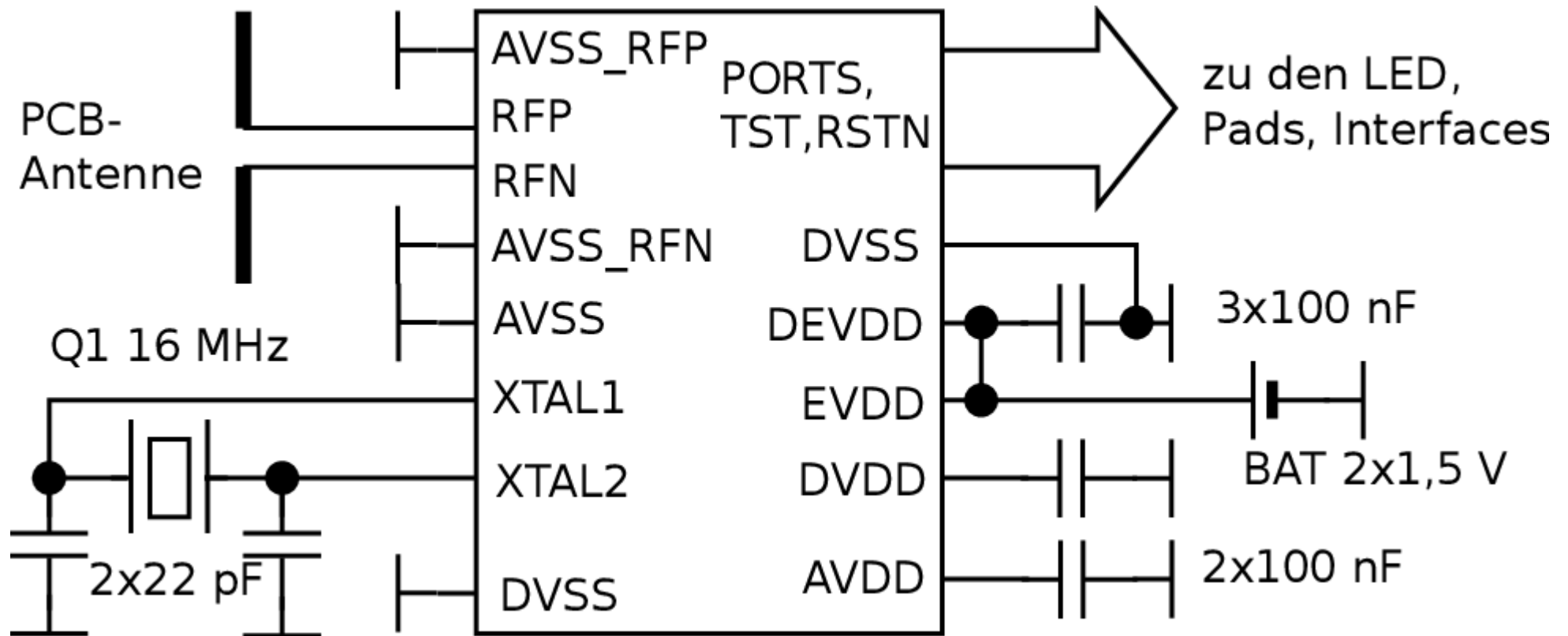
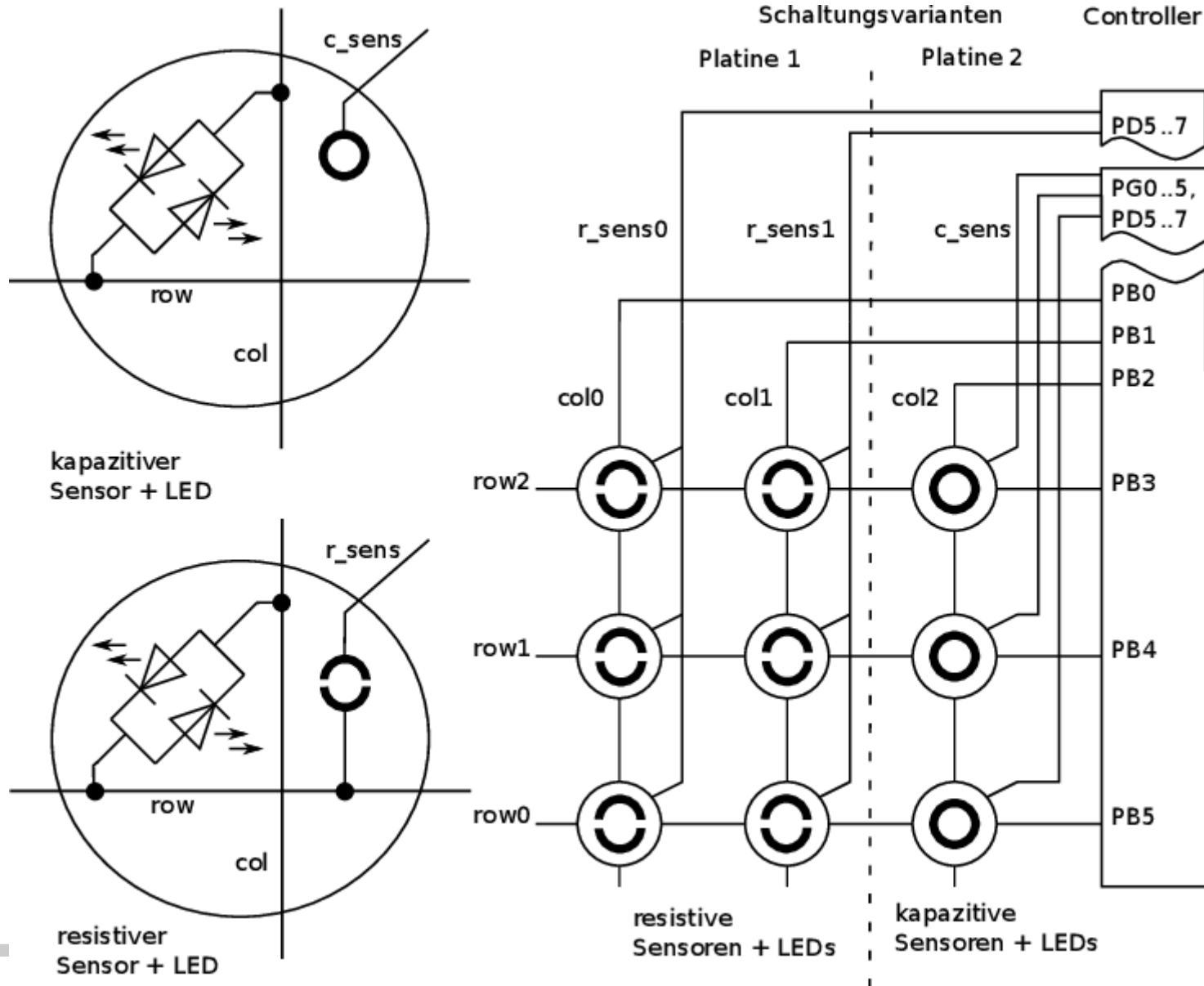


AVR Programmierung

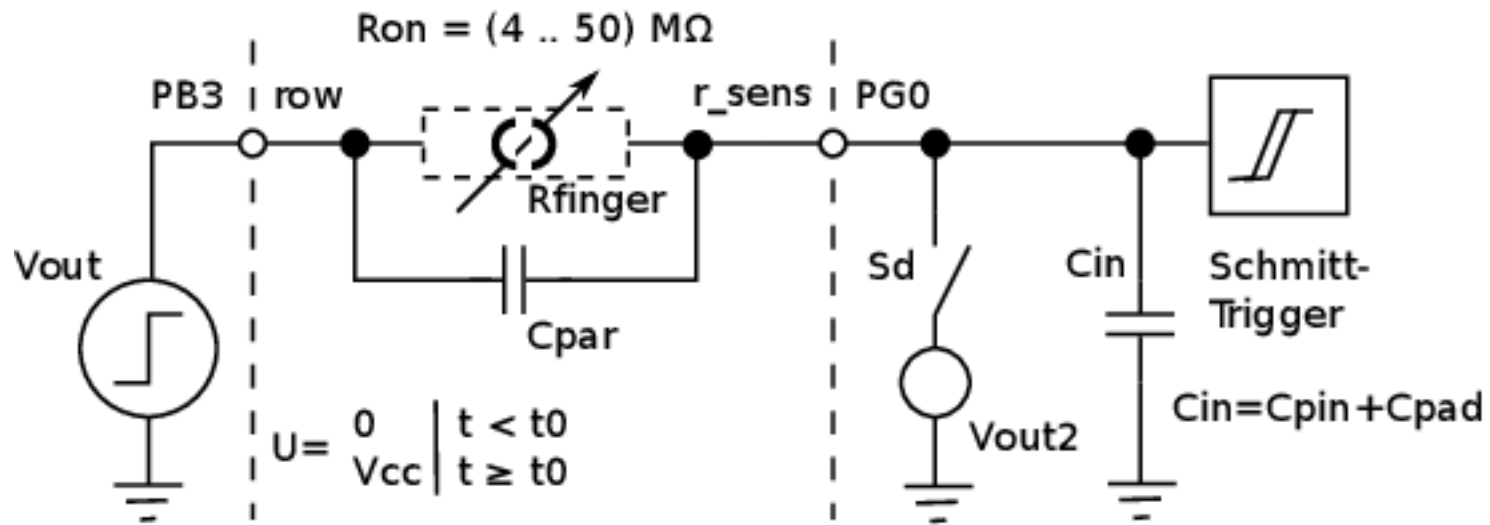
Die Schaltung I



Die Schaltung II

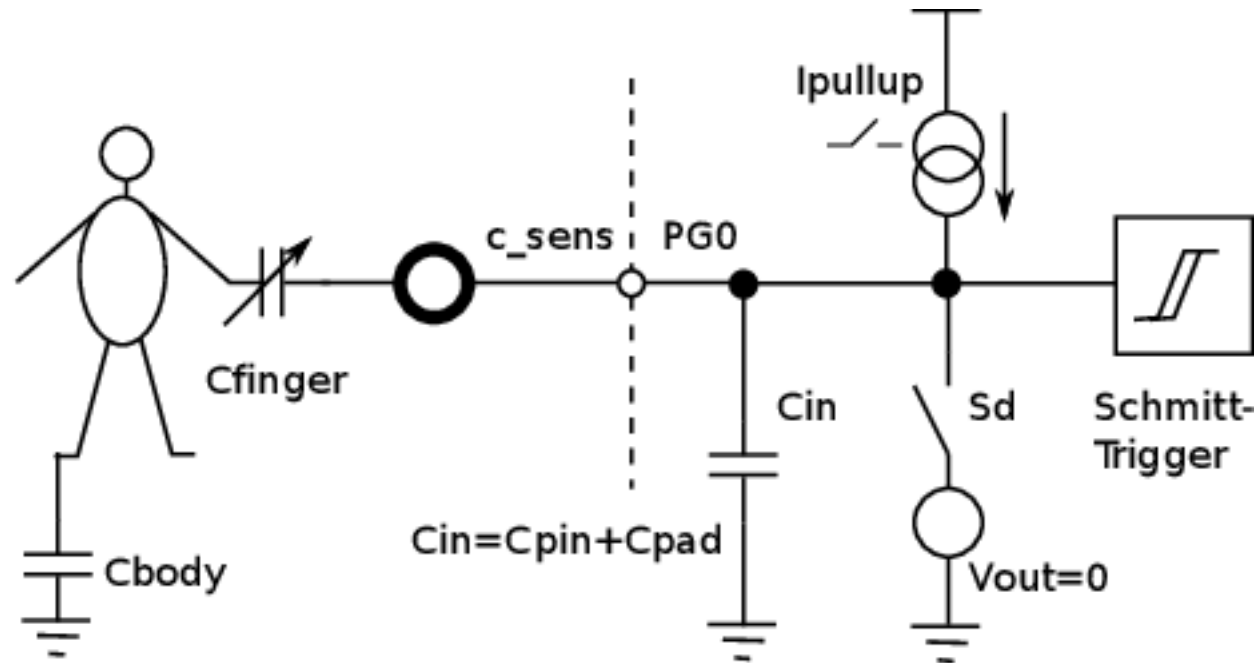


Resistive Sensor



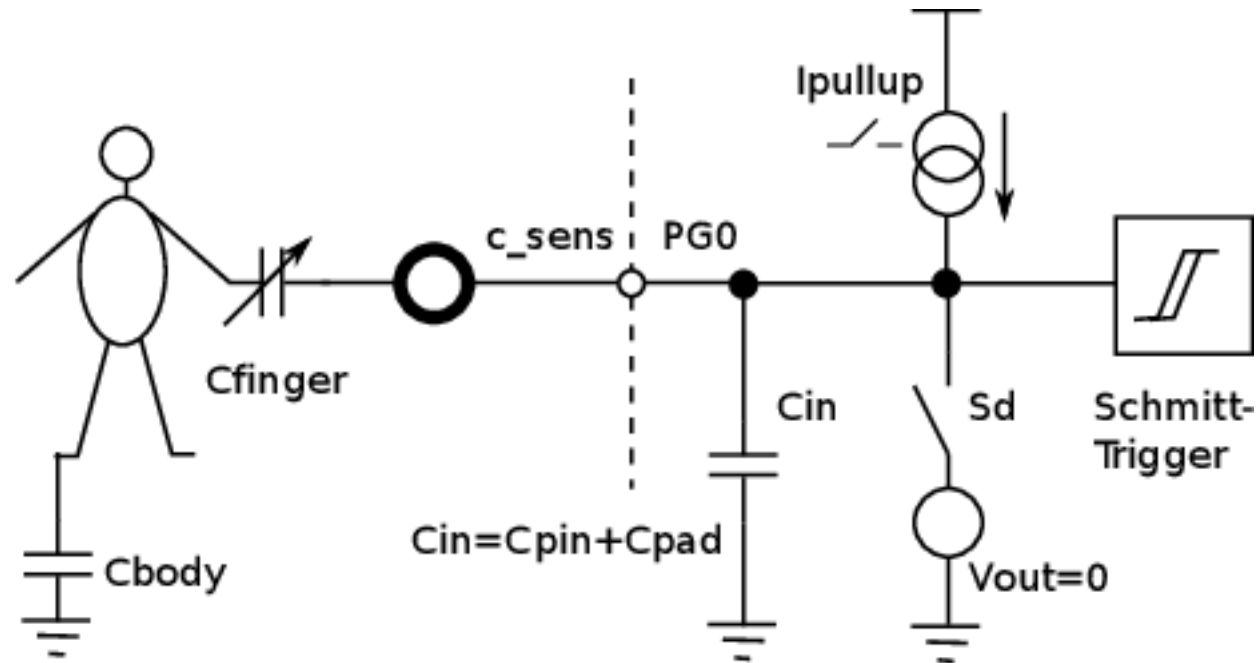
Schema resistive Sensor

Kapazitiver Sensor



Schema kapazitiver Sensor

Kapazitiver Sensor



Schema kapazitiver Sensor

- Oft Ansteuerung / Abfrage von einzelnen Bits nötig (Ein-/Ausgabeports)
 - In C: Bitmanipulations-Operatoren:
 - >> Rechts Schieben
 - << Links Schieben
 - & binäre UND-Verknüpfung
 - | binäre ODER-Verknüpfung
-

- $(1 \ll 3)$
- $00000001 \rightarrow 00001000 : 8$
- Gemeint: "Gesetztes Bit 0 um 3 Stellen nach links verschoben", also "Bit 3 ist gesetzt"
- $(1 \ll 3)$ bedeutet also "Gesetztes Bit 3, alle anderen 0"
- Wird zur Identifizierung einzelner Bitpositionen in einer Speicherstelle (Register) verwendet

- 2 | 4 (ODER Verknüpfung)
- 00000010 | 00000100 \rightarrow 00000110 : 6
- "Die Bits der Werte (Binärdarstellung) werden durch ODER verknüpft"

- 6 & 2 (UND Verknüpfung)
- 00000110 | 00000010 \rightarrow 00000010 : 2
- "Die Bits der Werte (Binärdarstellung) werden durch UND verknüpft"

- $(1 \ll 3) \parallel (1 \ll 5) \rightarrow 00101000 : 40$
- $_BV(3) \rightarrow 00001000$ (Bit 3)
- AVR Libc:

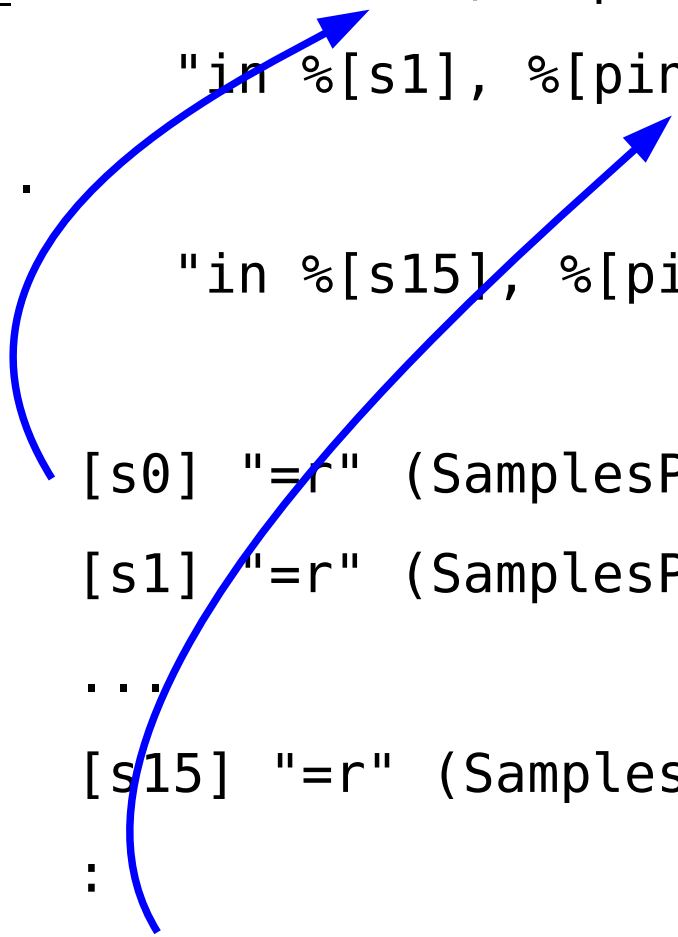
```
#define  _BV(bit)      (1 << (bit))
```
- leds.c:

```
#define COL_LEDS  (_BV(PB0) | _BV(PB1) |  
                  _BV(PB2))
```

Col. LED Bitpositionen:

Port B, Bits 0,1,2 $\rightarrow 00000111$ (PORT B)

```
/* File: captouch.c */  
__asm ("in %[s0], %[pind]" "\n\t"  
      "in %[s1], %[pind]" "\n\t"  
      ...  
      "in %[s15], %[pind]"  
      :  
      [s0] "=r" (SamplesPortD[0]),  
      [s1] "=r" (SamplesPortD[1]),  
      ...  
      [s15] "=r" (SamplesPortD[15])  
      :  
      [pind] "I" (_SFR_IO_ADDR(PIND)));
```



- Erzeugter Assembler-Code:

```
; 41 "captouch.c" 1
in r14, 9
in r15, 9
in r25, 9
in r16, 9
in r17, 9
in r27, 9
in r26, 9
in r31, 9
in r30, 9
in r23, 9
in r22, 9
in r21, 9
in r20, 9
in r19, 9
in r18, 9
in r24, 9

/* #NOAPP */
sts SamplesPortD, r14
sts SamplesPortD+1, r15
sts SamplesPortD+2, r25
sts SamplesPortD+3, r16
sts SamplesPortD+4, r17
sts SamplesPortD+5, r27
sts SamplesPortD+6, r26
sts SamplesPortD+7, r31
sts SamplesPortD+8, r30
sts SamplesPortD+9, r23
sts SamplesPortD+10, r22
sts SamplesPortD+11, r21
sts SamplesPortD+12, r20
sts SamplesPortD+13, r19
sts SamplesPortD+14, r18
sts SamplesPortD+15, r24
```

- `visudo -f /etc/sudoers`

```
mylogin ALL=(root) NOPASSWD: /usr/bin/avrdude
```

```
mylogin ALL=(root) NOPASSWD: /usr/bin/avarice
```

- Bei der Arbeit:

```
source  setup-sudotools.sh
```

```
mkdir work
cd work
unzip tic_tac_toe_reloaded-1.0.zip
unzip uracoli-src-0.2.0.zip
ln -s uracoli-src-0.2.0 uracoli
make -C uracoli/src xxo
cd 07_Spiel
make -f spiel.mk BOARD=cap flash
make -f spiel.mk BOARD=res flash
```
