

# Melody Control

Giacomo Driussi / Philipp Hoffmann  
Interaktion Engineering / WS 2018/2019  
Prof. Dr. Michael Kipp

# Idea

The goal is to create a novel interface for any classic game. Beside the controlling the user also creates his own melody and an unique atmosphere while controlling. This work should explore if the use of an interface with musical feedback entertains more than an common controllers.

# Related Work



## THE SYLPHYO

WIRELESS MIDI  
WIND CONTROLLER

A NEW KIND OF  
MUSICAL INSTRUMENT



AODYO INSTRUMENTS

Processing in Unity



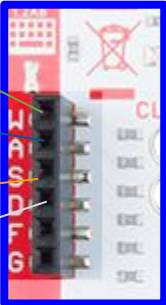
Metallophon



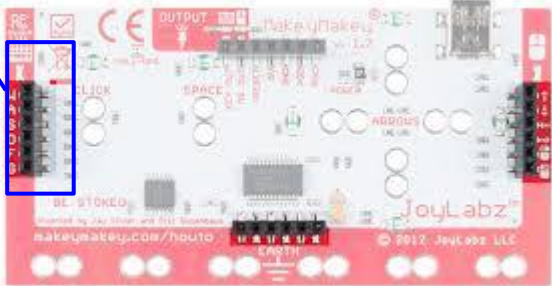
IR-Sensor



Plugged in predefined Key Slots



PC Connection via USB



MaKey MaKey

# Game Development

jump,duck



vertical move group

switch left/right



horizontal move group

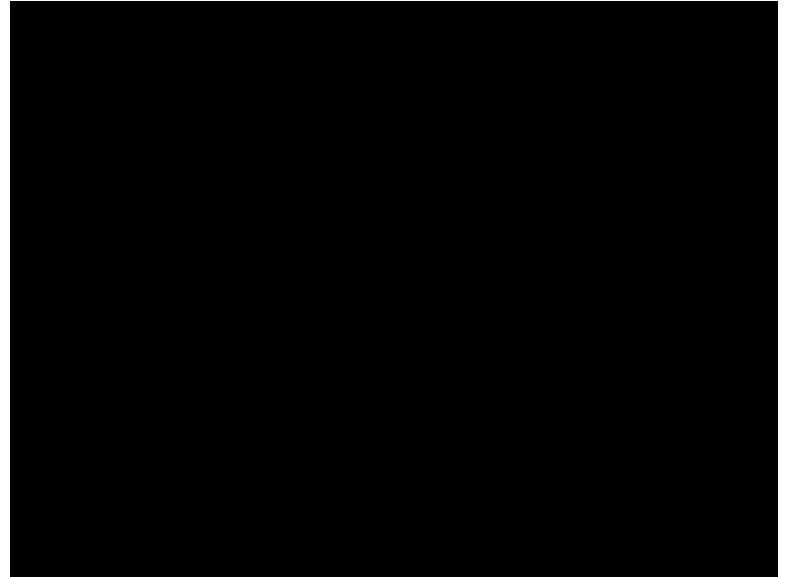
shield



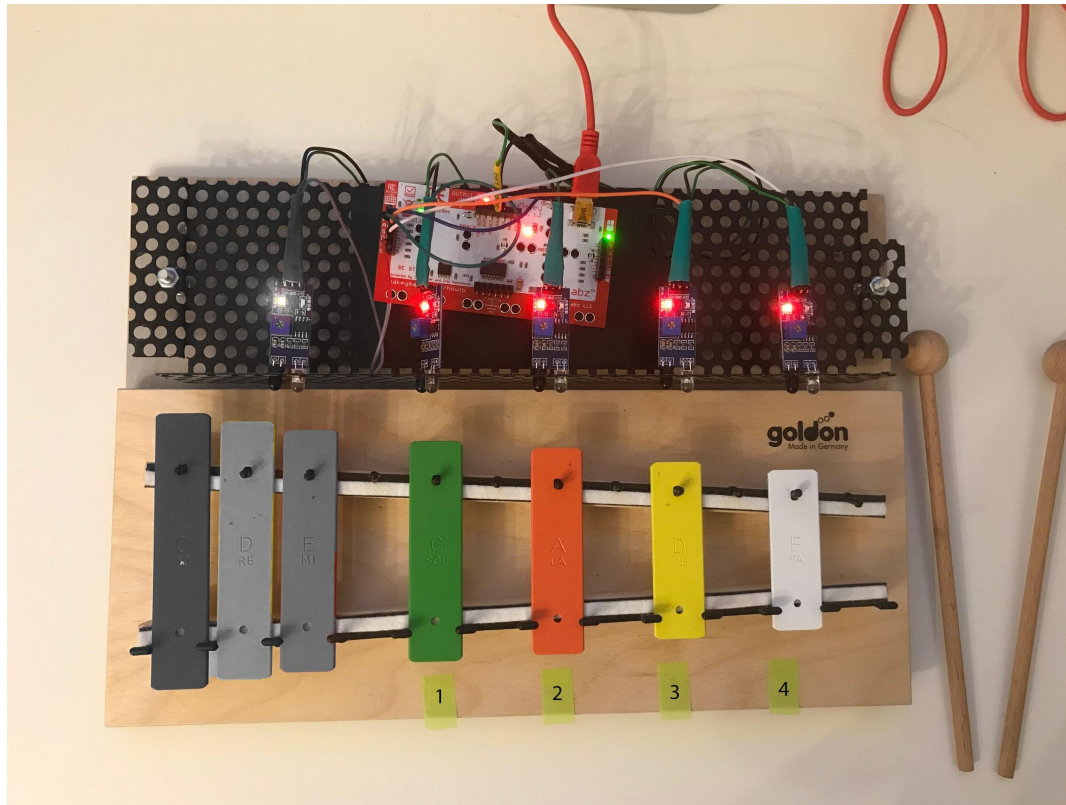
special move

Ignoring Collision, no adding score to the “Hitting Counters”.

In case of collision the player is not stopped/slowed down by the obstacle but the speedbonus is reset to 0.



# Layout



# Controlling Interface V1 : *Less varianz / more strict*

## Interface Rules:

Jumping/Ducking: \*\*

- Only one Combination
- The 2 Highest/Low Notes at the same time

Switching L/R: \*\*\*

- Only one Combination
- Base of two defined Notes plus the highest/lowest to turn right/left

Shield\*\*

- Only one Combination
- Notes with Highest possible Distance at the same time

## Interface:

**Jump:** 3+4

**Duck :** 1+2

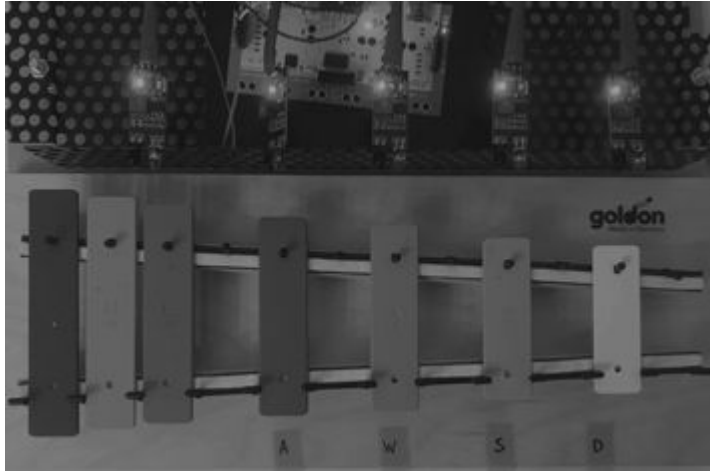
**left:** 2-3-1

**right:** 2-3-4

**shield:** 1+4

# Controlling Interface V1 : *Less varianz / less strict*

Jump: 3+4



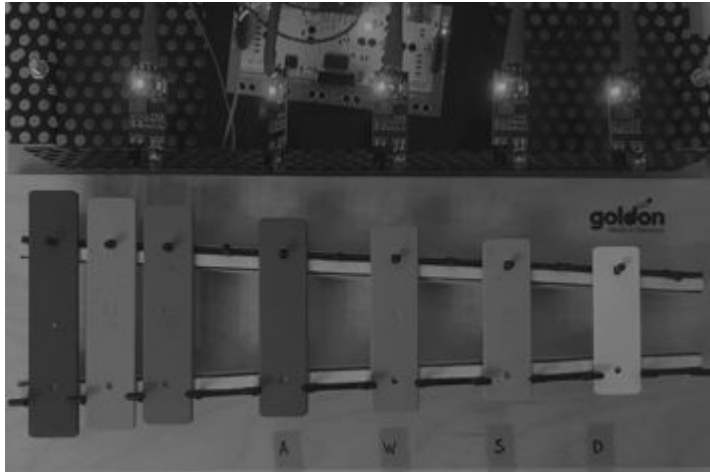
Duck : 1+2





# Controlling Interface V1 : *Less varianz / less strict*

left: 2-3-1

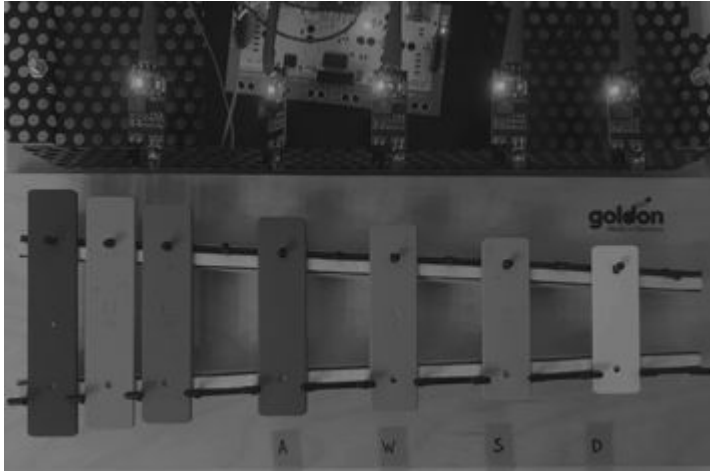


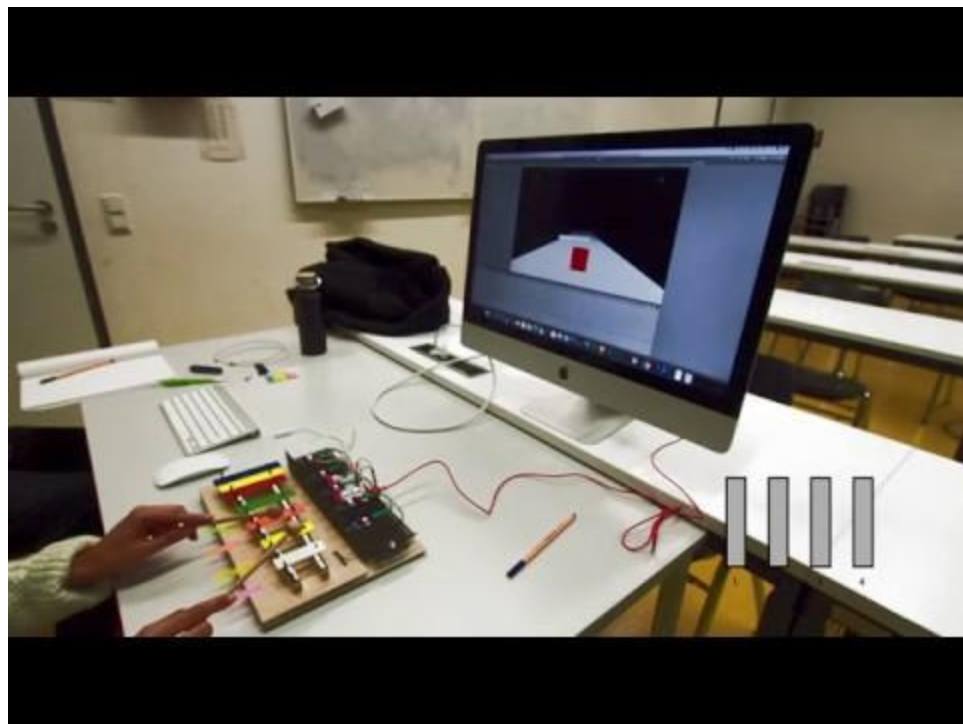
right: 2-3-4



# Controlling Interface V1 : *Less varianz / less strict*

shield: 1+4





# Controlling Interface V2 : *High varianz / High time-consuming / more flexible*

## Interface Rules:

Jumping/Ducking: \*\*\*

- Double Hit to activate
- Over “Jump” one tile up for jumping / down to duck

Switching L/R: \*\*

- Hit the neighbour in the desired direction

Shield\*\*

- Double Hit on two tiles at the same time with the highest distance
- Only one Possibility

## Interface:

**Jump:** 1-1-3 | 2-2-4

**Duck :** 4-4-2 | 3-3-1

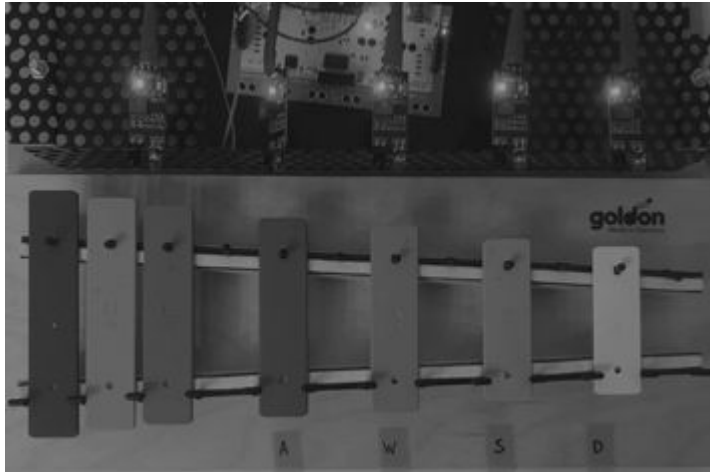
**left:** 4-3 | 3-2 | 2-1

**right:** 1-2 | 2-3 | 3-4

**shield:** 1+4 - 1+4

# Controlling Interface V2 : *High varianz / High time-consuming*

Jump: 1-1-3 | 2-2-4

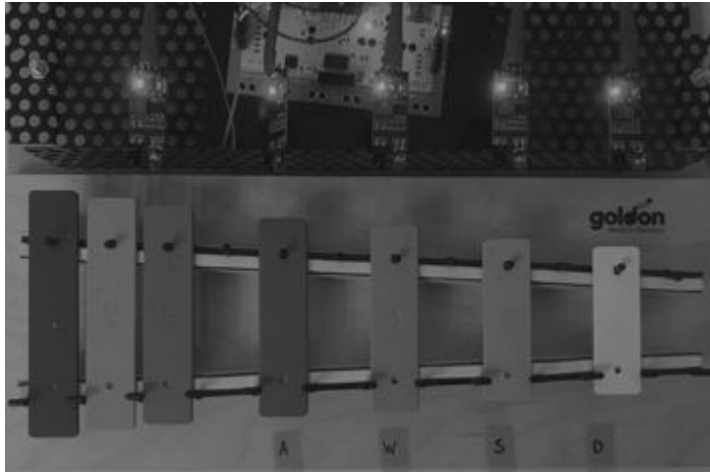


Duck : 4-4-2 | 3-3-1

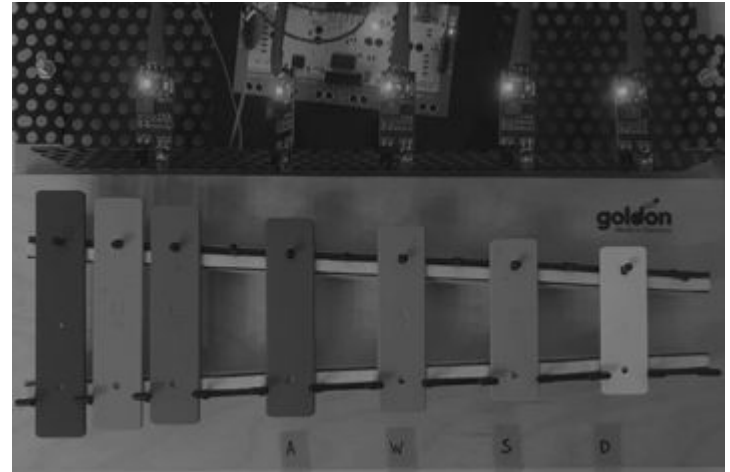


# Controlling Interface V2 : *High varianz / High time-consuming*

left: 4-3 | 3-2 | 2-1



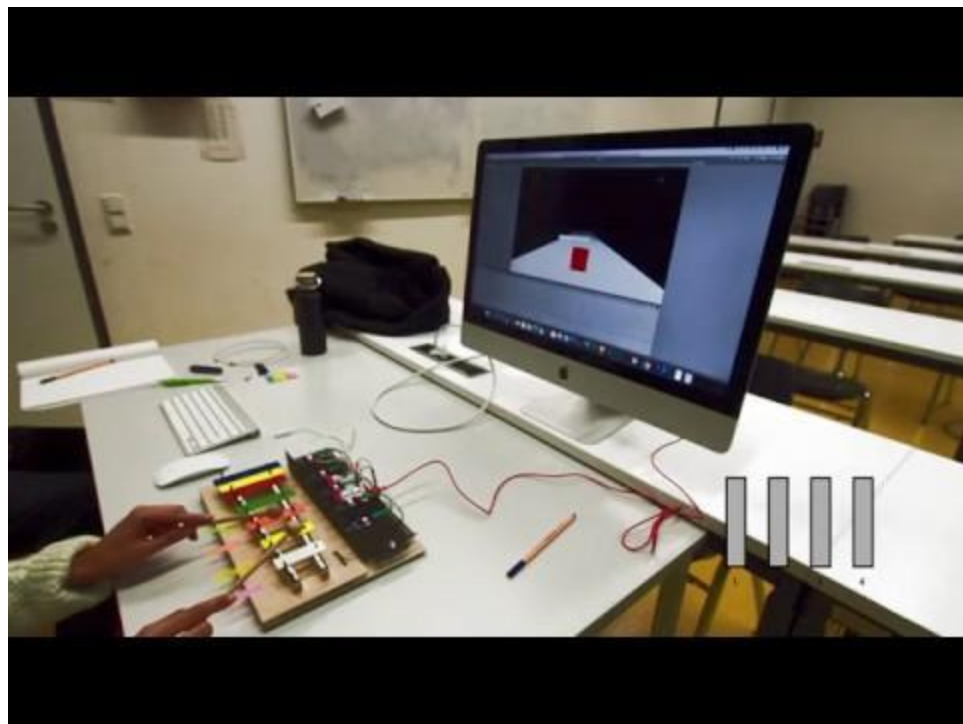
right: 1-2 | 2-3 | 3-4



# Controlling Interface V2 : *High varianz / High time-consuming*

shield: 1+4 - 1+4







# Controlling Interface V3 : *Less varianz / very simple / strict*

## Interface Rules:

Jumping/Ducking: \*

- Hit Notes with the highest distance at the same time

Switching L/R: \*

- The highest/lowest two notes at the same time

Shield\*

- Double Hit on one at the same time

## Interface:

**Jump:** 1+4

**Duck :** 2+3

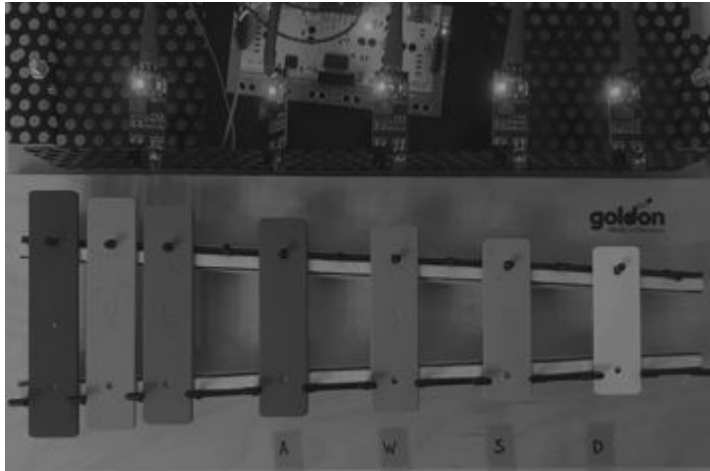
**left:** 1+2

**right:** 3+4

**shield:** 1+1 | 2+2 | 3+3 | 4+4

# Controlling Interface V3 : *Less varianz / very simple / very strict*

Jump: 1+4



Duck : 2+3

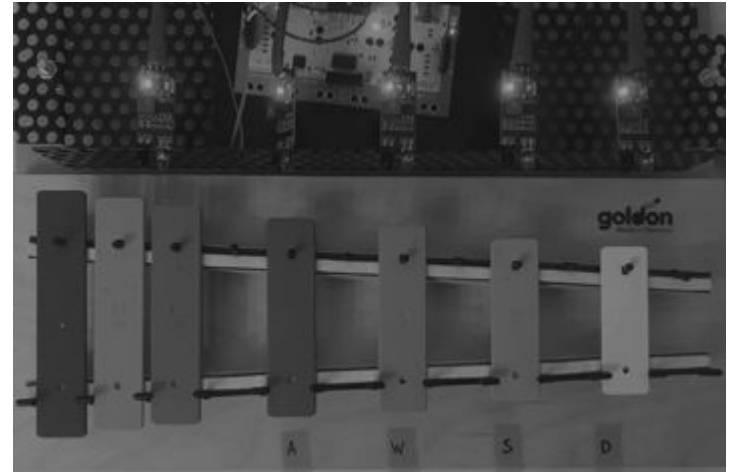


# Controlling Interface V3 : *Less varianz / very simple / very strict*

left: 1+2

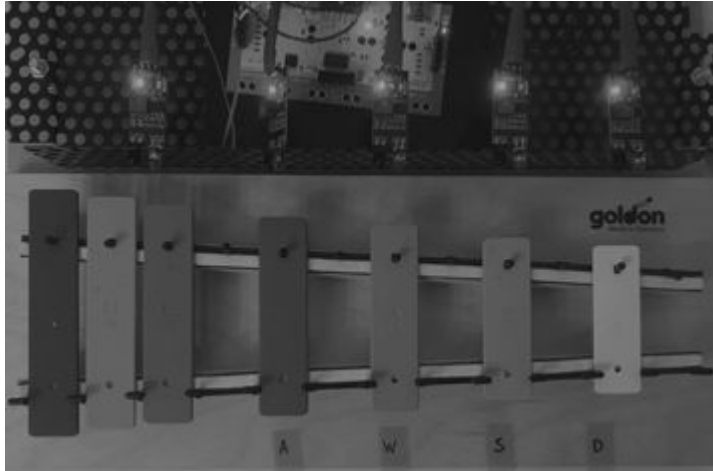


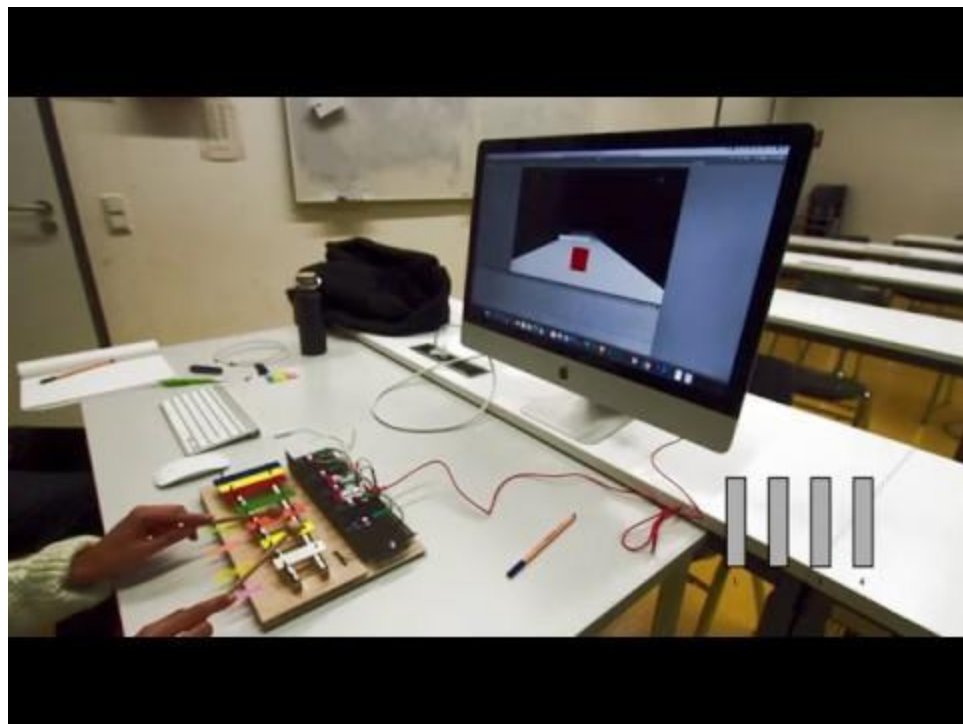
right: 3+4



# Controlling Interface V3 : *Less varianz / very simple / very strict*

shield: 1+1 | 2+2 | 3+3 | 4+4





Thank you