

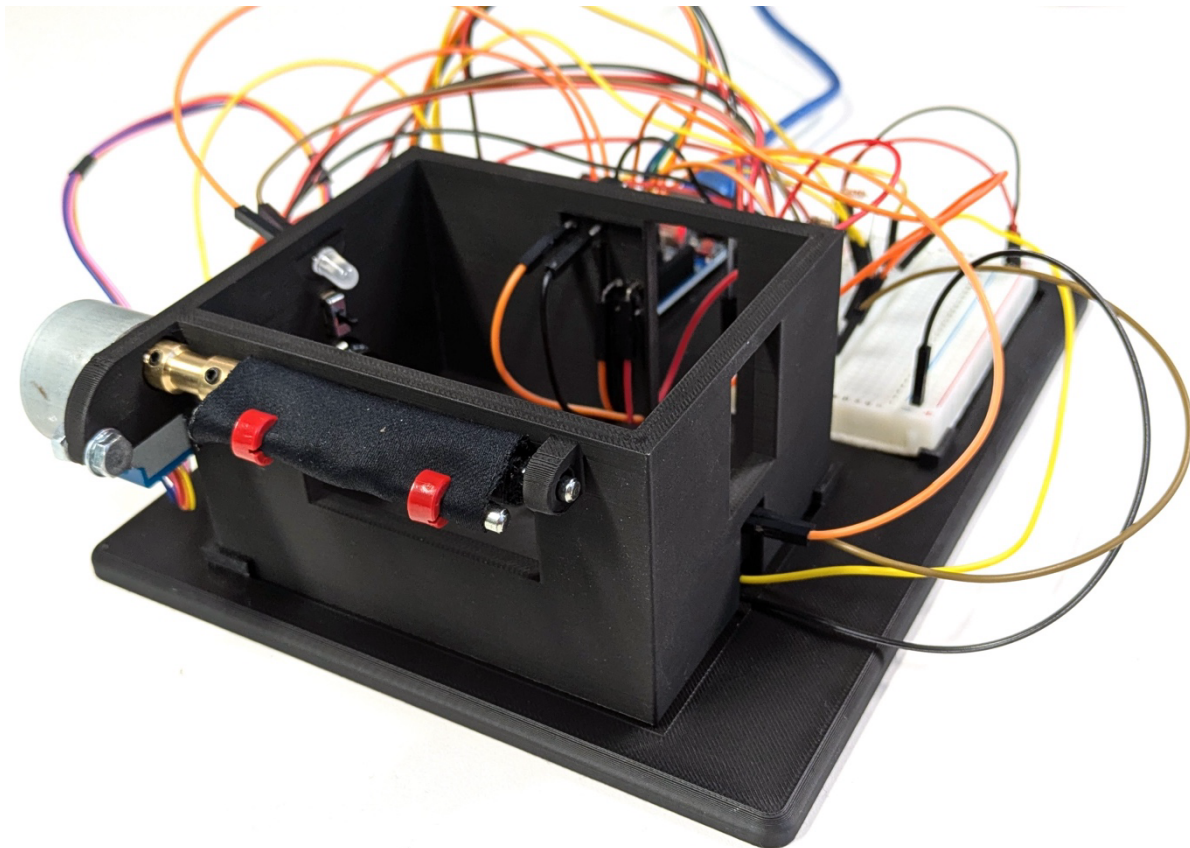
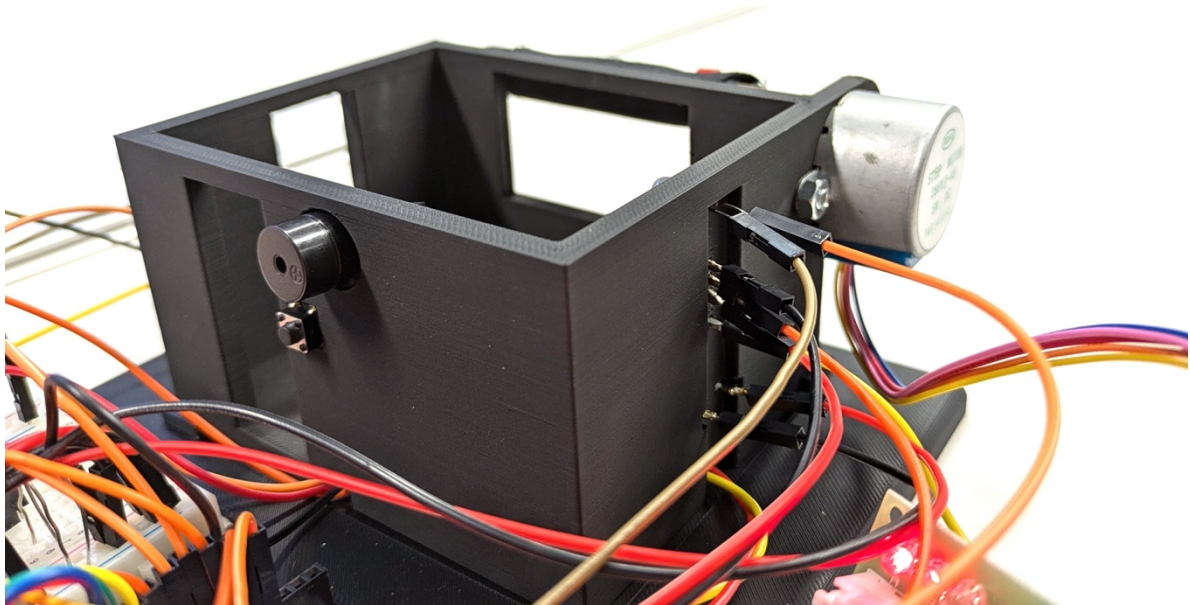
Lehren mit dem All

Smarthome im Weltall

Die Zukunft des Wohnens
zwischen den Planeten

Programmierung eines Smarthomes

Eigene Beispielbilder



Kurzinformationen, Überblick, Lernziele	Seite 4
Zusammenfassung der Aufgaben	Seite 5
Verfügbare Downloads	Seite 6
Material	Seite 6
Software / Programmieroberfläche	Seite 7
Ziele des Kurses, Aufbau und Inhalt	Seite 8

Lehren mit dem All –
www.esa.int/education

Das ESA Education Office freut sich über Feedback und Kommentare
teachers@esa.int

Eine Produktion von ESA Education
Copyright 2024 © European Space Agency

→ Smarthome im Weltall

Die Zukunft des Wohnens zwischen den Planeten

Programmierung eines Smarthomes

Kurzinformationen

Alter: 14-20 Jahre

Lehrplanbezug: Informatik, Technik

Schwierigkeitsgrad: mittel

benötigte Zeit: 240-320 Minuten

Örtlichkeit: in einem Gebäude

Kosten: ca. 50,-€/Arbeitsplatz

Voraussetzungen: Computer/Laptop, Arduino IDE, 3D-Druck Möglichkeit

Schlüsselbegriffe: Arduino, Smarthome, Sensor, Quelltext

Überblick

Dieser Kurs bietet Schülerinnen und Schülern (SuS) die Möglichkeit, das Thema Smarthome in einem innovativen Kontext zu erforschen. Im Fokus steht die Programmierung des Arduino UNO, eines vielseitigen Mikrocontrollers, mit dem grundlegende Automatisierungsfunktionen in einem Smarthome-Modellhaus umgesetzt werden können. Die Schülerinnen und Schüler lernen nicht nur die theoretischen Grundlagen moderner Smarthome-Technologien kennen, sondern entwickeln auch praktische Fähigkeiten im Umgang mit Sensoren, Aktoren und deren Programmierung. Dabei erfahren sie, wie automatisierte Systeme gestaltet und an individuelle Anforderungen angepasst werden können.

Lernziele

- Die SuS verbessern ihre analytischen Fähigkeiten, indem sie Schaltungen aufbauen und Probleme systematisch lösen.
- Die SuS lernen, wie die Programmiersprache C/C++ funktioniert, und setzen diese zur Steuerung des Mikrocontrollers Arduino UNO ein.
- Die SuS verstehen, wie Sensoren Daten erfassen und diese für Smarthome-Anwendungen genutzt werden können.
- Die SuS erkennen, wie Aktoren eingesetzt werden, um automatisierte Abläufe in Smarthome-Systemen zu realisieren.
- Die SuS verbessern ihre Teamfähigkeit durch gemeinsame Umsetzung der Aufgaben.

Zusammenfassung der Aufgaben

	Bezeichnung	Beschreibung	Ergebnis	Voraussetzungen	Zeit
1	Thematischer Einstieg Smarthome	Es soll ein kurzer Einstieg in die Thematik Smarthome und deren Einsatzbereiche erfolgen. Dies kann sowohl lehrkraftzentriert als auch im offenen Unterrichtsgespräch erarbeitet werden.	Die Lernenden erhalten ein grundlegendes Verständnis darüber, was ein Smarthome ausmacht und welche Anwendungen realisiert werden können.	Keine	10-15 Minuten
2	Einführung in die Hard- und Software	Die SuS erhalten eine Vorstellung des Arduino UNO als Mikrocontroller und seiner möglichen Einsatzbereiche. Zusätzlich wird die Programmieroberfläche der Arduino IDE präsentiert.	Die SuS kennen die grundlegenden Funktionen des Arduino und können die Arduino IDE bedienen und wissen, wie Programme erstellt und hochgeladen werden.	Keine	20 Minuten
3	Einführende Aufgaben	Die SuS erstellen selbstständig ihr erstes Programm, das die interne LED des Arduino steuert. Anschließend übertragen sie die Steuerung auf eine externe LED mit Schaltung auf einem Breadboard.	Die SuS verstehen, wie digitale Ausgänge gesteuert werden, und sind in der Lage, einfache Schaltungen aufzubauen.	Einführung in Arduino und IDE	30 Minuten
4	Smarthome-Programmierung	Mit Hilfe anleitender Aufgaben programmieren die SuS ein Smarthome-System in einem 3D-gedruckten Modellhaus. Dies umfasst die Steuerung von Licht, Sensoren, Aktoren und Automatisierungen wie Heizung oder Rollläden.	Die SuS entwickeln ein funktionales Smarthome-Programm, mit dem sie ein Modellhaus automatisieren können.	Erste Programmiererfahrungen	140-260 Minuten

Verfügbare Downloads

Lernmaterialien

	1_Smarthome_Handreichung_für_Lehrkraefte.pdf
	2_Smarthome_Lernmaterialien_mit_Loesungen.pptx
	3_Smarthome_Lernmaterialien_mit_Loesungen.pdf
	4_Smarthome_Lernmaterialien_SuS.pptx
	5_Smarthome_Lernmaterialien_SuS.pdf
	6_Smarthome_Stueckliste.pdf
	7_Smarthome_Druckdatei_Grundplatte.stl
	8_Smarthome_Druckdatei_Haus.stl
	9_Smarthome_Beispielprogramm.ino

Material

In diesem Kurs wird der Arduino UNO und die Programmieroberfläche Arduino IDE verwendet. Diese kann kostenlos unter <https://www.arduino.cc/en/software> heruntergeladen und installiert werden.

Für den Kurs werden zwei Bibliotheken verwendet:

- **Stepper.h:** Diese Bibliothek wird zur Steuerung des Schrittmotors benötigt und wird mit der Installation der Arduino IDE installiert.
- **NTC_Thermistor.h:** Diese Bibliothek dient zur Verwendung des Thermistors und muss installiert werden. Die Schülerinnen und Schüler finden in der Präsentation eine Anleitung dazu.
- **Thermistor.h:** Diese Bibliothek dient zur Verwendung des Thermistors und wird mit der vorherigen Bibliothek installiert.

Im Vorfeld müssen die Grundplatte und das Haus 3D-gedruckt sowie die Materialien beschafft werden, welche Sie in der Stückliste finden.

Die Beinchen des Tasters, des Buzzers, des Schalters und des Potentiometers sollten mit den in der Stückliste aufgeführten DuPont-Crimpstiften verlängert bzw. verdickt werden, sodass die Bauteile optimal mit den Jumperkabeln verbunden werden können.

Partnerarbeit wird empfohlen.

Software / Programmieroberfläche

Die Arduino IDE ist eine benutzerfreundliche Entwicklungsumgebung, die speziell für die Programmierung von Arduino-Mikrocontrollern entwickelt wurde. Sie ermöglicht Schülerinnen und Schülern den einfachen Einstieg in die Welt der Mikrocontrollerprogrammierung und bietet gleichzeitig genügend Funktionen für fortgeschrittene Projekte. Die IDE ist kostenlos und auf verschiedenen Plattformen (Windows, macOS, Linux) verfügbar.

Funktionen der Arduino IDE

Die Arduino IDE kombiniert die wichtigsten Werkzeuge, die für die Entwicklung von Projekten erforderlich sind:

- **Code-Editor:** Ein übersichtlicher Editor für das Schreiben von Programmen in C/C++.
- **Kompilierung:** Ein Compiler übersetzt den geschriebenen Code in maschinenlesbaren Code, der vom Arduino ausgeführt werden kann.
- **Upload:** Das Hochladen des Programms auf das angeschlossene Arduino-Board erfolgt über einen USB-Anschluss.
- **Serieller Monitor:** Ein Tool zur Überwachung von Daten, die vom Arduino gesendet oder empfangen werden, was besonders nützlich für Debugging und Sensorwertanalysen ist.

Vorteile der Arduino IDE

Die Arduino IDE bietet eine intuitive Benutzeroberfläche, die sich ideal für Einsteiger eignet. Dank zahlreicher Beispiele und einer aktiven Community stehen umfangreiche Lernressourcen zur Verfügung, die den Einstieg erleichtern. Vorinstallierte Bibliotheken ermöglichen die Nutzung von Sensoren, Aktoren und komplexen Funktionen ohne tiefgehende Programmiererfahrung.

Erste Schritte mit der Arduino IDE

1. Laden Sie die Arduino IDE von der offiziellen Website herunter und installieren Sie sie: <https://www.arduino.cc/en/software>.
2. Schließen Sie das Arduino-Board an den Computer an.
3. Wählen Sie unter Werkzeuge das entsprechende Board (z. B. „Arduino UNO“) und den richtigen Port aus.
4. Schreiben Sie ein einfaches Programm, z. B. zur Steuerung einer LED.
5. Laden Sie das Programm auf das Board hoch und beobachten Sie die Ergebnisse.

Programmieren in der Arduino IDE

Die Programmierung erfolgt in C/C++ und verwendet eine einfache Struktur, die aus zwei Hauptfunktionen besteht:

- **void setup():** Enthält den Code, der einmalig beim Start des Programms ausgeführt wird (z. B. Initialisierung von Pins als In- oder Output).
- **void loop():** Beinhaltet den Code, der kontinuierlich ausgeführt wird (z. B. Abfragen von Sensoren oder das Ansteuern von Aktoren).

Die einfache Syntax und der logische Aufbau der Programme machen die Arduino IDE ideal für den Einsatz im Unterricht, da sie den Schülerinnen und Schülern die Möglichkeit bietet, schnell Ergebnisse zu erzielen und eigene Ideen umzusetzen.

Ziel des Kurses, Aufbau und Inhalt

Die Schülerinnen und Schüler sollen ein grundlegendes Verständnis für Smarthome-Technologien und deren Relevanz in verschiedenen Kontexten, insbesondere in außergewöhnlichen Umgebungen wie dem Weltall, entwickeln. Ziel ist es, sowohl die technischen als auch die praktischen Aspekte solcher Systeme zu verstehen. Im Mittelpunkt des Kurses steht die schrittweise Entwicklung eines Smarthome-Modellhauses, in dem die Schülerinnen und Schüler Sensoren und Aktoren programmieren, um verschiedene Automatisierungsfunktionen wie Lichtsteuerung oder Temperaturregelung zu realisieren. Dabei erlernen sie grundlegende Programmiertechniken wie die if-Abfrage, um Bedingungen zu prüfen, und die for-Schleife, um wiederholende Aufgaben effizient zu lösen. Diese technischen Fertigkeiten werden praxisnah vermittelt und ermöglichen es den Schülerinnen und Schülern, ein funktionsfähiges Smarthome-System zu entwickeln.

Zu Beginn des Kurses wird das Konzept des Smarthomes ausführlich eingeführt. Die Schülerinnen und Schüler lernen, wie smarte Technologien Prozesse wie Lichtsteuerung, Temperaturregelung, Sicherheitsüberwachung oder Energieeinsparung automatisieren. Sie erhalten Einblicke in die Vorteile, Herausforderungen und potenziellen Einsatzbereiche solcher Systeme.

Anschließend wird eine Überleitung zu Smarthome-Technologien im Weltall geschaffen. In diesem Abschnitt wird verdeutlicht, wie essenziell solche Technologien in Raumstationen oder auf zukünftigen Weltraumbahnhöfen sind, wo die Bedingungen fundamental anders sind als auf der Erde:

- **Klimaregulierung:** Da Lüften nicht möglich ist, übernehmen smarte Systeme die Kontrolle über Luftzirkulation, Temperatur und Feuchtigkeit.
- **Sauerstoffaufbereitung:** In geschlossenen Systemen wie der ISS muss der Sauerstoff künstlich erzeugt und recycelt werden.
- **Wassermanagement:** Wasser ist begrenzt und muss durch Filtration und Wiederaufbereitung recycelt werden.

Diese theoretische Einführung legt den Grundstein für den praktischen Teil des Kurses, in dem die Schülerinnen und Schüler eigene Smarthome-Anwendungen entwickeln.

In der Praxisphase beginnen die Schülerinnen und Schüler mit einer schrittweisen Einführung in die Programmierung. Nach einem Überblick über die Arduino-Hardware und die Entwicklungsumgebung erstellen sie ihr erstes Programm, das die interne LED des Arduino steuert. Im Anschluss wird das Programm erweitert, um eine externe LED über ein Breadboard zu steuern.

Nach der Grundlagenvermittlung programmieren die Schülerinnen und Schüler angeleitet klassische Smarthome-Funktionen in ihrem Modellhaus, wie:

- **Anwesenheitssimulation:** Ein Blinkmuster simuliert die Anwesenheit von Personen im Haus.
- **Klingel:** Ein Taster und ein Buzzer werden zu einem einfachen Klingelsystem kombiniert.
- **Dimmbare Lichtsteuerung:** Ein Potentiometer regelt die Helligkeit einer LED.
- **Automatische Lichtsteuerung:** Ein Lichtsensor steuert die LED abhängig von der Umgebungshelligkeit.
- **Heizungssteuerung:** Ein Temperatursensor aktiviert eine simulierte Heizung.
- **Automatische Rollläden:** Ein Schrittmotor wird verwendet, um Rollläden zu steuern. Diese bewegen sich abhängig von der Umgebungshelligkeit automatisch hoch oder herunter. Zusätzlich werden Bedingungen wie „Licht ein/aus“ integriert.

Zum Kursabschluss präsentieren die Schülerinnen ihre Smarthome-Systeme, vergleichen deren Funktionen und diskutieren mögliche Erweiterungen. In einer Gruppenreflexion wird beleuchtet, welche Anforderungen Smarthome-Systeme im Weltall im Vergleich zur Erde erfüllen.

Folgende Beispiele können dazu eingebracht werden:

- **Automatische Warnsysteme:** LEDs und akustische Signale informieren über kritische Werte wie Temperatur oder Sauerstofflevel. Diese könnten lebensrettend sein, insbesondere in geschlossenen Systemen wie Raumstationen.
- **Simulation der Luftqualität:** Ein Lichtsensor könnte verwendet werden, um Umweltbedingungen wie Luftreinheit oder Helligkeit zu simulieren. Solche Sensoren könnten den Alltag in einem Smarthome oder einer Raumstation unterstützen.
- **Erweiterte Steuerung:** Schrittmotoren könnten spezifische Aufgaben übernehmen, wie das Öffnen und Schließen von Luken. Dies zeigt, wie präzise Motorsteuerungen für komplexere Anwendungen eingesetzt werden können.

Die abschließende Diskussion regt die Schülerinnen und Schüler dazu an, über die Zukunft smarter Technologien nachzudenken und ihre Bedeutung für außergewöhnliche Lebensräume zu bewerten.