

Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

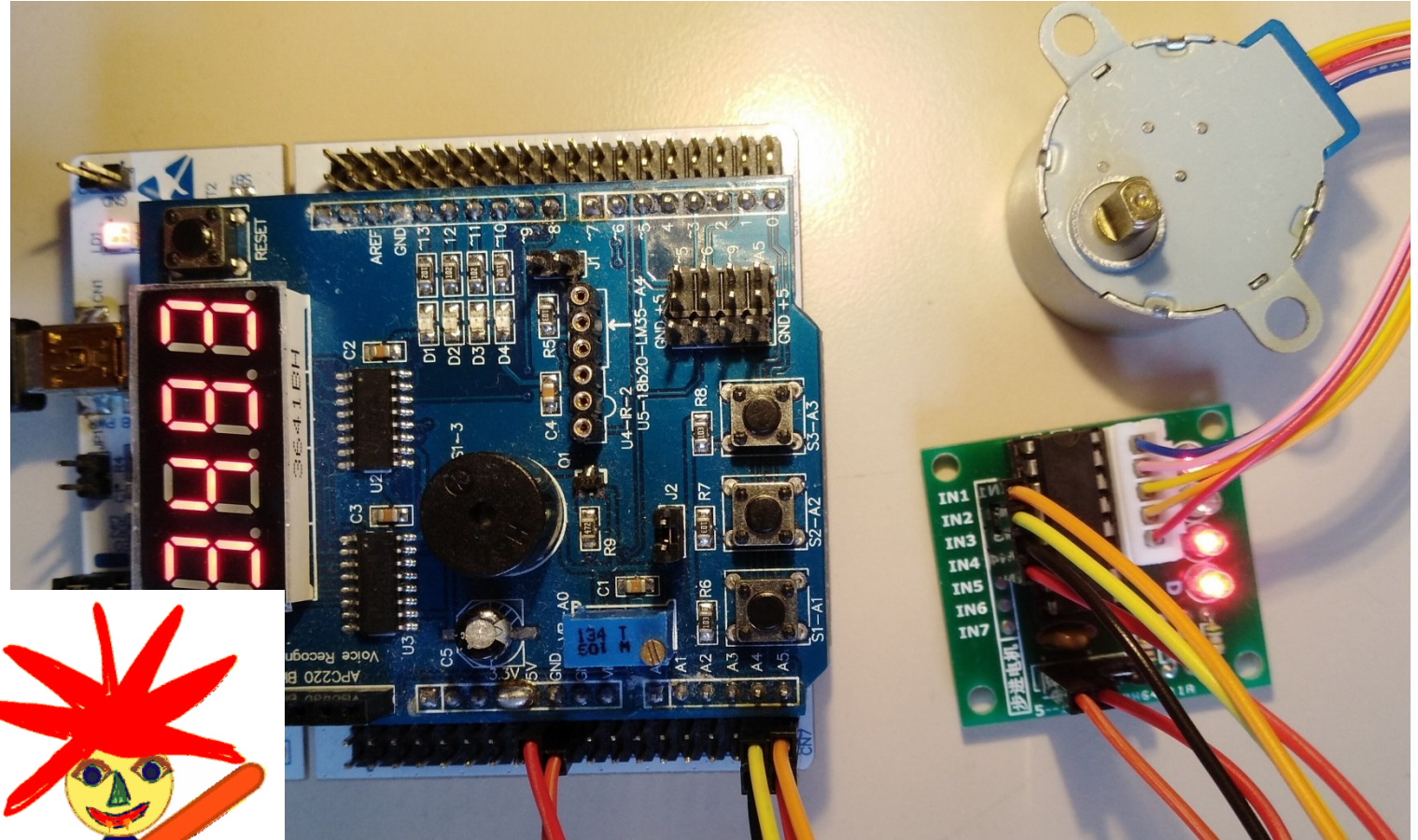
Multifunctionshield + Schrittmotorshield



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Projekthighlights:

- Schrittmotor an PC3..PC0
- Multifunctionshield
 - Analoge Geschwindigkeitsvorgabe mit Poti
 - 4-stellige Geschwindigkeitsanzeige auf Siebensegmentanzeige
 - Interruptgesteuerte Richtungsumkehr mit Taste A1

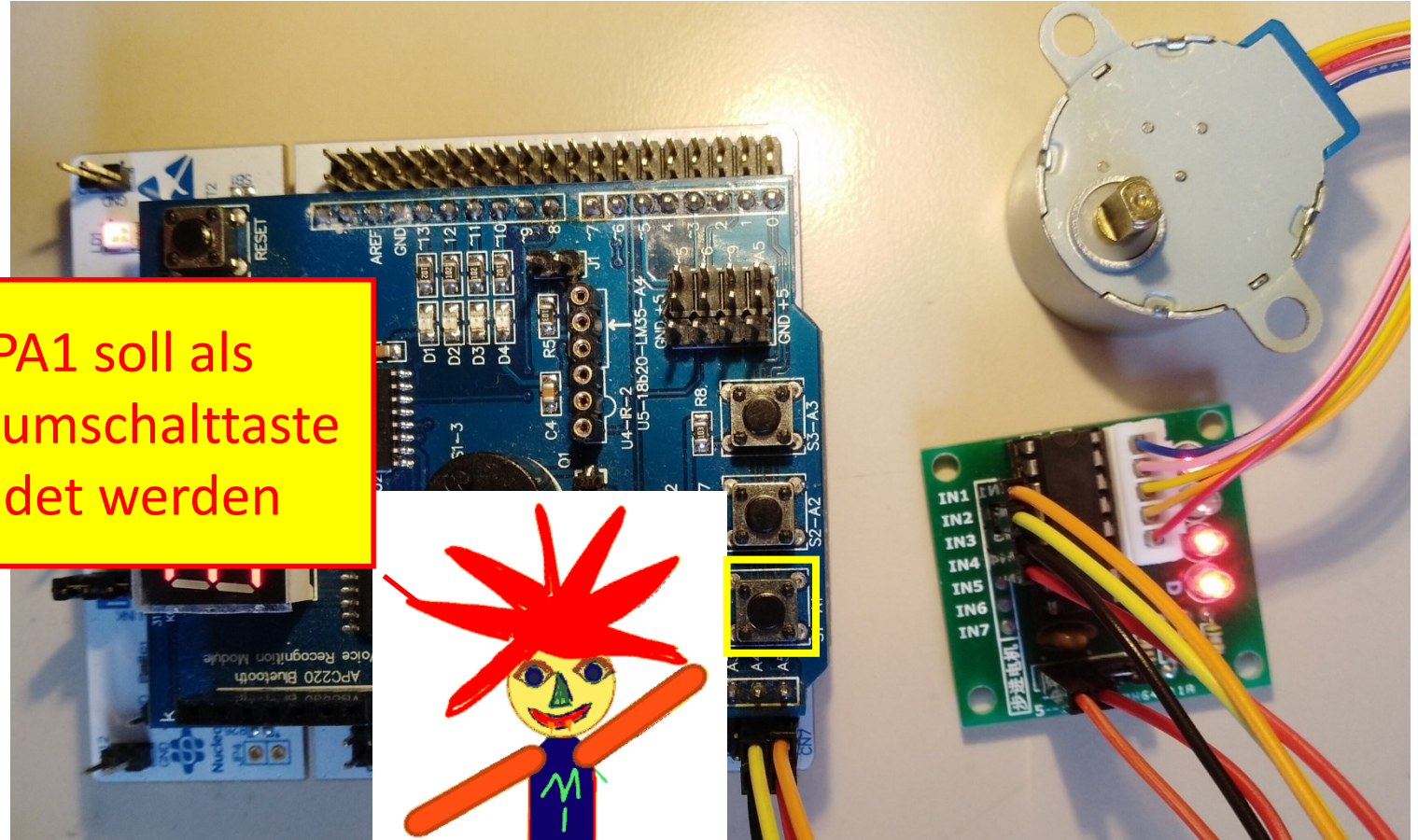


Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Projekthighlights:

- Schrittmotor an PC3..PC0
- Multifunctionshield
 - Analoge Geschwindigkeitsvorgabe mit Poti
 - 4-stellige Geschwindigkeitsanzeige auf Siebensegmentanzeige
- Interruptgesteuerte Richtungsumkehr mit Taste A1

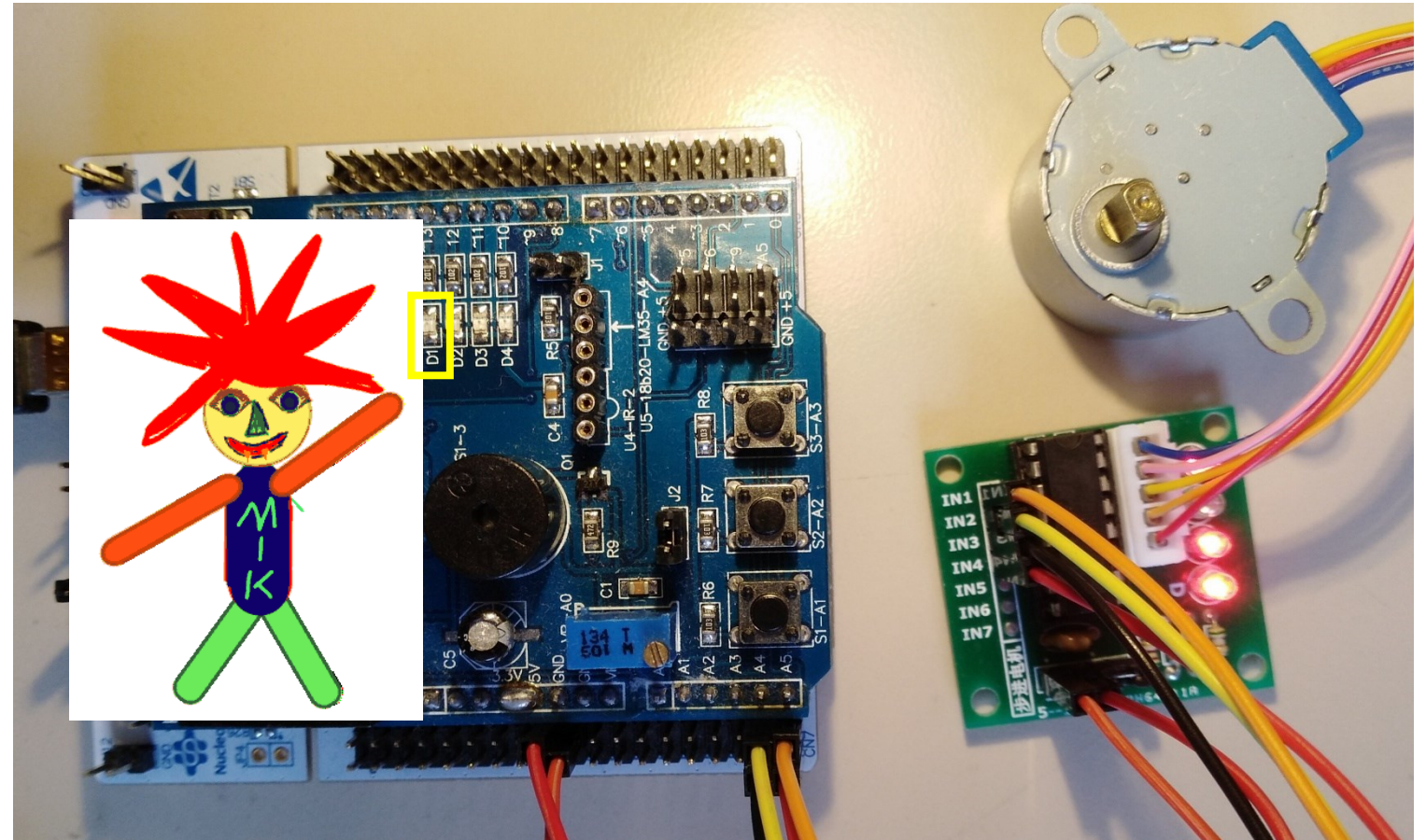
Taste PA1 soll als
Richtungsumschalttaste
verwendet werden



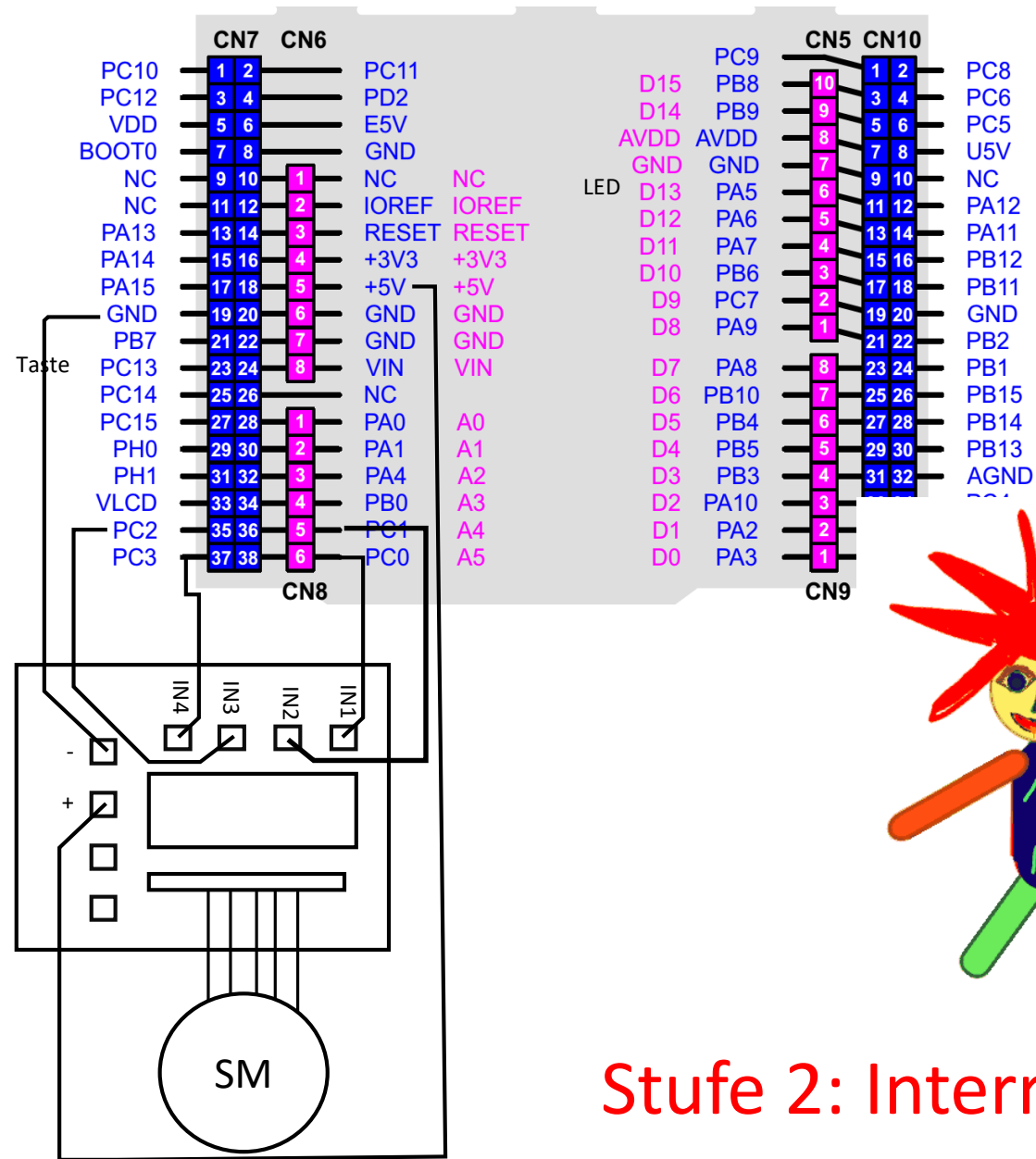
Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Projekthighlights:

- Schrittmotor an PC3..PC0
- Multifunctionshield
 - Die Drehrichtung soll mit LED D1 angezeigt werden.
- 4 anzeige auf Siebensegment-anzeige
- Interruptgesteuerte Richtungsumkehr mit Taste A1



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe



Anschlussbelegung und Konfiguration:

Nucleo	Schrittmotor
CN7 Pin 38 PC0	IN1 GPIO-Output
CN7 Pin 36 PC1	IN2 GPIO-Output
CN7 Pin 35 PC2	IN3 GPIO-Output
CN7 Pin 33 PC3	IN4 GPIO-Output
CN7 Pin 18 +5V	+
CN7 Pin 19 GND	-
Nucleo	Multifunctionshield
CN8 Pin 2 PA1	Taste PA1
CN5 Pin 6 PA5	LED D1



Anschluss des Schrittmotors

Stufe 2: Interrupteingang nutzen

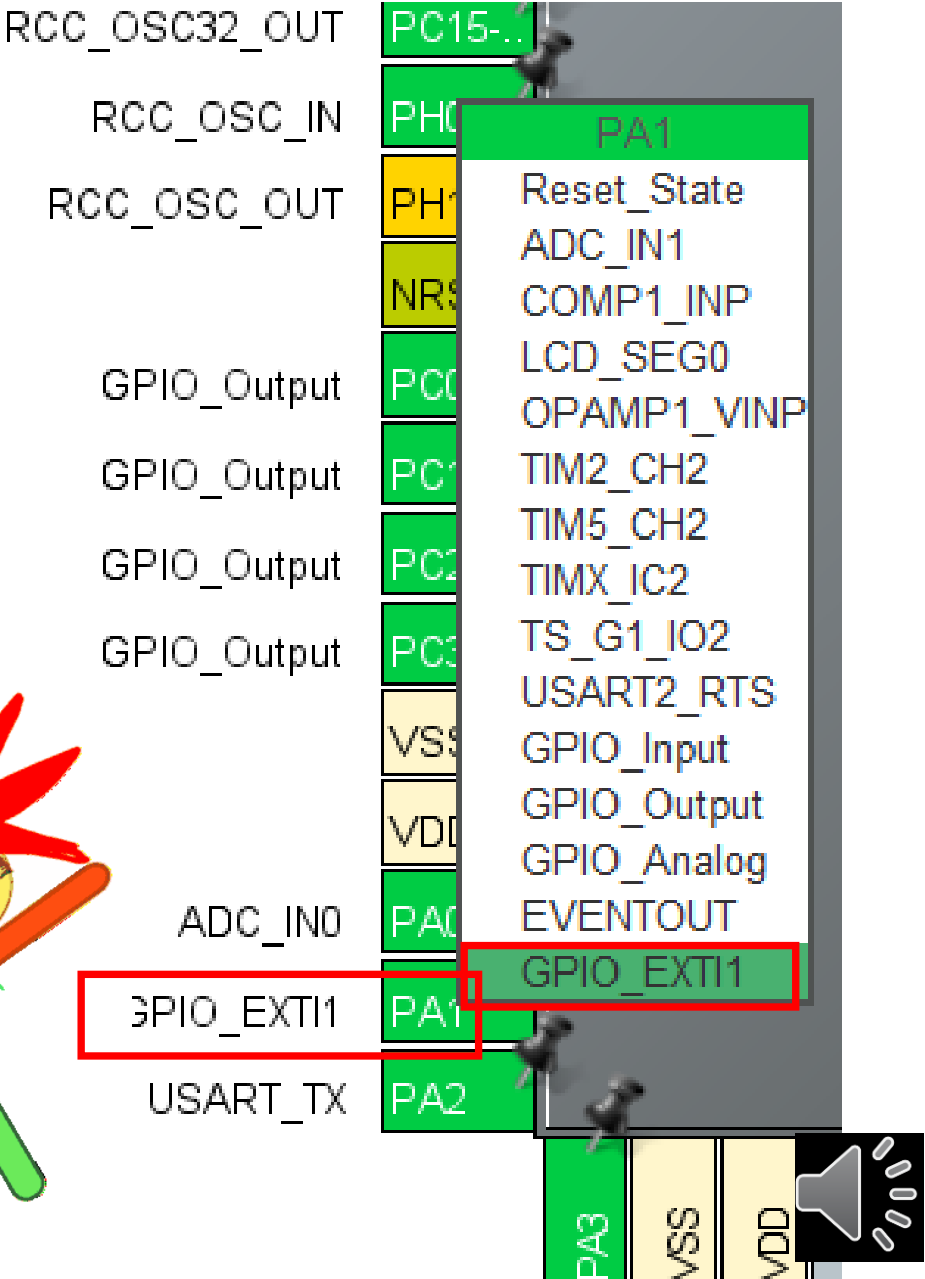


Projekt Schrittmotor mit Siebensegmentanzeige und analoger
Geschwindigkeitsvorgabe

Nucleo	Schrittmotor
CN7 Pin 38 PC0	IN1 GPIO-Output
CN7 Pin 37 PC3	IN4 GPIO-Output
CN7 Pin 36 PC1	IN3 GPIO-Output
CN7 Pin 35 PC2	IN2 GPIO-Output
CN7 Pin 18 +5V	+
CN7 Pin 19 GND	-
Nucleo	Multifunctionshield
CN8 Pin 2 PA1	Taste PA1

PA1 als GPIO_EXTI1 konfigurieren

Stufe 2: Interrupteingang nutzen



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Q

A->Z

Categories

System ...

DMA

GPIO

IWDG

NVIC

RCC

SYS

TS

WWDG

Analog

Multimedia

Comoutinga

GPIO Mode and Configuration

Configuration

Group By Peripherals

RCC

SYS

USART

NVIC

GPIO

Single Mapped Signals

ADC

Search Signals


Search (Ctrl+F)


☐ Show only Modified Pins

Pin N...	Signal on...	GPIOOut...	GPIO mode	GPIO Pul...	Maximu...	User Label	Modified
PA1	n/a	n/a	External I...	No pull-u...	n/a		<input type="checkbox"/>
PA5	n/a	Low	Output P...	No pull-u...	Very Low	LD2 [Gre...	<input checked="" type="checkbox"/>
PC0	n/a	Low	Output P...	No pull-u...	Very Low		<input type="checkbox"/>
PC1	n/a	Low	Output P...	No pull-u...	Very Low		<input type="checkbox"/>
F		w	Output P...	No pull-u...	Very Low		<input type="checkbox"/>
F		w	Output P...	No pull-u...	Very Low		<input type="checkbox"/>

External Interrupt Mode with Rising edge trigger detection

No pull-up and no pull-down





PA1 als GPIO_EXTI1 für steigende Flanke konfigurieren

Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Pinout & Configuration | Clock Configuration | Project Manager

Software Packs | Pinout

GPIO Mode and Configuration

Configuration

Group By Peripherals

✓ RCC	✓ SYS	✓ USART	✓ NVIC
✓ GPIO	✓ Single Mapped Signals		✓ ADC

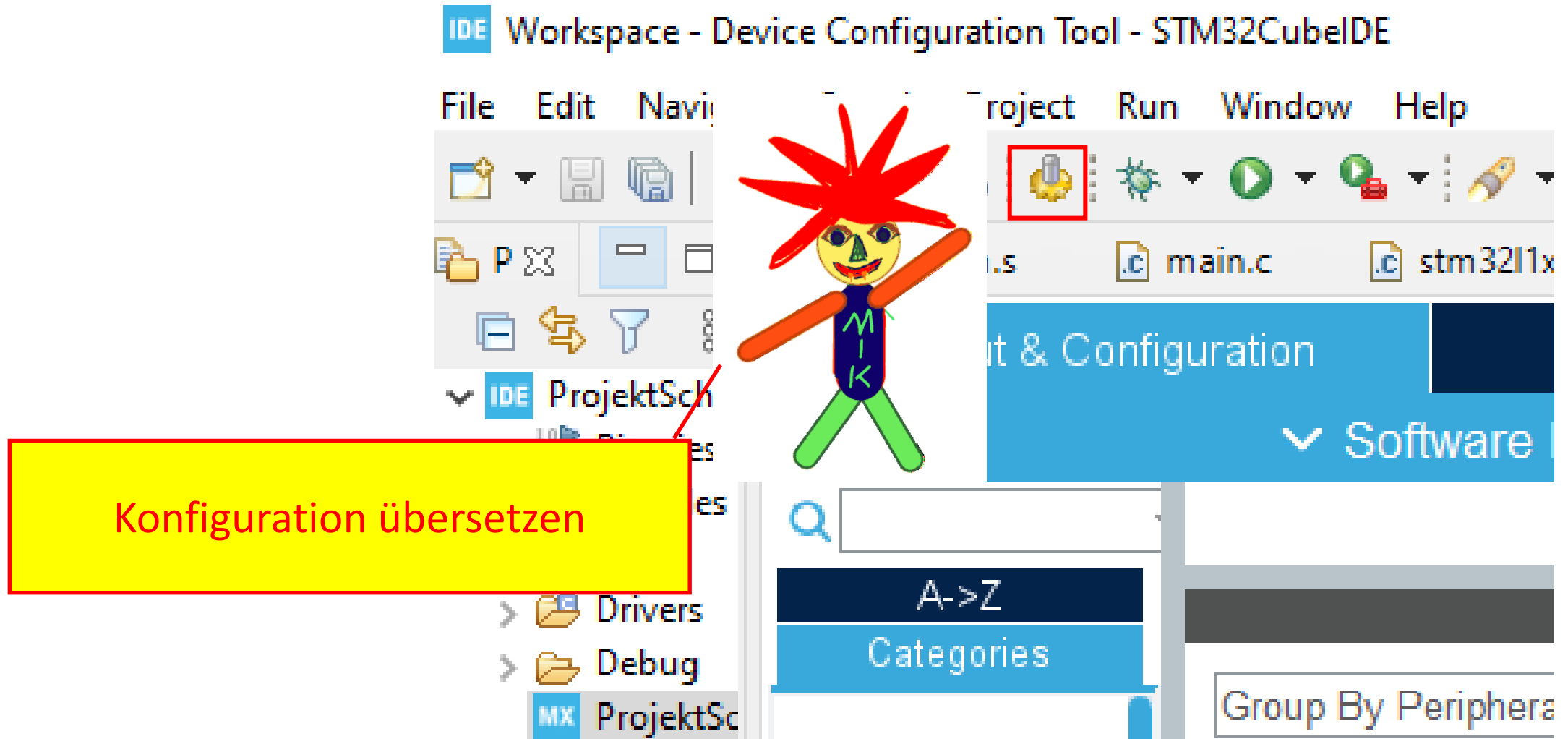
NVIC Interrupt Table	Enabled	Preemption Priority	Sub Priority
EXTI line1 interrupt	<input checked="" type="checkbox"/>	0	0
EXTI line[15:10] interrupts	<input type="checkbox"/>	0	0



Den externen Tasteninterrupt im Nested Vector Interrupt Controller (NVIC) freigeben



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Programm



b schleife


```
global HAL_GPIO_EXTI_Callback  
HAL_GPIO_EXTI_Callback:  
    ldr     r6, [R6, ODR]  
    eor     r6, r6, #Bit5  
    strb    r6, [R6, ODR]  
    bx      lr  
86  
87  
88  
89  
90  
91 .end
```

Zwischen b schleife
und .end



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Programm



```
80  
81 b schleife  
  
    .global HAL_GPIO_EXTI_Callback  
      
    ldr    R0,[R6,ODR]  
    eor    R0,Bit5  
    strb   R0,[R6,ODR]  
    bx     lr  
  
89  
90  
91 .end  
92
```

Schreiben wir ein
Unterprogramm mit dem
Namen
HAL_GPIO_EXTI_Callback



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Programm

```
80  
81      b      schleife  
82  
83      .global HAL_GPIO_EXTI_Callback  
84 HAL_GPIO_EXTI_Callback:  
85      ldr     R0,[R6,ODR]  
      eor     R0,Bit5  
      strb    R0,[R6,ODR]  
        
      :end
```



Das Unterprogramm
endet mit
bx lr (=Return)



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

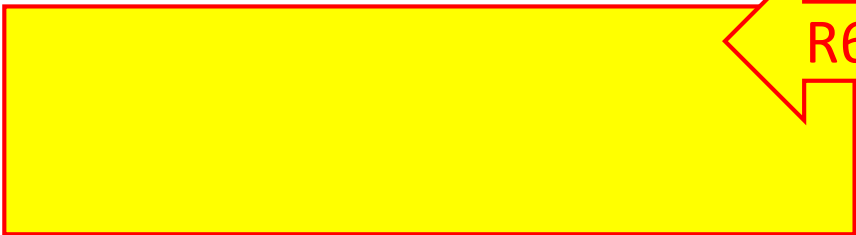
Programm

80
81
82

b schleife



obal HAL_GPIO_EXTI_Callback
_GPIO_EXTI_Callback:



← R6 speichert GPIOA

bx lr

83
90
91 .end
92

Das Unterprogramm
schaltet Bit5 von GPIOA
um



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Programm

80



b

schleife

HAL_GPIO_EXTI_Callback:

ldr R0,[R6,ODR]

eor R0,Bit5

strb R0,[R6,ODR]

bx lr

88

89

90

91 .end

92

Wir machen das Unterprogramm global im Projekt bekannt damit der Compiler das Unterprogramm als Interruptprogramm kennt



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Programm

Jetzt erst einmal
testen

Clean Project
+
Debug As




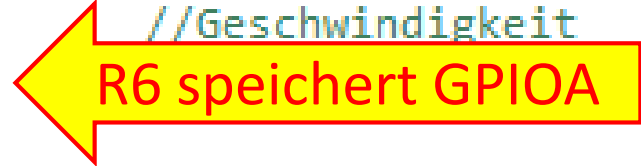


Bei Tastendruck
muss die LED D1 ein-
und ausschalten
(Achtung, prellt)



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Programm

Änderungen im
Hauptprogramm

```
11
12 main:
13     ldr     R0,=hadc
14     bl      HAL_ADC_Start
15     ldr     R4,=GPIOC
16     mov     R5,#10           //Geschwindigkeit
17       R6 speichert GPIOA
18
19 schleife:
20     ldr     R0,=hadc
21     bl      HAL_ADC_GetValue
22     strh    R0,[R8,ARR]
23     mov     R5,R0
24
25       R6 speichert GPIOA
26     mov     R0,#0b0011      //Schritt 0
27     strb    R0,[R4,ODR]      //ausgeben
28     lsr     R0,R5,8          //Wartezeit=Geschwindigkeit
29     bl      HAL_Delay        //warten
```



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Programm

Falls die LED aus ist:
Sprung zur
Drehrichtung links



```
11
12 main:
13     ldr     R0,=hadc
14     bl     HAL_ADC_Start
15     ldr     R4,=GPIOC
16     mov     R5,#10
17
18
19 schleife:
20     ldr     R0,=hadc
21     bl     HAL_ADC_GetValue
22     strh    R0,[R8,ARR]
23     mov     R5,R0
24
25     mov     R0,#0b0011
26     strb    R0,[R4,ODR]
27     lsr     R0,R5,8
28     bl     HAL_Delay
29
30
31
32
33
34
```

//Geschwindigkeit

R6 speichert GPIOA

R6 speichert GPIOA

//Schritt 0
//ausgeben
//Wartezeit=Geschwindigkeit
//warten



Projekt Schrittmotor mit Siebensegmentanzeige und analoger Geschwindigkeitsvorgabe

Programm

Der Code für
linksum
unterscheidet sich
nur in der
Schrittfolge



rechtsrum

linksum

