

## ZUSATZMODELL

# Funktionsmodell

## Robotik, Sensorik, Aktorik und KI verstehen

### LEITFRAGEN:

- Wie konfiguriere ich einen Controller so, dass alle Sensoren und Aktoren korrekt erkannt und genutzt werden können? (*Kritisches Denken*)
- Wie werden digitale und analoge Signale verarbeitet und wie kann ein Programm darauf reagieren? (*Kreativität & Kritisches Denken*)
- Wie lassen sich mechanische Bauteile wie Motoren präzise steuern und mit Sensoren verknüpfen? (*Kritisches Denken*)
- Wie können Ereignisse (Events) und Sensordaten genutzt werden, um selbstständig reagierende Systeme zu entwickeln? (*Kreativität*)
- Wie lernt ein neuronales Netz anhand von Trainingsdaten, selbstständig logische Entscheidungen zu treffen? (*Kritisches Denken / Kommunikation*)

### DIE UNTERRICHTSIDEES AUF EINEN BLICK

**Klassenstufe:** 7–12

**Zeitaufwand:** 6–8 Doppelstunden (modular einsetzbar)

**Schwierigkeitsgrad:**

- Modell: Einfach
- Programmierung: Einfach

**Modellart:** Funktionsmodell/Testboard mit Tastern, LEDs, Motor, digitalen und analogen Sensoren, Kamera, neuronalen Netzen

## **MODELLBESCHREIBUNG / AUFGABE**

Das Funktionsmodell ist ein modulares Lern- und Trainingssystem, mit dem Schülerinnen und Schüler die grundlegende Arbeitsweise eines Controllers sowie die Interaktion von Sensoren, Aktoren und KI-Modellen erlernen. Jede Einheit baut auf der vorherigen auf, kann jedoch auch einzeln eingesetzt werden.

Das Modell umfasst folgende Lerneinheiten:

1. Controller-Konfiguration und Grundlagen
2. Programmierung von Tastern und LEDs
3. Programmierung von Motoren (mit Encoder)
4. Programmierung digitaler Sensoren (Fototransistor & Spursensor)
5. Programmierung analoger Sensoren (Temperatur & Ultraschall)
6. Einsatz der Kamera und Eventsteuerung
7. Training eines neuronalen Netzes (Logikgatter)

Der didaktische Mehrwert des Modells liegt im unmittelbaren Arbeiten mit realen Signalen, Messwerten und Reaktionen. Dadurch entsteht ein authentischer Praxisbezug, der kritisches Denken, Problemlösekompetenz und technisches Verständnis stärkt.

Das Modell eignet sich für vielfältige Unterrichtskontexte – von Informatik und Technik über Robotik-AGs bis hin zu Projekttagen oder vertiefenden KI-Modulen. Es verbindet einfache Zugänglichkeit mit inhaltlicher Tiefe und bietet Lehrkräften ein universell einsetzbares Werkzeug zur Förderung moderner digitaler Kompetenzen.

## **Einheit 1 – Controller-Konfiguration & Grundlagen**

### **Leitfragen:**

- Wie werden Sensoren und Aktoren am TXT 4.0 zugeordnet?
- Wie verbindet man die STEM Suite mit dem Controller?
- Wie prüft man die Funktion von Tastern und LEDs?
- Wie schreibt man ein erstes einfaches Programm?

### **Unterrichtsidee:**

**Ziel:** Grundlagen des Arbeitens mit dem Controller: Konfiguration, Verbindung, Funktionsprüfung und erstes Programm.

**Dauer:** 1 Doppelstunde.

### **Beschreibung:**

- Eingänge für Mini-Taster konfigurieren.
- LEDs zu Ausgängen zuordnen.
- Verbindung per WLAN + API-Key.
- Testoberfläche nutzen (0/1-Werte prüfen, LEDs schalten)
- Erstes Programm: LED ein-/ausschalten, blinken lassen, Tastersteuerung

### **Methodik:**

- Schnelle Ergebnisse, einfache Fehlerdiagnose.
- Grundlage für Motoren, Sensoren, Kamera, KI.

## Einheit 2 – Programmierung von Tastern und LEDs

### Leitfragen:

- Wie funktionieren digitale Eingänge/Ausgänge?
- Wie reagiert ein Programm auf Tastereingaben?

### Unterrichtsidee:

**Ziel:** Grundlegende digitale Logik verstehen und sichtbar umsetzen.

**Dauer:** 1 Doppelstunde.

### Beschreibung:

- Mini-Taster den Eingängen zuordnen
- LEDs als Ausgänge konfigurieren
- Test: Taster → 0/1
- Programme entwickeln:
  - o LED ein- und ausschalten
  - o Umschaltfunktion („toggle“)
  - o Zwei LEDs abhängig voneinander steuern-

### Methodik:

- Sichtbare Rückmeldung unterstützt Verständnis
- Klarer Einstieg in grundlegende digitale Logik

## **Einheit 3 – Programmierung von Motoren (Encoder)**

### **Leitfragen:**

- Wie werden Motoren angesteuert?
- Wie funktioniert ein Encoder?

### **Unterrichtsidee:**

**Ziel:** Drehrichtung und Drehzahl kontrollieren sowie Encoderwerte auslesen.

**Dauer:** 1–2 Stunden.

### **Beschreibung:**

- Motoren an Ausgängen konfigurieren
- Testoberfläche nutzen (Drehrichtung Motor im Uhrzeigersinn/gegen den Uhrzeigersinn(cw/ ccw)
- Programme entwickeln:
  - o Zeitgesteuerte Bewegung
  - o Drehzahlregelung
  - o Encoder auslesen und verarbeiten

### **Methodik:**

- Hohe Praxisnähe zur Robotik
- Gute Grundlage für spätere Regelkreise oder Bewegungslogik

## **Einheit 4 – Programmierung digitaler Sensoren (Fototransistor & Spursensor)**

### **Leitfragen:**

- Wie arbeitet ein digitaler Sensor?
- Wie wird ein Spursensor für Linienerkennung genutzt?

### **Unterrichtsidee:**

**Ziel:** Digitale Sensordaten auslesen und für Reaktionen nutzen.

**Dauer:** 1–2 Stunden.

### **Beschreibung:**

- Fototransistor und Spursensor konfigurieren
- Test: Helligkeits- und Linienwechsel als 0/1-Signal
- Programme entwickeln:
  - LED bei Dunkelheit aktivieren
  - LED bei Linienerkennung einschalten

### **Methodik:**

- Sensordaten werden unmittelbar sichtbar
- Relevanz für mobile Robotik

## **Einheit 5 – Programmierung analoger Sensoren (Temperatur & Ultraschall)**

### **Leitfragen:**

- Was unterscheidet analoge von digitalen Sensoren?
- Wie liest man Temperatur- oder Abstandswerte aus?

### **Unterrichtsidee:**

**Ziel:** Analoge Werte interpretieren und in Programmlogik überführen.

**Dauer:** 1–2 Stunden.

### **Beschreibung:**

- NTC-Temperatursensor und Ultraschallsensor konfigurieren
- Werte in der Testoberfläche anzeigen lassen
- Programme entwickeln:
  - Temperaturgrenzen festlegen
  - Abstandswarnung über LED ausgeben

### **Methodik:**

- Förderung mathematischen Denkens (Grenzwerte, Vergleiche)
- Vorbereitung auf KI-basierte Abstandserkennung

## **Einheit 6 – Programmierung der Kamera & Events**

### **Leitfragen:**

- Wie arbeitet die Kamera des TXT?
- Was ist ein Event-System?

### **Unterrichtsidee:**

**Ziel:** Bewegungserkennung nutzen und ereignisbasierte Programme erstellen.

**Dauer:** 1–2 Stunden

### **Beschreibung:**

- Kamera aktivieren.
- Events definieren („Movement detected“)
- Programme entwickeln:
  - o LED bei Bewegung einschalten.
  - o Erweiterbar: Alarm, Logging, Aktionen auslösen

### **Methodik:**

- Einführung in Event-basierte Programmierung.
- Fundament für reaktive Systeme



## Einheit 7 – Training neuronale Netze (Logikgatter)

### Leitfragen:

- Wie funktioniert ein neuronales Netz?
- Wie trainiert man einfache logische Funktionen AND/OR/XOR?

### Unterrichtsidee:

**Ziel:** Struktur und Funktionsweise eines einfachen neuronalen Netzes verstehen.

**Dauer:** 1 Doppelstunde.

### Beschreibung:

- Neuronales Netz mit Eingängen und Ausgängen konfigurieren
- Trainingsdaten eingeben
- Netz trainieren und anschließend ausführen
- Verhalten analysieren und mit Logikgattern vergleichen

### Methodik:

- Niederschwelliger Einstieg in KI-Konzepte
- Leichter Transfer zu komplexeren Netzen möglich