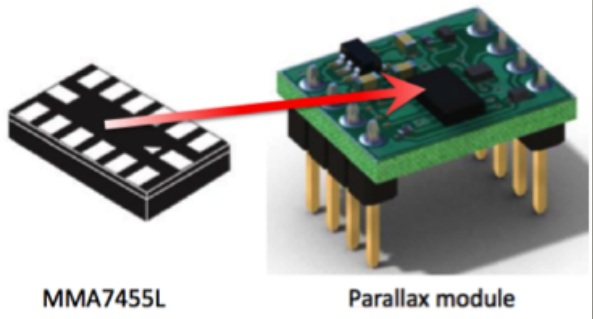
Minimum hardware: I2C on VGA

Only FPGA : Handel-C

FPGA and Interface on Windows

Only one accelerometer

- Some human movement can be quick ~50ms
- Measure position and speed at high frequency



Relevant information about the accelerometer	
Name	MMA7455L
Axes	3 axes: X, Y, Z
Communication protocol	I2C or SPI
SPI frequency	8MHz
Selectable range	±2g ; ±4g ; ±8g
Voltage	2.4V – 3.6V

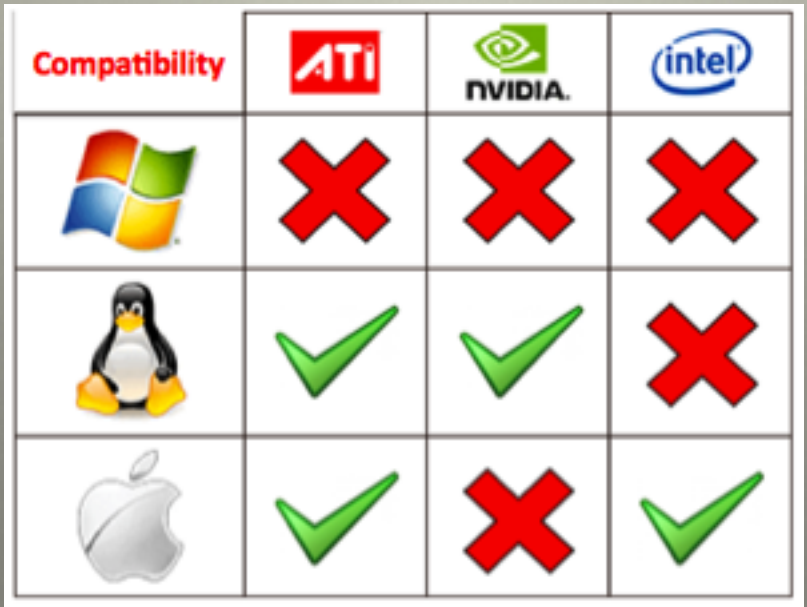
25¢ I2C Adapter Project

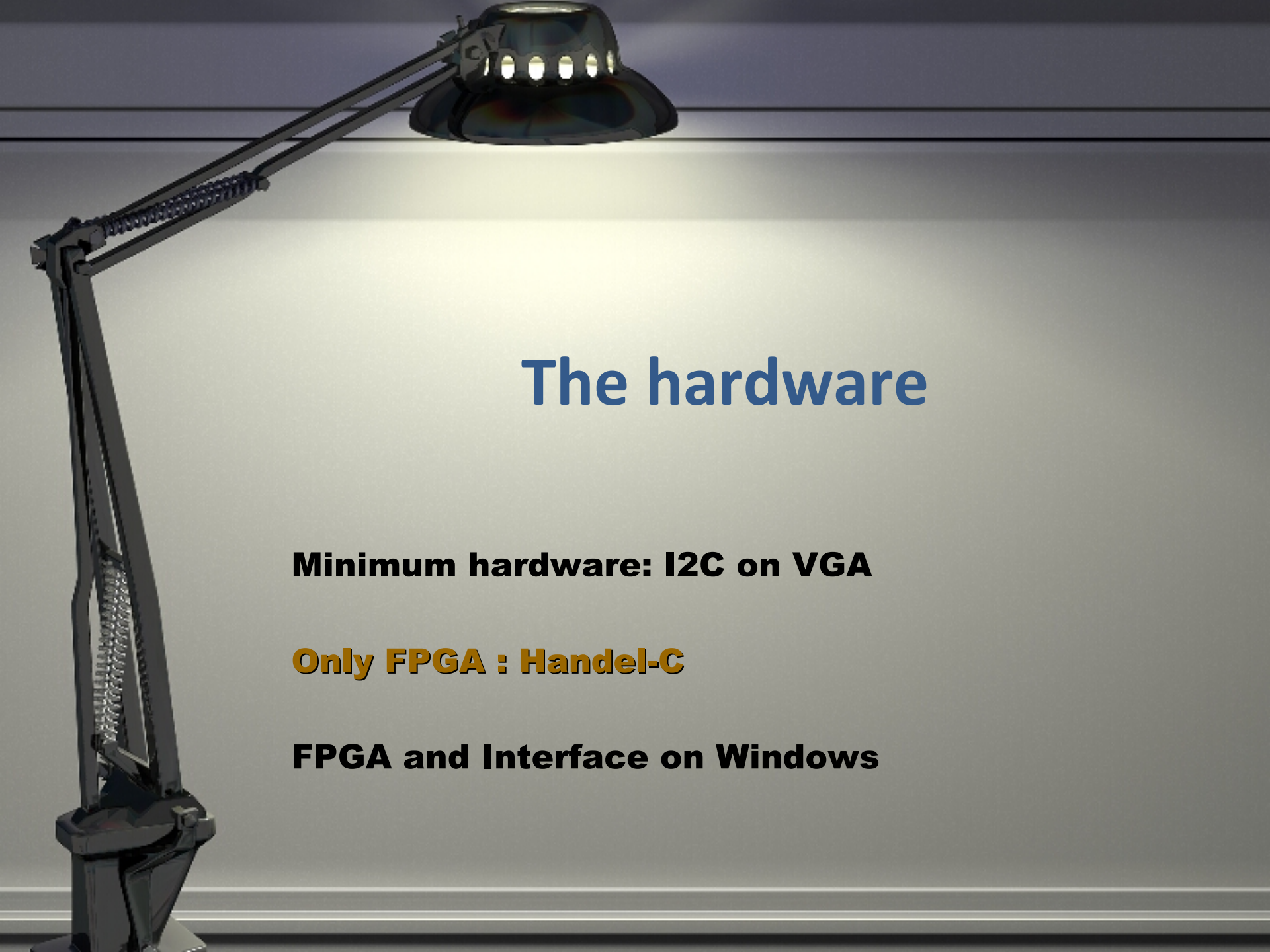
- Plug-and-play system
- Only needs a 0,5£ connector
- Use the controls lines present on many graphics card to interface I2C device

VGA wire	I2C wire
PIN 5 : Ground	Ground
PIN 9 : +5V	VIN
PIN 12 : Data	SDA
PIN 15 : Clock	SCL

Limitations

- Compatibility range is limited
- Likely warranty-voiding operation
- I was unable to get data from MMA7455





The hardware

Minimum hardware: I2C on VGA

Only FPGA : Handel-C

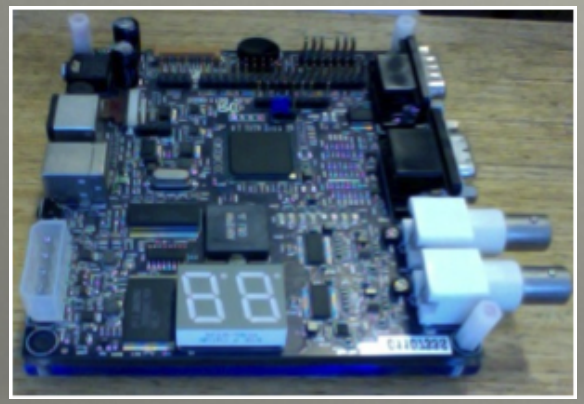
FPGA and Interface on Windows

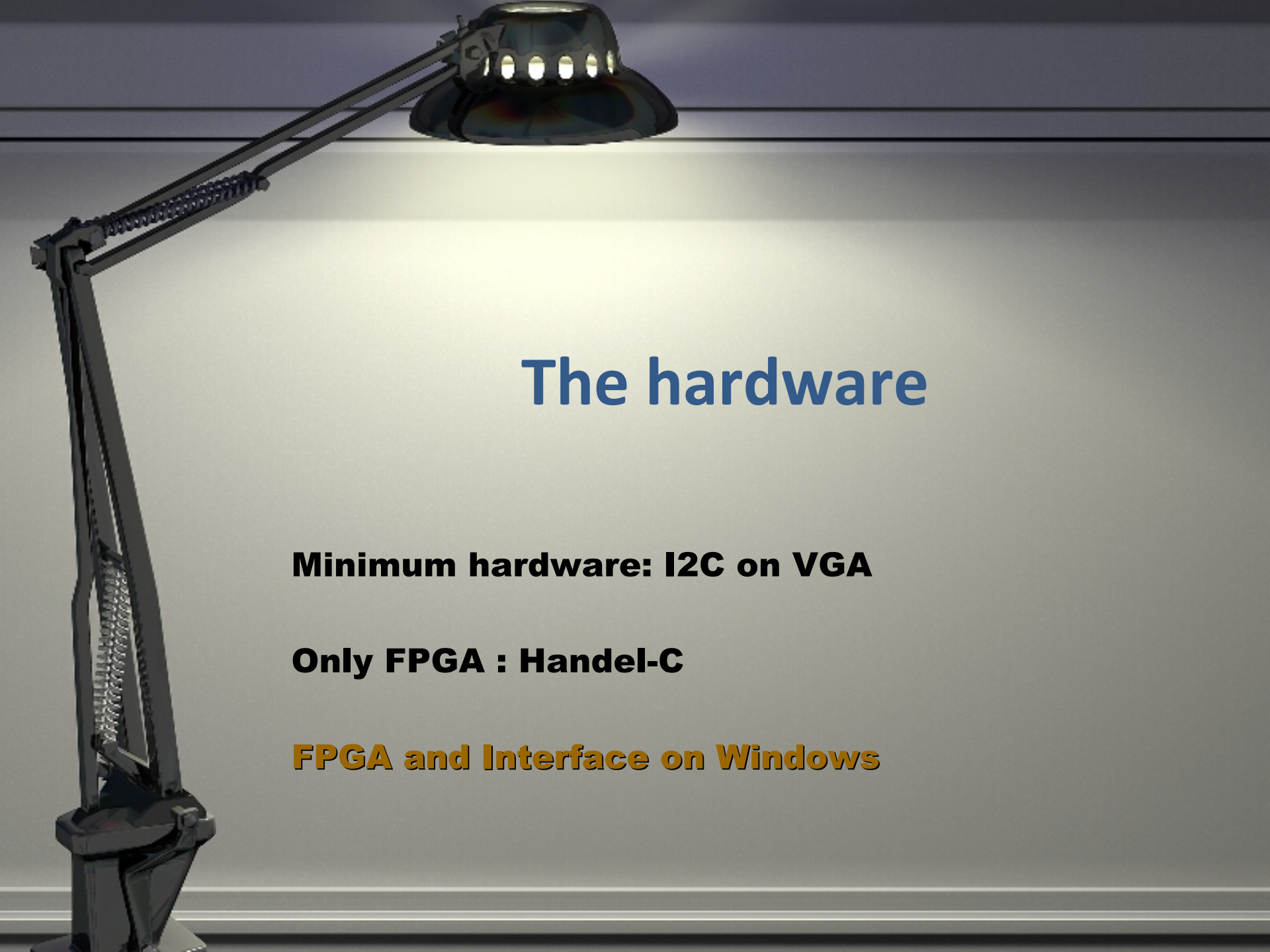
Celoxica RC10

+ Start from the work already done

- Celoxica doesn't support this board any more

- Need to create an elaborate graphic interface





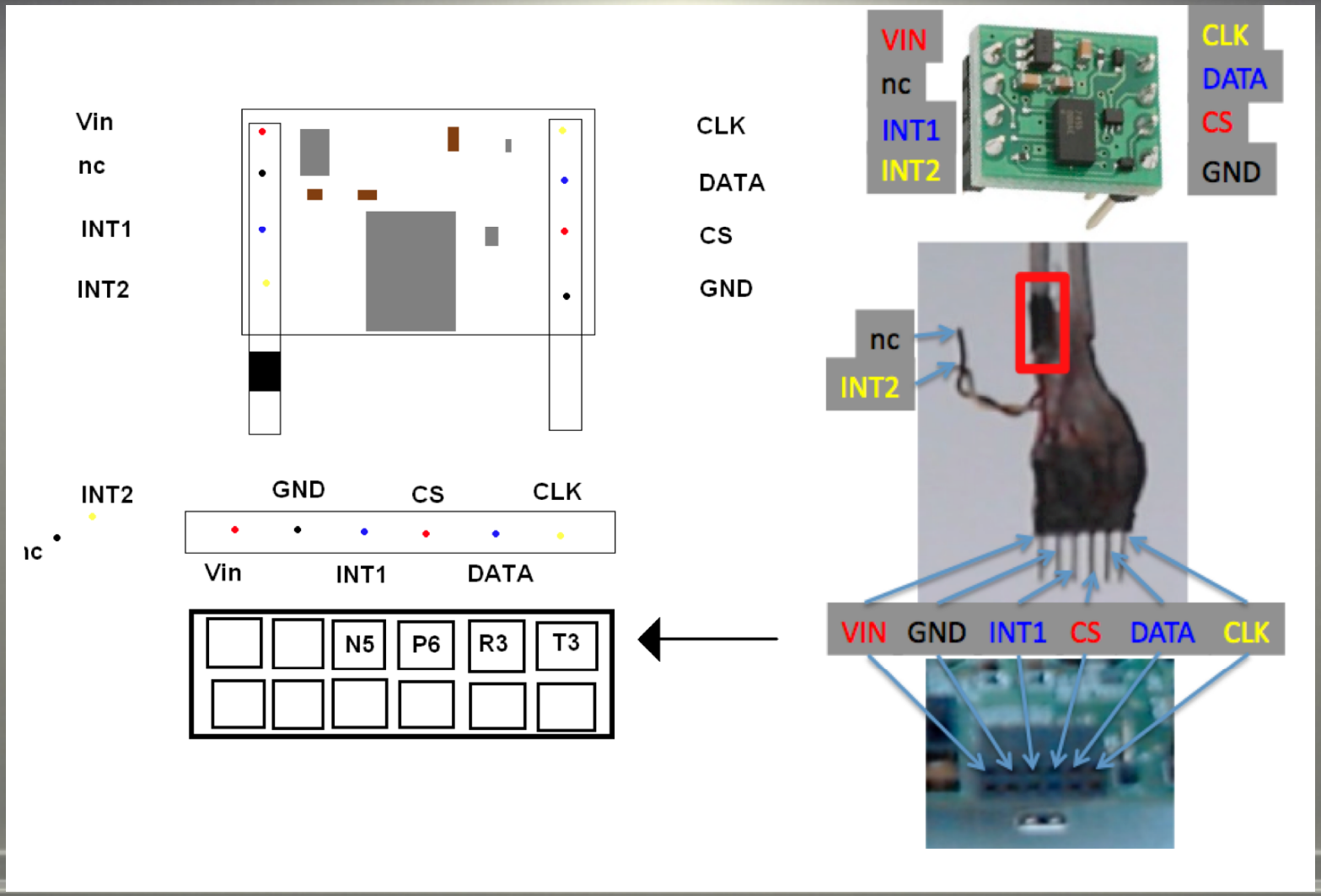
The hardware

Minimum hardware: I2C on VGA

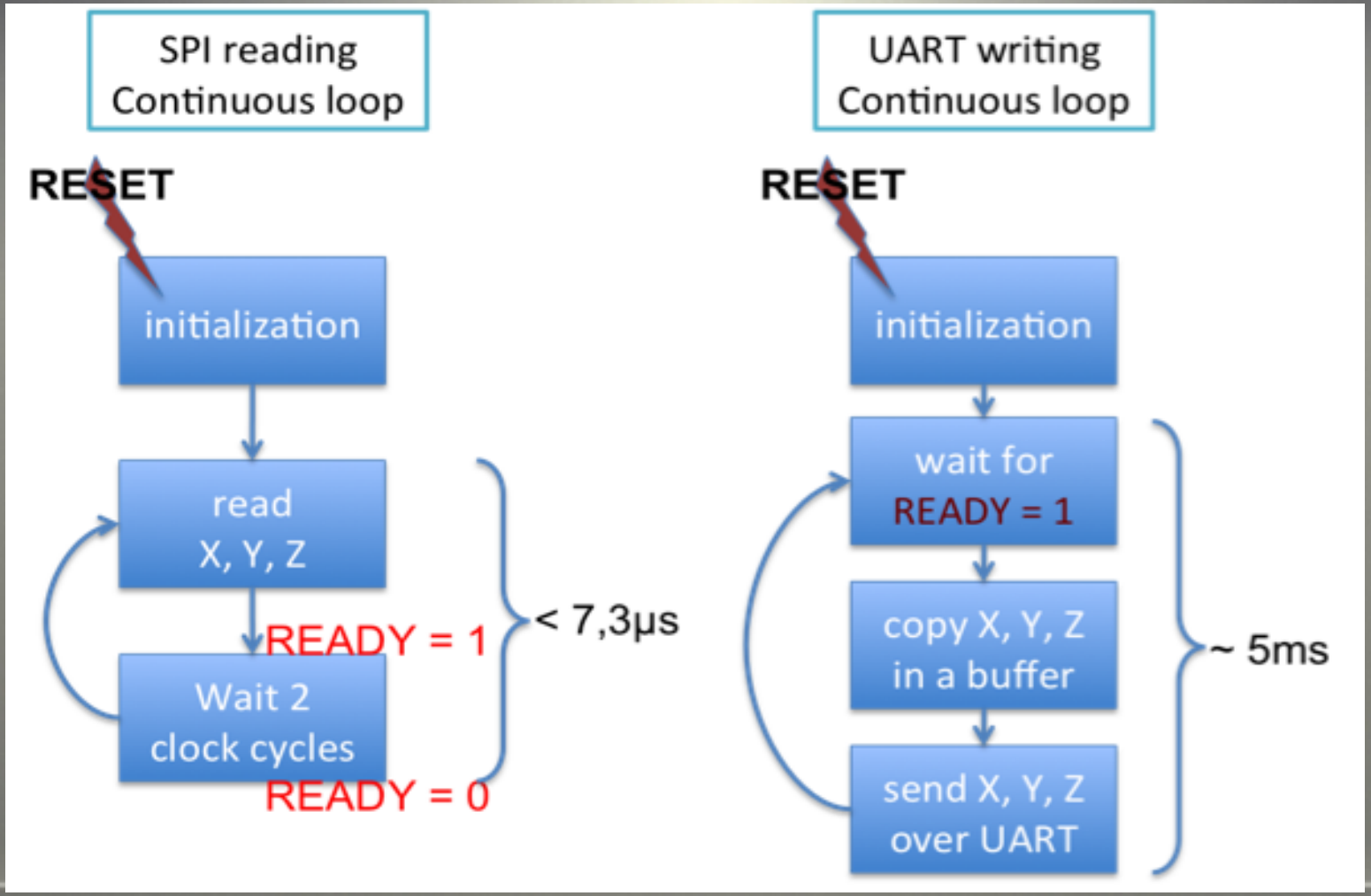
Only FPGA : Handel-C

FPGA and Interface on Windows

Acquire data : SPI is faster than I2C



Acquire data and sending then to the computer

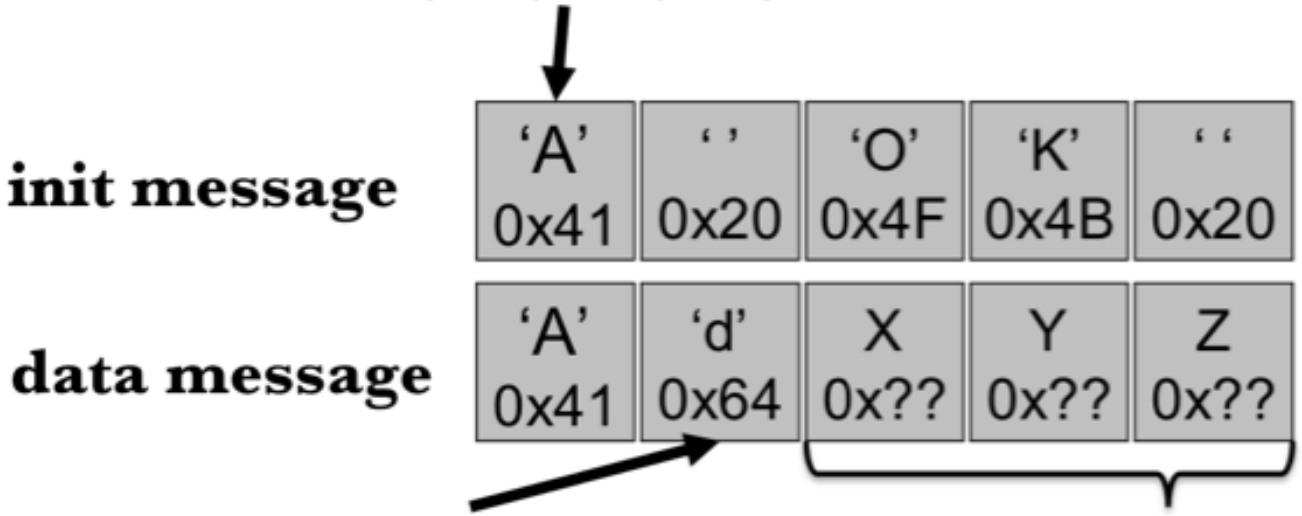


Sending data to the computer

- Chip XR21V1410 embedded on the Atlys is an USB-UART
- 9600 baud, 8 bits data, 1 stop bit, no parity
- Send 5 bytes during each transmission

Sending data to the computer

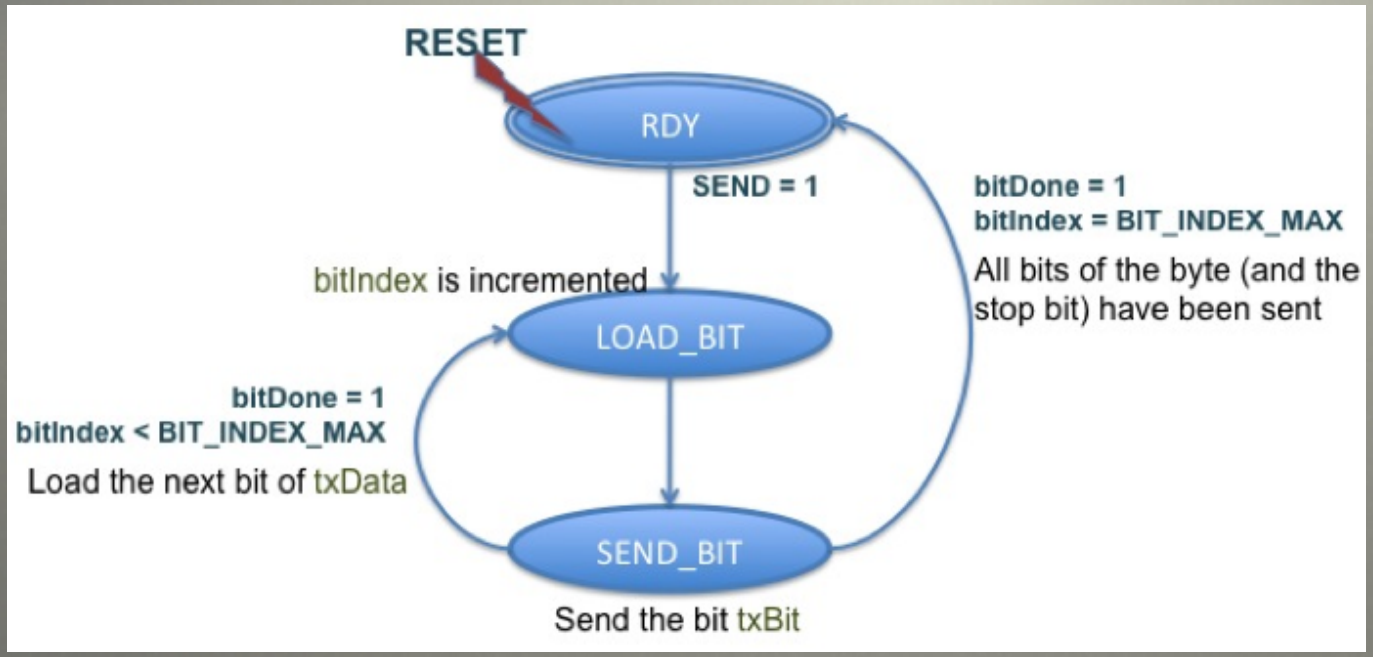
The transmission begins with a known character so appears to be correct. The message is about the sensor A. If several sensors, it could be 'B' (0x42), 'C' (0x43), etc



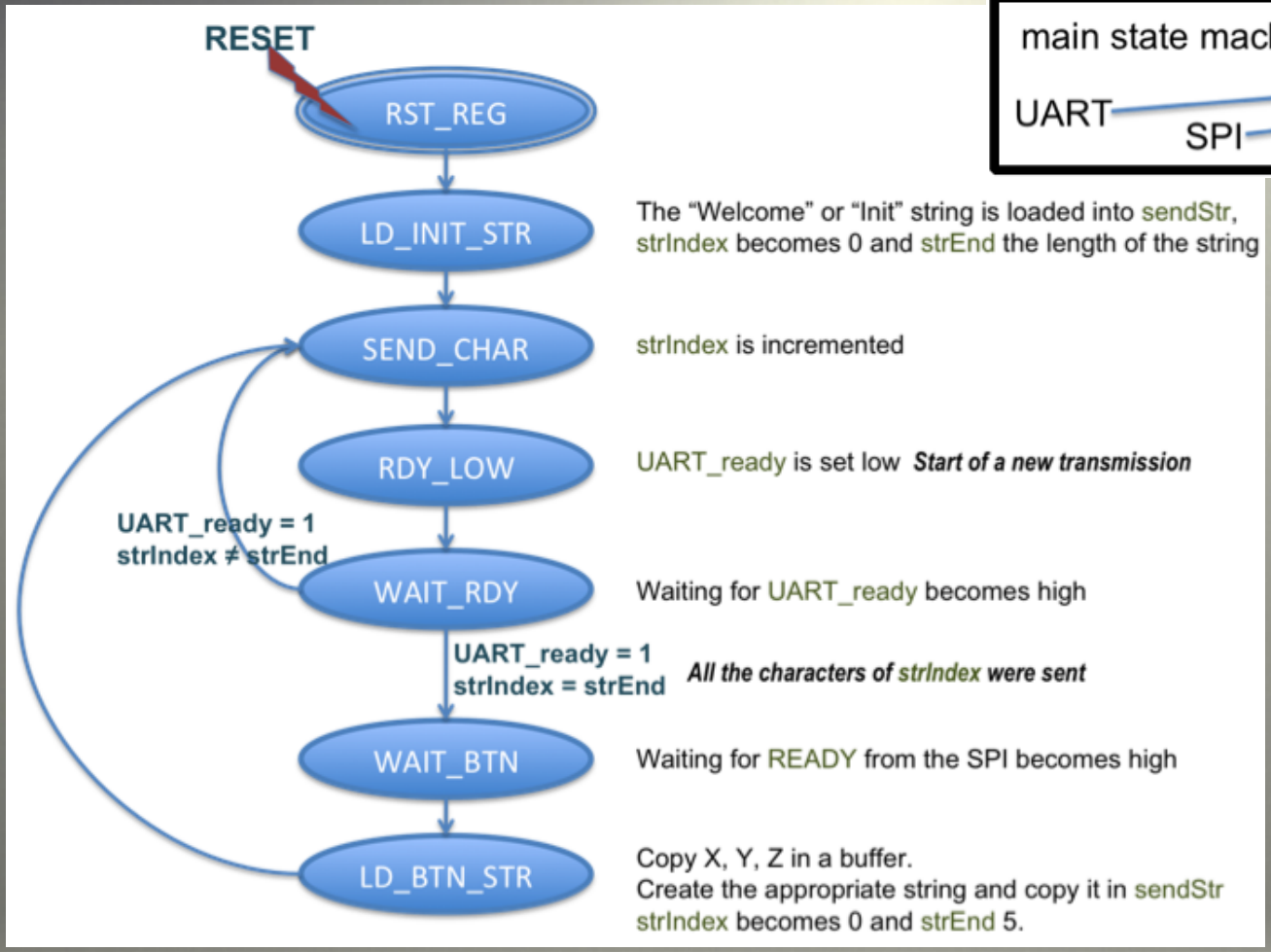
If 'd' (0x64), the message contains data.
If ' ' (0x20), the message contains information
If other, the message is corrupted or is not from the correct source.

Sending data to the computer

UART is operated in “UART_TX_CTRL”

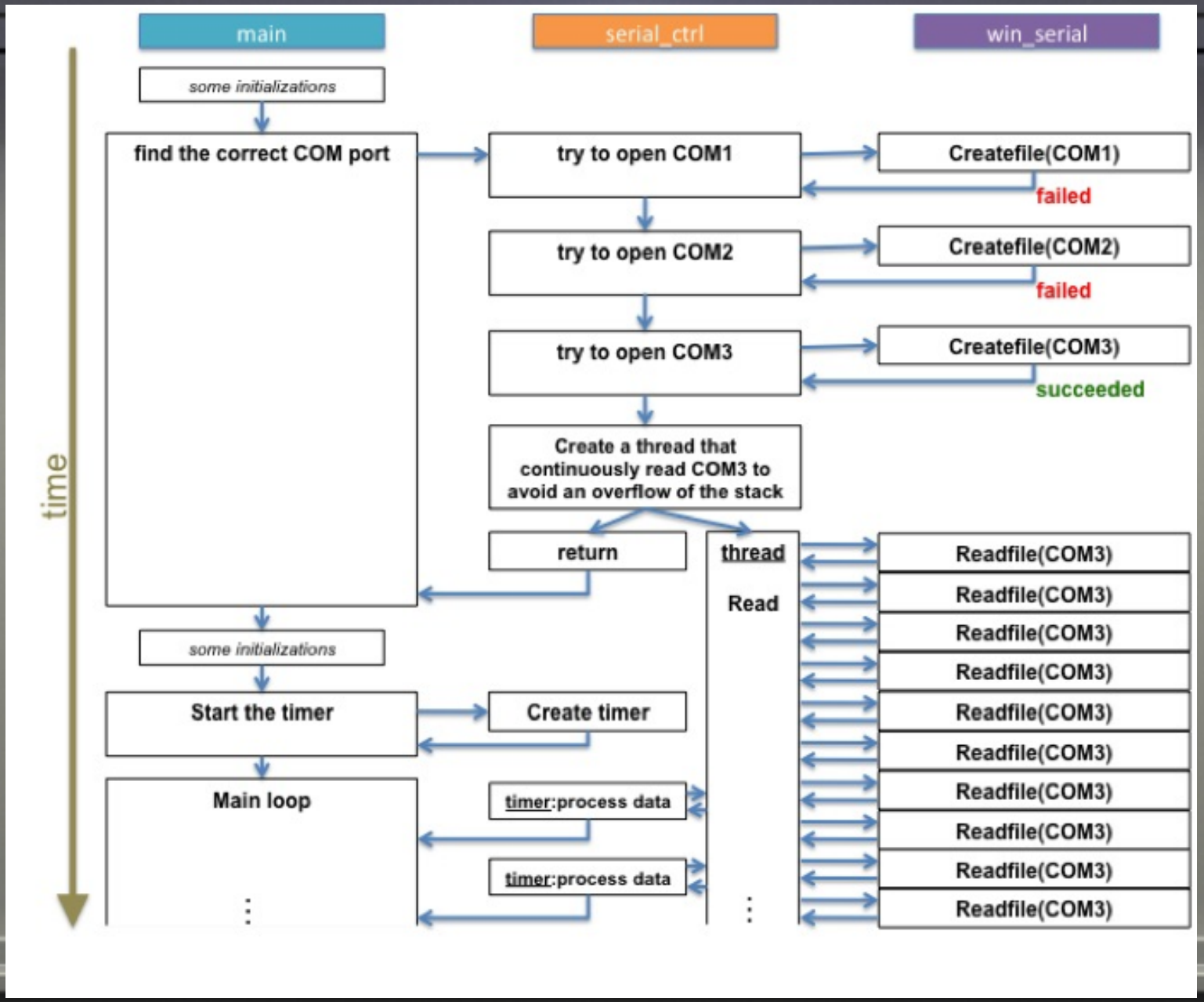


Sending data to the computer



Main state machine "GPIO_demo"

The hardware > FPGA and Interface on Windows

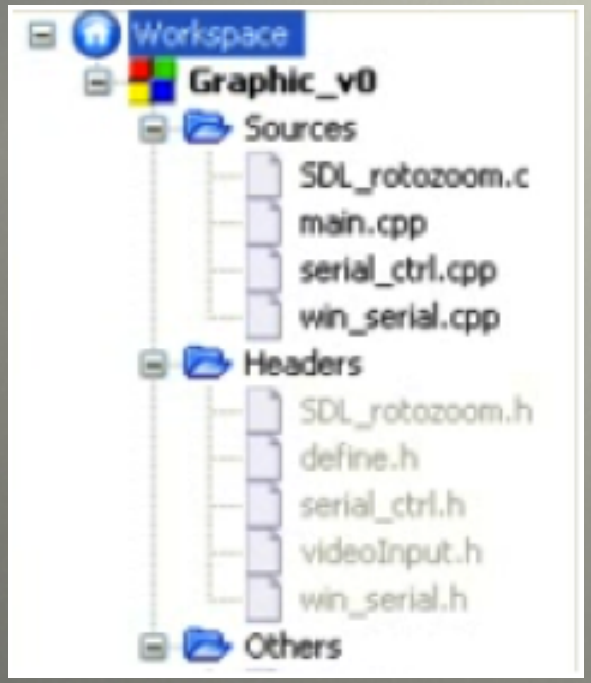


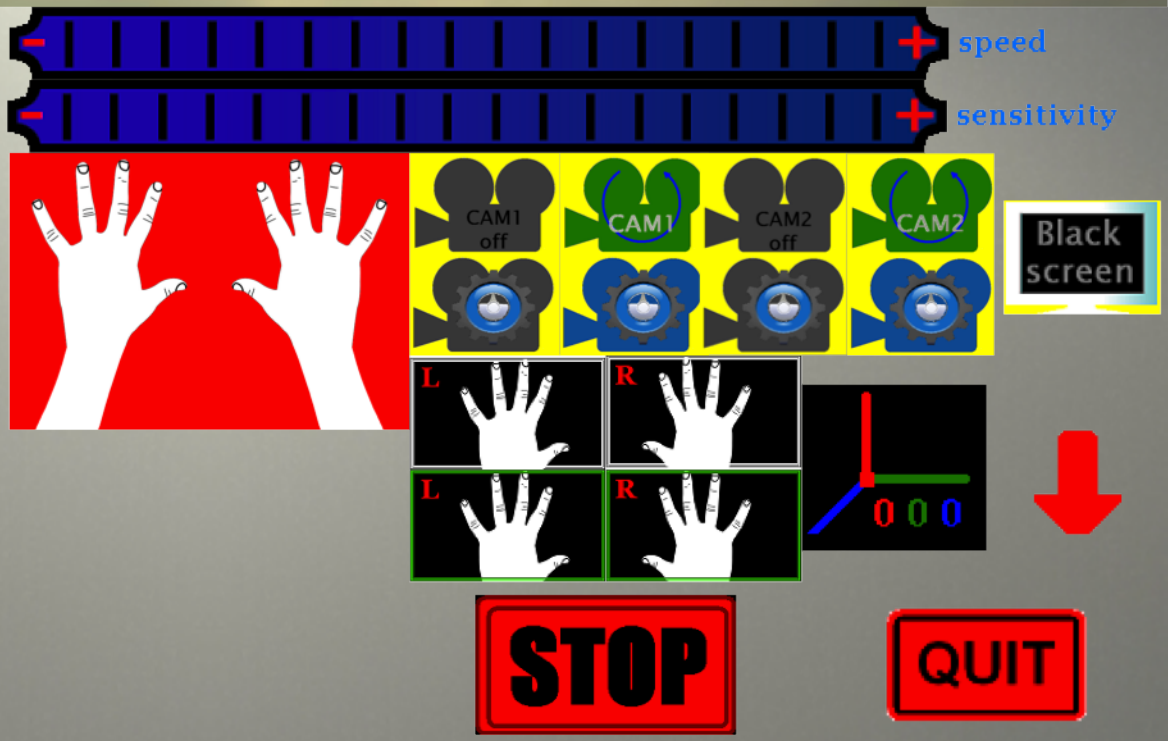
The graphic interface

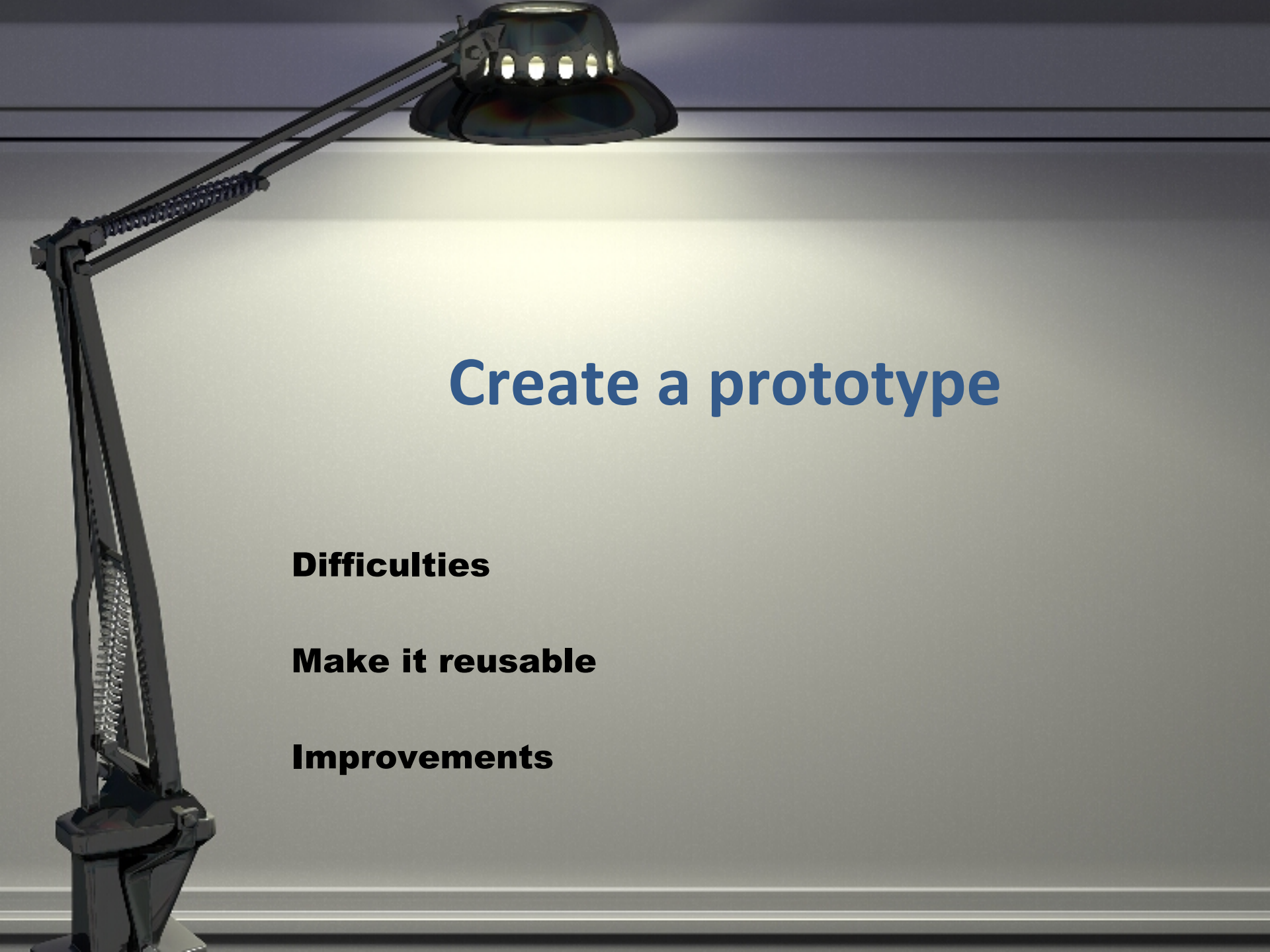
- Simple and user friendly interface

- Development environment : MinGW
- IDE : Code::Blocks
  - Graphics library : SDL
    - SDL_ttf
    - SDL_gfx
  - Capture from webcam : VideoInput

The graphic interface







Create a prototype

Difficulties

Make it reusable

Improvements

Create a prototype

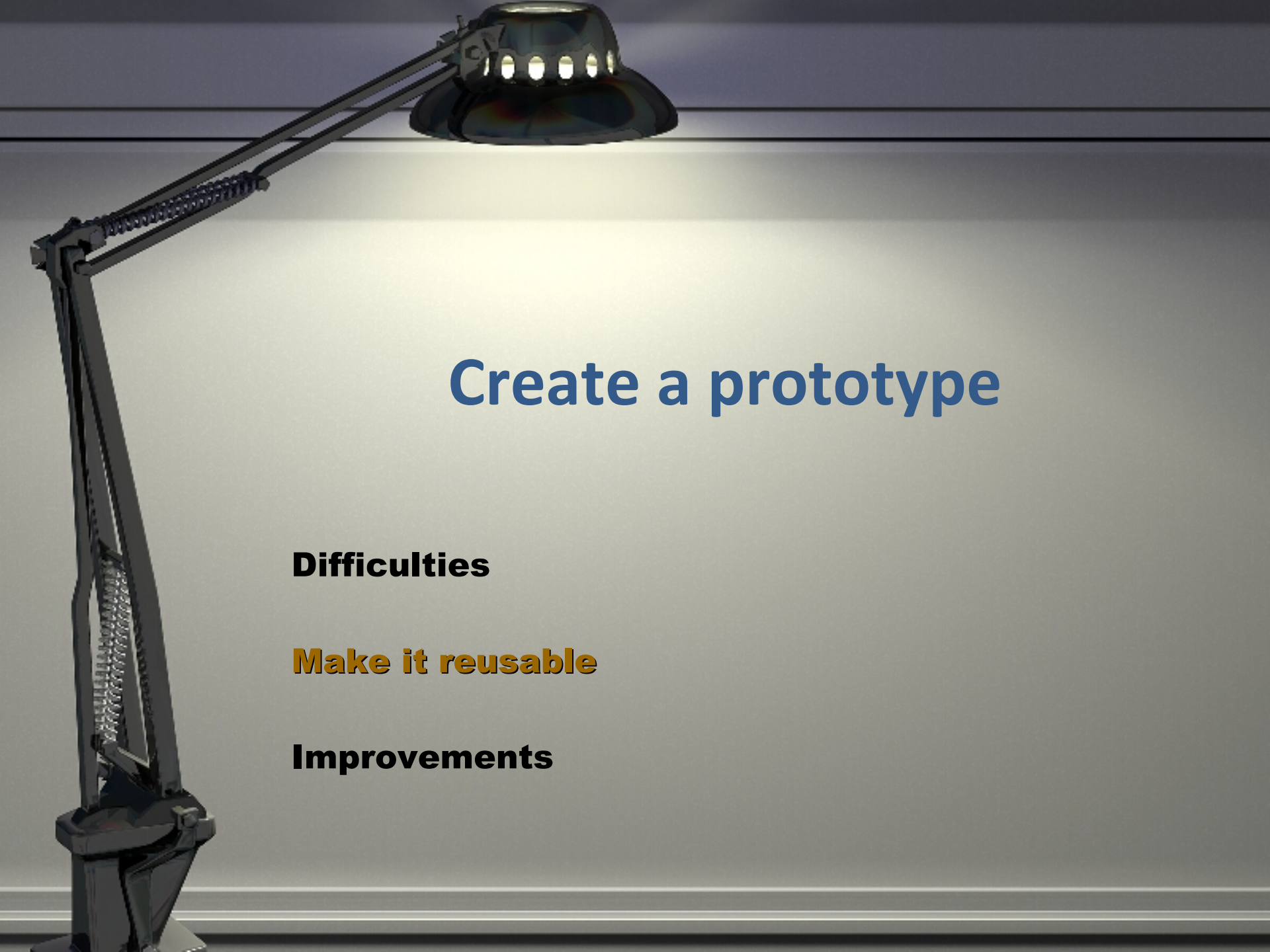
Difficulties

Make it reusable

Improvements

Technical and managerial

- Use a single pin of the FPGA both as an input and a output
  - Use an *inout* I/O
  - Create a signal that determines the direction
- Manage my time
  - Set goals and achieve them as much as possible
  - Limit my objectives to not try developing something that I would not have time to finish


Create a prototype
Difficulties
Make it reusable
Improvements

How ?

- Comments in the code
- All constants in the define.h
- Useful information to retrieve my work in the Technical Manual
- Last versions and resources on plp.antoinem.com

Create a prototype

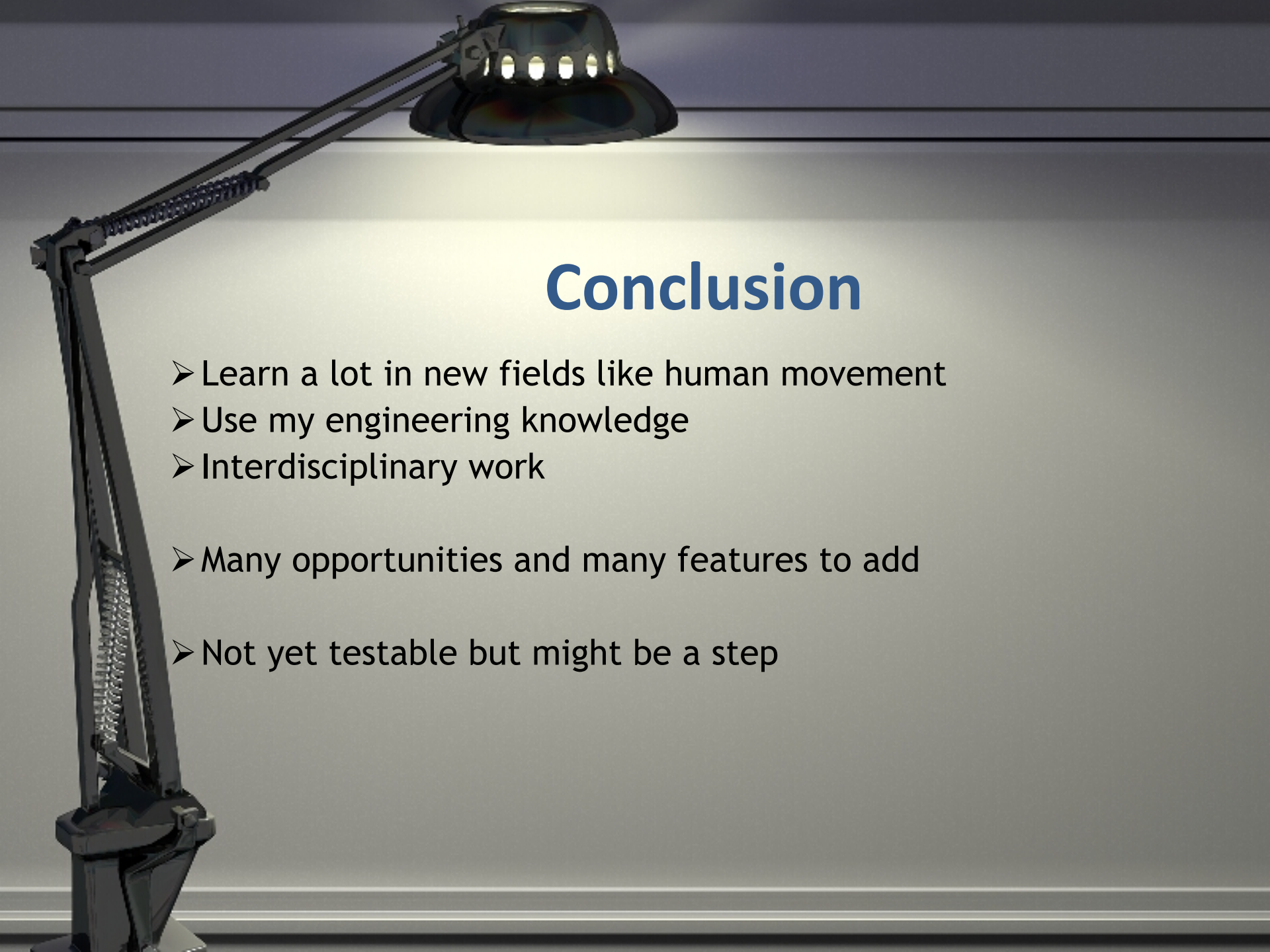
Difficulties

Make it reusable

Improvements

Possibilities of improvement

- Replace the FPGA by a simple micro-controller
- Add a different sensor like a gyroscope
- Make it entirely cross-platform
- Rely a lot more on the equations
  - Develop scenarios to help the user calibrate the system

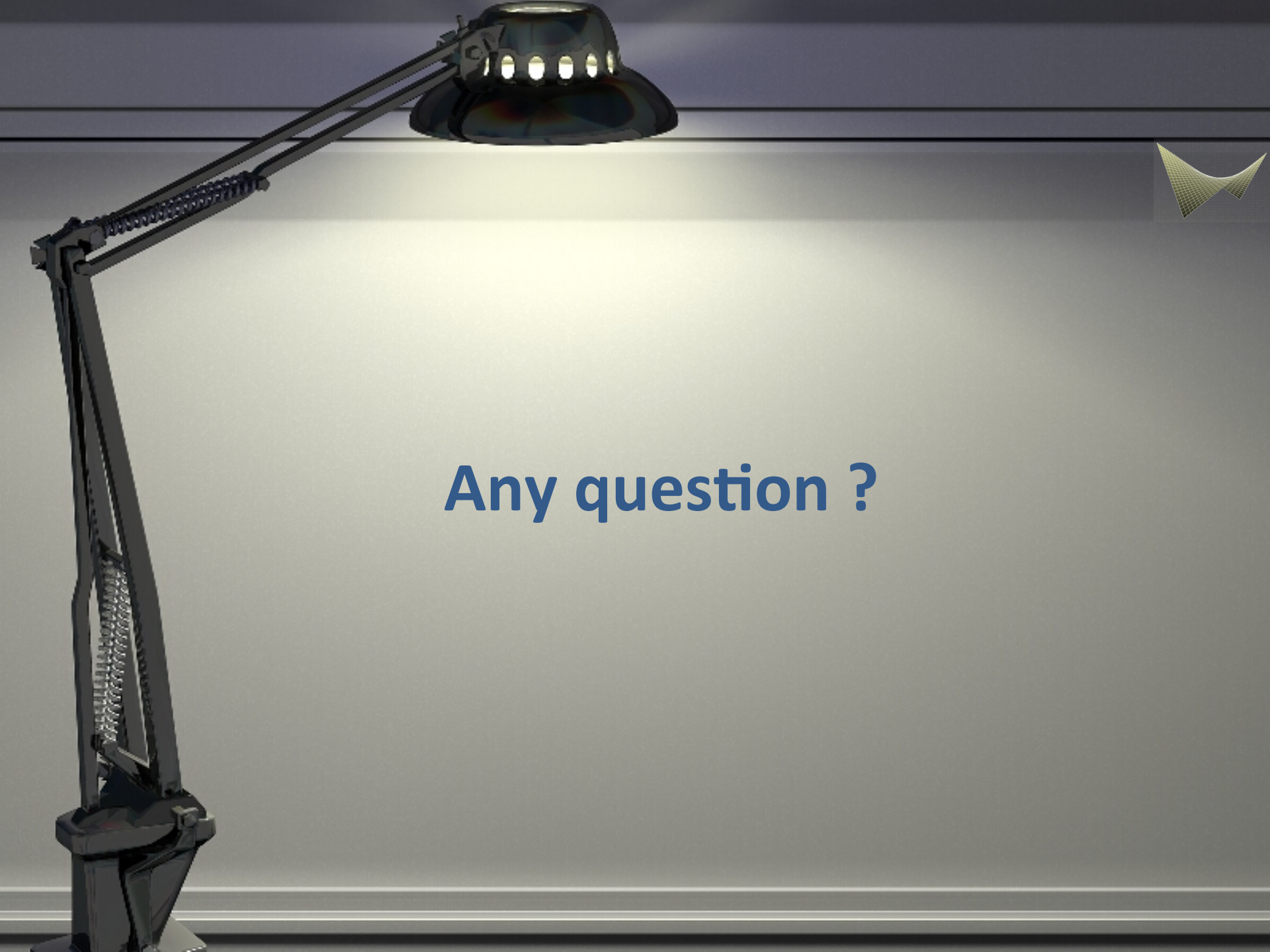


Conclusion

- Learn a lot in new fields like human movement
- Use my engineering knowledge
- Interdisciplinary work

- Many opportunities and many features to add

- Not yet testable but might be a step


Any question ?