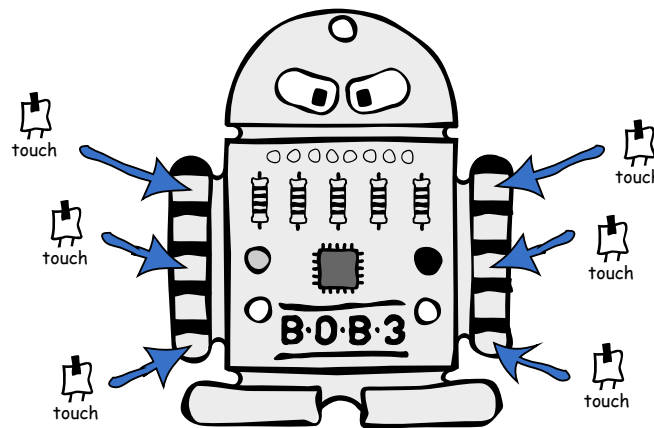


# Arbeitsblatt 10 - Touch-Sensoren

## BOB3's Arme sind Multifeld-Touch-Sensoren



Beide Arme vom BOB3 sind **Multifeld-Touch-Sensoren**. Die Arme „merken“, ob sie berührt werden oder nicht! Weil der Bob sogar bemerkt, **wo** du den jeweiligen Arm berührst, ob oben, mittig oder unten, sind es Multifeld-Touch-Sensoren. Bob hat damit insgesamt **sechs Tastsensoren**, die du ansteuern oder abfragen kannst!

Durch die Unterscheidbarkeit zwischen **oben**, **mittig** und **unten** kann man Programme schreiben, die die sechs Sensoren zur Ansteuerung verschiedener Aktivitäten verwenden, z.B. um Spiele zu programmieren oder bestimmte Verhaltensweisen des Roboters zu **starten**. Ein codiertes Programm könnte z.B. erst dann starten, wenn der Roboter mittels einer bestimmten Kombination der Armsensoren **entsperrt** wurde, oder man verwendet eine andere Kombination dazu, den Roboter in einen **gesperrten** Zustand zu versetzen!

## Zeit-Multiplex-Verfahren

Bob's Touch-Sensoren arbeiten im **Zeit-Multiplex-Verfahren**, dies ist eine Methode zur Signalübertragung, bei der mehrere Signale zusammengefasst werden. So kann man viele Signale an wenigen Mikrocontroller-Eingängen anschließen. Dabei teilen sich mehrere Signale einen Eingang: Der Controller von BOB3 hat **zwei Eingänge**, einen für Arm 1 und einen für Arm 2, die Arme können jedoch an insgesamt sechs verschiedenen Stellen angefasst werden und liefern also **sechs Signale!**

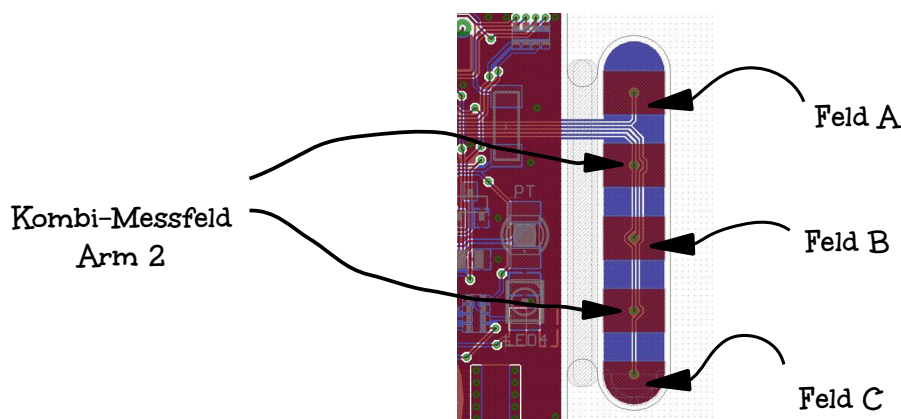


Der Trick ist das Zusammenspiel aus Aktivierungs- und Messfeldern. Von den drei Aktivierungsfeldern (A (oben), B (mittig) oder C (unten)) wird jeweils eins eingeschaltet und dann werden an beiden Armen die beiden Messfelder gemessen. Der Mikrocontroller sendet z.B. ein Signal an das

Aktivierungsfeld A und misst danach, ob er dieses Signal über das Kombi-Messfeld zurückempfängt. Falls in dem Moment Arm 1 oben berührt wird, hat der Controller einen Rückempfang und 'bemerkt' so, dass gerade das Messfeld und das Aktivierungsfeld A über einen Finger kurzgeschlossen werden. Nachdem alle drei Aktivierungsfelder aktiviert wurden haben wir sämtliche 6 Ergebnisse. Dabei interessiert uns nur die Ja/Nein Information, d.h. ob die Felder berührt werden oder nicht. Der Controller von BOB3 macht das ca. 200 mal pro Sekunde!!!

### Aufbau der Sensoren

Jeder Arm besteht jeweils aus fünf Feldern: 3 **Aktivierungsfelder** (A, B, C) und 2 **Messfelder**. Sobald du ein *Aktivierungsfeld* **gleichzeitig** mit einem *Messfeld* berührst, bekommt der Bob ein Signal, ob Feld A, Feld B oder Feld C berührt wurde.



Die beiden Rechtecke des Kombi-Messfelds sind elektrisch miteinander verbunden!

### Software-Bibliothek

Zur Ansteuerung der Armsensoren stehen in der Software-Bibliothek des BOB3 fertig implementierte Methoden zur Verfügung. Die Methode `bob3.getArm(id)` liefert den **aktuellen Wert** des jeweiligen Sensors:

<b>Rückgabewert</b>	0	1	2	3
<b>Bedeutung</b>	Keine Berührung	Berührung oben	Berührung mittig	Berührung unten

<b>id</b>	1	2
<b>Sensor</b>	Arm 1	Arm 2

Mit der Methode `bob3.enableArms(ARMS_DETECTOR)` können die Sensoren z.B. für den **Friend-Detection-Mode** aktiviert bzw. deaktiviert werden. Dies ist ein anderes Verfahren, bei dem die Leitfähigkeit der Finger gemessen wird: Alle Aktivierungsfelder werden hierbei gleichzeitig aktiviert und an den beiden Messfeldern wird der Stromfluss gemessen.

**Aufgabe 1:** Beschreibe, wie das Multiplexverfahren funktioniert.

---



---



---



---



---

**Aufgabe 2:** Beschreibe, aus welchen Teilen die Armsensoren von BOB3 bestehen und wie sie funktionieren.

---



---



---



---



---

**Aufgabe 3:** Welche Programm-Codes sind korrekt und würden compilieren? Kreuze die richtigen Antworten an, es sind mehrere Antworten möglich:

- `int sensorWert = bob3.getArm();`
- `int sensorWert == bob3.getArm(2);`
- `int sensorWert = bob3.getArm(1,2);`
- `int sensorWert = bob3.getArm(2,1);`
- `int wert = bob3.getArm(2,1);`
- `int WERT = bob3.getArm(2);`
- `int wert = getArm(1);`
- `if (2 == bob3.getArm(1))`
- `if (bob3.getArm(2) == 1)`
- `int sensorWert = bob3.getArm(1);`
- `int sensorWert = bob3.getArm(2);`

**Aufgabe 4:** Betrachte das folgende Programm und beschreibe genau, was bei welcher Aktion am Bob passiert!

```

1 #include <BOB3.h>
2
3 void loop() {
4     int wert1 = bob3.getArm(1);
5     int wert2 = bob3.getArm(2);
6
7     if (wert1 == 1) {
8         bob3.setEyes(WHITE, WHITE);
9         delay(200);
10        bob3.setEyes(OFF, OFF);
11        delay(200);
12    }
13
14    if (wert1 == 3) {
15        bob3.setWhiteLeds(ON, ON);
16        delay(200);
17        bob3.setWhiteLeds(OFF, OFF);
18        delay(200);
19    }
20
21    if ((wert1 == 2) && (wert2 == 2)) {
22        bob3.setEyes(ORANGE, WHITE);
23        bob3.setWhiteLeds(ON, OFF);
24        delay(200);
25        bob3.setEyes(WHITE, ORANGE);
26        bob3.setWhiteLeds(OFF, ON);
27        delay(200);
28    }
29
30 }
31

```

---

---

---

---

---

---

---

---

---

---

**Aufgabe 5:** Nenne alle möglichen Rückgabewerte der Methode `bob3.getArm()`.

---

---