

ARD-SET02

joy-it



INHALTSVERZEICHNIS

1. Allgemeine Informationen	3
2. Geräteübersicht & Pin-Belegung.....	3
3. Arduino IDE	5
4. Module im Detail.....	6
4.01. Buzzer	6
4.02. Button	7
4.03. Relais.....	8
4.04. Ultraschallsensor.....	9
4.05. Potentiometer.....	10
4.06. Lichtsensor	11
4.07. LED	12
4.08. Servomotor.....	13
4.09. Soundsensor.....	14
4.10. Bewegungssensor	15
4.11. Infrarotsensor mit Fernbedienung	16
4.12. 16x2 LCD-Bildschirm	17
4.13. Beschleunigungssensor	18
4.14. Temperatur- & Luftfeuchtigkeitssensor.....	19
4.15. Interfaces	20

INHALTSVERZEICHNIS

5. Projekte.....	21
5.01. Hauptprojekt	21
5.02. Soundsensor & LED	22
5.03. Relais & Button	22
5.04. Bewegungssensor & Buzzer	22
5.05. Potentiometer & Servomotor.....	22
5.06. Lichtsensor & LED	22
5.07. Ultraschallsensor & Buzzer	23
5.08. Infrarotsensor & 16x2 LCD-Bildschirm	23
5.09. Temperatur- & Luftfeuchtigkeitssensor & 16x2 LCD-Bildschirm.....	23
5.10. Beschleunigungssensor, 16x2 LCD-Bildschirm & Button.....	23
6. Informations- & Rücknahmepflichten.....	24
7. Support	25

1. ALLGEMEINE INFORMATIONEN

Sehr geehrte*r Kunde*in,

vielen Dank, dass Sie sich für unser Produkt entschieden haben. Im Folgenden zeigen wir Ihnen, was bei der Inbetriebnahme und der Verwendung zu beachten ist.

Sollten Sie während der Verwendung unerwartet auf Probleme stoßen, so können Sie uns selbstverständlich gerne kontaktieren.

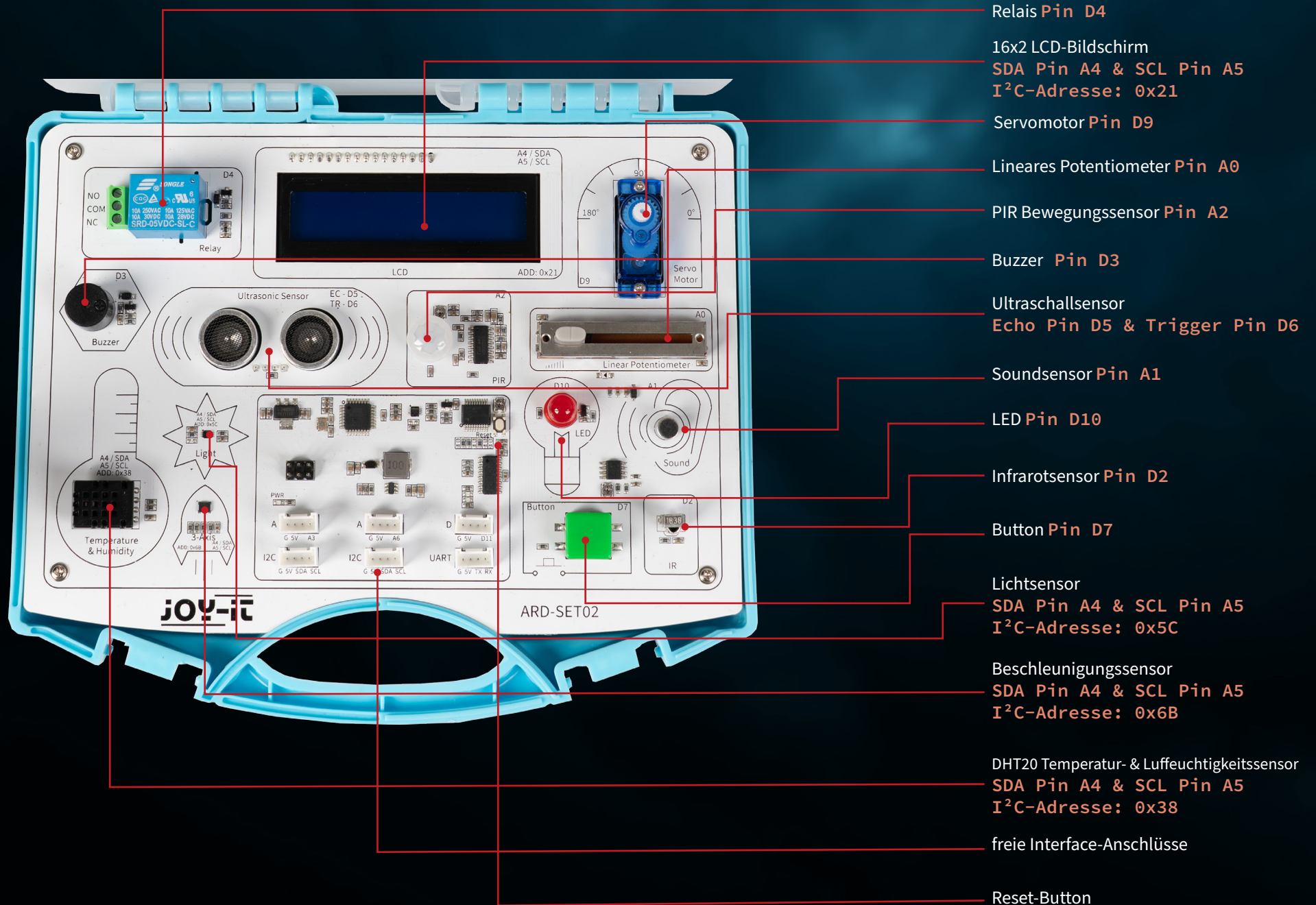
Diese Anleitung wurde unter der Arduino IDE 2.3.5 geschrieben.

2. GERÄTEÜBERSICHT & PIN-BELEGUNG

Das All-in-One-Mikrocontroller-Lernkit ist eine umfassende und einsteigerfreundliche Lernplattform, die Benutzern die Welt der Elektronik, Programmierung und eingebetteten Systeme näherbringen soll. Das kompakte Kit ist in einem robusten Kunststoffgehäuse untergebracht und verfügt über eine vollständig integrierte Hauptplatine mit einem ATmega328P-kompatiblen Mikrocontroller. Dank seines geschlossenen und vorverdrahteten Designs ist keine zusätzliche Hardware-Montage erforderlich, sodass Benutzer sofort mit dem Lernen und Experimentieren beginnen können. Da die wichtigsten Komponenten bereits integriert sind, sparen Sie Zeit und Mühe beim Verkabeln. Das All-in-One-Mikrocontroller-Lernkit verfügt über eine breite Palette an Sensoren und Aktoren, sodass Sie Ihre Projekte direkt mit einer Vielzahl von Modulen starten können. Mit dem integrierten Breadboard lassen sich eigene Projekte schnell aufbauen und realisieren.

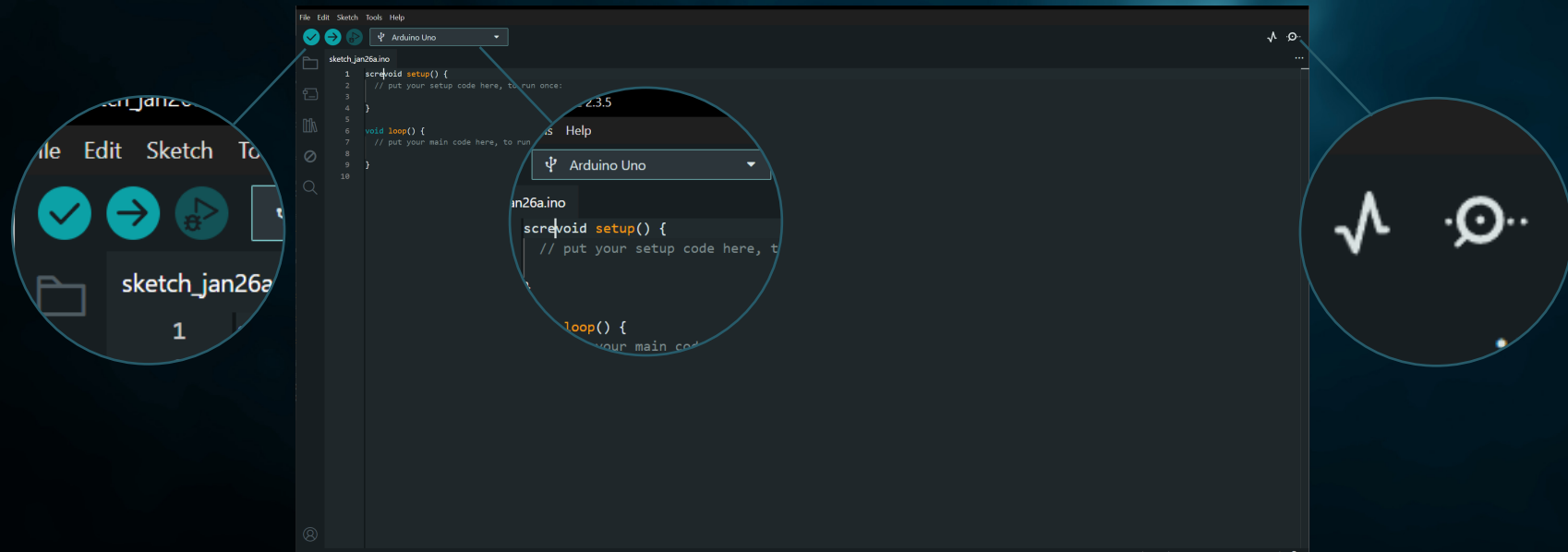
Eine Vielzahl von Sensoren und Aktoren ist bereits in das System integriert, sodass wichtige Mikrocontroller-Konzepte wie digitale und analoge Eingänge, Signalverarbeitung und Peripheriesteuerung praktisch erforscht werden können. Zu den typischen Anwendungen gehören die Messung von Umgebungsdaten, die Erkennung von Bewegungen und Entfernungen, die Steuerung von Ausgängen wie LEDs, Summern, Relais und Servomotoren sowie die Anzeige von Informationen auf einem integrierten LCD-Display. Diese Vielfalt macht das Kit sowohl für strukturierten Unterricht als auch für kreatives, projektbasiertes Lernen geeignet.

Die Programmierung und Stromversorgung erfolgt über einen bequem zugänglichen USB-C-Anschluss am Gehäuse. Diese moderne Schnittstelle gewährleistet eine zuverlässige Verbindung und vereinfacht den täglichen Gebrauch in Klassenzimmern, Labors oder zu Hause.



3. ARDUINO IDE

Dank der ATmega328P-kompatiblen Architektur kann das Kit mit bekannten Entwicklungsumgebungen programmiert werden, wie der Arduino IDE. Die können Sie sich [hier](#) herunterladen.



Schließen Sie zur Programmierung nun das beigelegte USB-C-Kabel an Ihren Computer an.

ACHTUNG! In der Arduino IDE müssen Sie (wie in der Abbildung) den richtigen Port und das richtige Board auswählen. Für das All-in-One-Mikrocontroller-Lernkit wählen Sie als Board den Arduino Uno aus.

Sie können den Code mittels des **Upload-Buttons** auf das Lernkit übertragen.

Der **serielle Monitor** dient dazu, Daten zwischen dem Lernkit und dem Computer auszutauschen. In den Beispielen wird die serielle Schnittstelle mit **Serial.begin(9600);** im **setup()** gestartet, wobei **9600** Baud die Übertragungsgeschwindigkeit festlegt. Der serielle Monitor kann in der Arduino IDE über **das Lupen-Symbol** oben rechts geöffnet werden. Damit die Ausgabe korrekt angezeigt wird, muss im seriellen Monitor ebenfalls die Baudrate **9600** eingestellt sein.

Von der Grundkonfiguration bis hin zur Ausführung von Projekten – in dieser Anleitung begleiten wir Sie durch den gesamten Prozess. Unsere Anleitung umfasst leicht verständliche Erklärungen und nützliche Tipps, um Ihnen zu helfen, Ihre Fähigkeiten im Umgang mit Mikrocontrollern schnell und effektiv zu entwickeln.

4. MODULE IM DETAIL

Im Folgenden werden alle Module, die auf dem All-in-One-Mikrocontroller-Lernkit vorhanden sind, einzeln mit Beispielcodes erläutert.



HIER können Sie alle Beispielcodes für die Verwendung einzelner Module herunterladen.

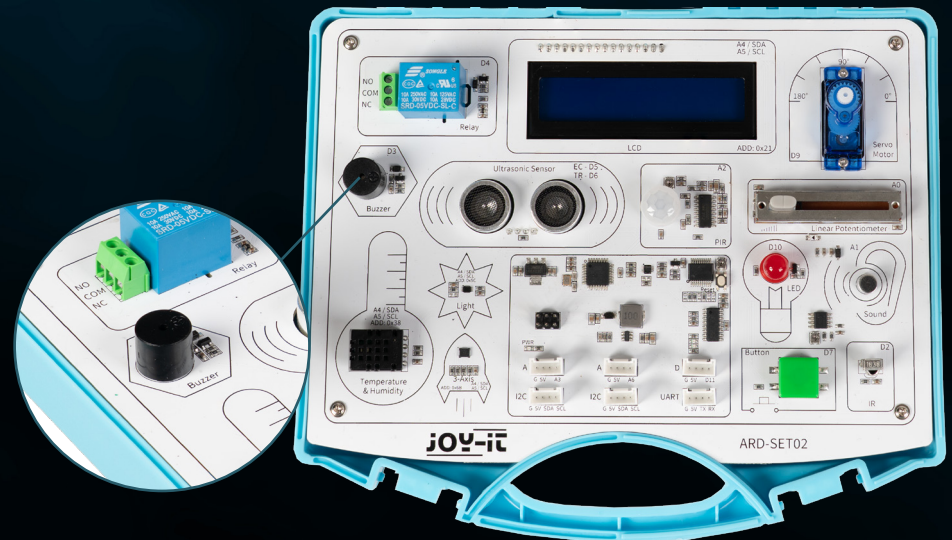
Für die Verwendung einiger Module werden externe Bibliotheken benötigt. Laden Sie sich die Bibliotheken über die IDE mittels des Bibliothekenmanagers herunter. Dort können Sie mittels der Suchfunktion die angegebenen Bibliotheken installieren.

4.01. BUZZER

Ein Buzzer oder Summer erzeugt einen Signalton, ähnlich wie ein Lautsprecher. Im Gegensatz zum Lautsprecher eignet er sich jedoch nur für einen eingeschränkten Frequenzbereich, sodass er keinen guten Klang zur Wiedergabe von Musik oder Sprache erzeugt. Dafür eignet er sich hervorragend, um laute Warntöne in Form von Piepsern zu erzeugen. Wann immer ein elektrisches Gerät einen Warnton erzeugt, übernimmt das fast immer ein Buzzer. Zum Beispiel im Wecker, Rauchmelder oder der Gurtwarner im Auto.

Der Buzzer ist an Pin D3 angeschlossen.

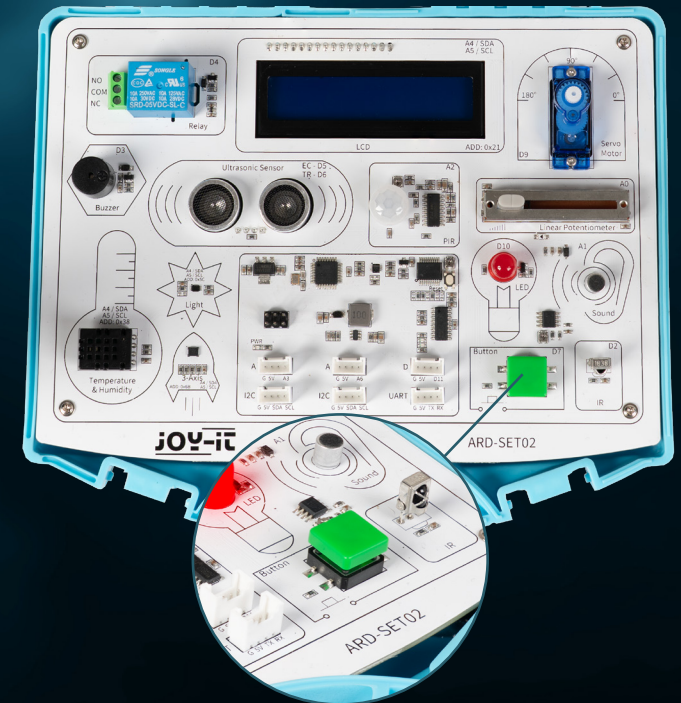
Im Codebeispiel werden die Lieder Für Elise von Ludwig van Beethoven und Frère Jacques von Jean-Philippe Rameau auf dem Buzzer abgespielt. Dazu wurden die verschiedenen Frequenzen für jede abspielbare Note definiert, sowie die zeitliche Abfolge der Noten des jeweiligen Liedes.



4.02. BUTTON

Buttons sind interaktive Elemente in Benutzeroberflächen, die eine einfache, aber essenzielle Funktion erfüllen: die Benutzereingabe. Sie dienen dazu, eine breite Palette von Befehlen und Aktionen in digitalen Umgebungen zu initiieren.

Der Button ist an Pin D7 angeschlossen.



Im Codebeispiel wird der Taster als Eingang (Input) konfiguriert. Der Zustand dieses Eingangs kann anschließend im Programm abgefragt werden. Erkennt das Lernkit am entsprechenden Pin ein Low-Signal, weiß der Code, dass der Taster betätigt wurde. Liegt hingegen ein High-Signal an, gilt der Taster als nicht gedrückt.

4.03. RELAIS

Relais gehören zu den ältesten elektromechanischen Bauteilen und arbeiten wie elektrisch gesteuerte Schalter. Mit einem kleinen Steuersignal am Eingang lässt sich eine deutlich größere Last am Ausgang ein- oder ausschalten. Dadurch können beispielsweise Lampen, Motoren oder andere Geräte mit höherem Strombedarf sicher über einen Mikrocontroller oder ein Lernkit gesteuert werden.

Im Inneren eines Relais befindet sich eine Spule. Wird sie aktiviert, erzeugt sie ein Magnetfeld, das einen mechanischen Schaltkontakt bewegt. Dieser Kontakt verbindet oder trennt anschließend die Ausgangsanschlüsse. Sobald das Steuersignal wieder abgeschaltet wird, springt der Kontakt in seine ursprüngliche Position zurück.

Ein Relais besitzt in der Regel drei Anschlussklemmen: **COM (Common)**, **NO (Normally Open)** und **NC (Normally Closed)**. COM ist der gemeinsame Anschluss. NO ist im Ruhezustand nicht mit COM verbunden und wird erst beim Aktivieren des Relais geschlossen. NC ist im Ruhezustand mit COM verbunden und wird beim Aktivieren getrennt.

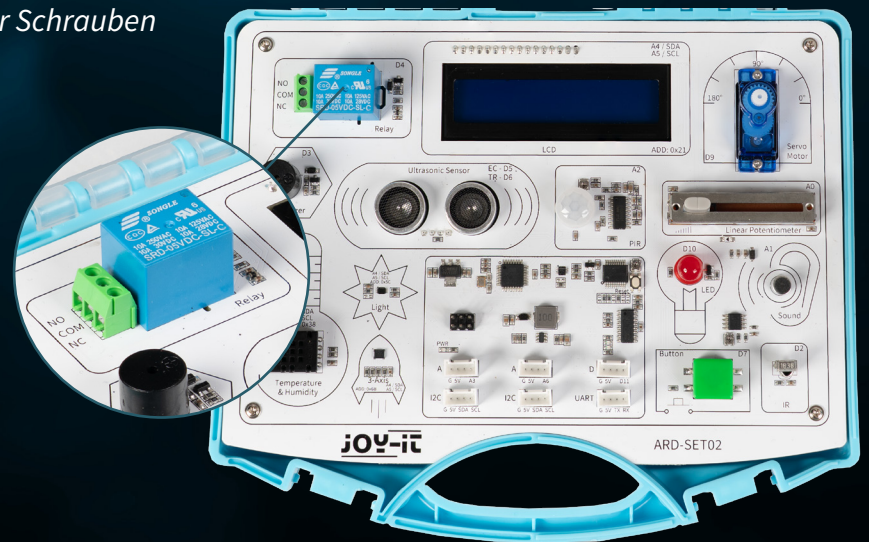
In die Klemmbuchse können abisolierte Kabelenden eingeführt und durch Festziehen der Schrauben sicher fixiert werden. So lassen sich die drei Anschlüsse einfach und zuverlässig nutzen.

Das Relais ist an Pin D4 angeschlossen.

Achtung! Arbeiten an elektrischen Anlagen mit Spannungen ab 60 Volt dürfen ausschließlich von qualifizierten Elektrofachkräften durchgeführt werden. Personen ohne entsprechende Ausbildung wird dringend empfohlen, nur die auf dem Board verfügbaren Kleinspannungen 3 & 5 V mittels des Relais zu schalten. Unsachgemäße Handhabung kann zu schweren Verletzungen bis zum Tode durch Hitze, Brandentwicklung oder Stromschlag führen.

Bitte halten Sie sich an die Sicherheitsvorschriften und wenden Sie sich bei Unsicherheiten an einen Fachmann.

Im Codebeispiel wird das Relais als Ausgang (Output) eingerichtet. Wird der zugehörige Pin aktiviert, zieht das Relais an und der Kontakt wechselt seine Stellung: Die Verbindung zwischen COM und NC wird getrennt, während COM mit NO verbunden wird. Wird der Pin wieder deaktiviert, fällt das Relais in die Ausgangsposition zurück. Dabei wird COM erneut mit NC verbunden und die Verbindung zu NO getrennt.

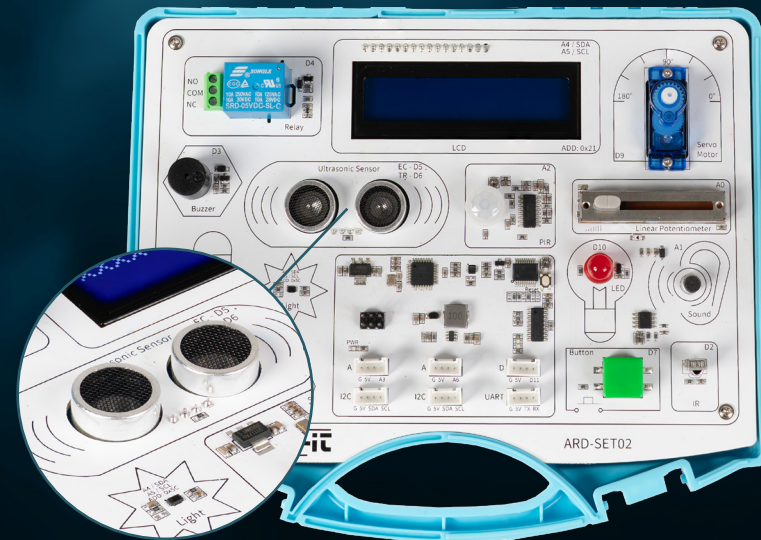


4.04. ULTRASCHALLSENSOR

Der Ultraschallsensor kann mittels Ultraschall Distanzen messen. Dazu besitzt der Sensor einen Trigger und einen Echo Pin. Mittels des Trigger-Pins sendet der Sensor ein Ultraschallsignal aus. Daraufhin wartet man bis der Echo-Pin darauf reagiert, dass der Ultraschallsensor das Signal wieder empfangen hat. Aufgrund der vergangen Zeit kann daraus eine Distanz berechnet werden.

Der Ultraschallsensor ist mit dem Echo-Pin an Pin D5 und mit dem Trigger-Pin an Pin D6 angeschlossen.

Wichtig! Der Ultraschallsensor kann nur von 2 cm bis 3m genaue Distanzen messen. Messungen welche diesen Bereich über- oder unterschreiten sind nicht zuverlässig.



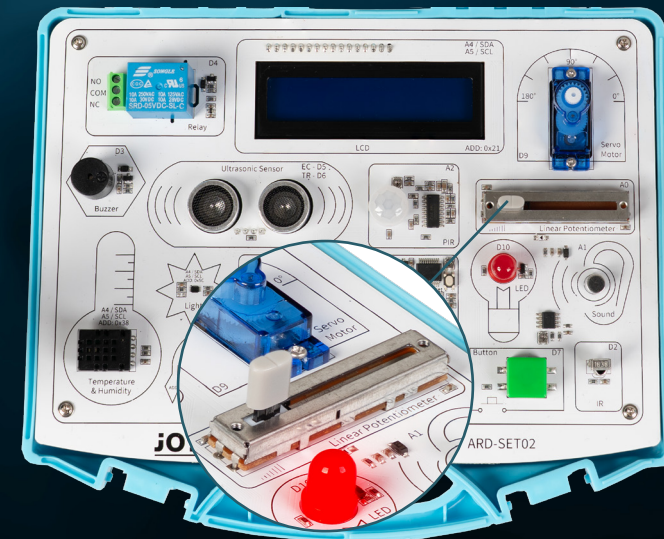
Im Codebeispiel wird der Trigger-Pin als Output und der Echo-Pin als Input deklariert. Zunächst wird der Trigger-Pin für 10 μ s auf eine hohe Spannung gesetzt, um die Messung zu triggern. Danach wartet der Code darauf, dass der Echo-Pin einen hohen Wert empfängt, und dabei wird die vergangene Zeit abgespeichert. Mit dieser Zeit kann die Distanz berechnet werden. Es folgt nun eine Kontrolle, ob die berechnete Distanz im gültigen Wertebereich liegt, und diese wird dann ausgegeben.

4.05. POTENTIOMETER

Ein lineares Potentiometer ist ein einstellbarer Widerstand, mit dem sich elektrische Werte stufenlos verändern lassen. Es wird häufig zur Einstellung von Spannung, Lautstärke, Helligkeit oder Position verwendet.

Das Kit misst am analogen Eingang die vom Potentiometer gelieferte Spannung. Diese Spannung hängt direkt von der Stellung des Potentiometers ab und verändert sich linear zwischen 0 V und 5 V. Der integrierte Analog-Digital-Wandler (ADC) des Kits wandelt diese Spannung in einen digitalen Wert zwischen 0 und 1023 um. Dadurch kann die aktuelle Position des Potentiometers einfach im Programm ausgewertet und für Steuerungs- oder Regelaufgaben genutzt werden.

Das Potentiometer ist an Pin A0 angeschlossen.



Im Codebeispiel wird der Pin A0 als Input deklariert und in einer Schleife analog ausgelesen. Der ausgelesene Wert befindet sich zwischen 0 und 1023 und wird über die Konsole ausgegeben.

4.06. LICHTSENSOR

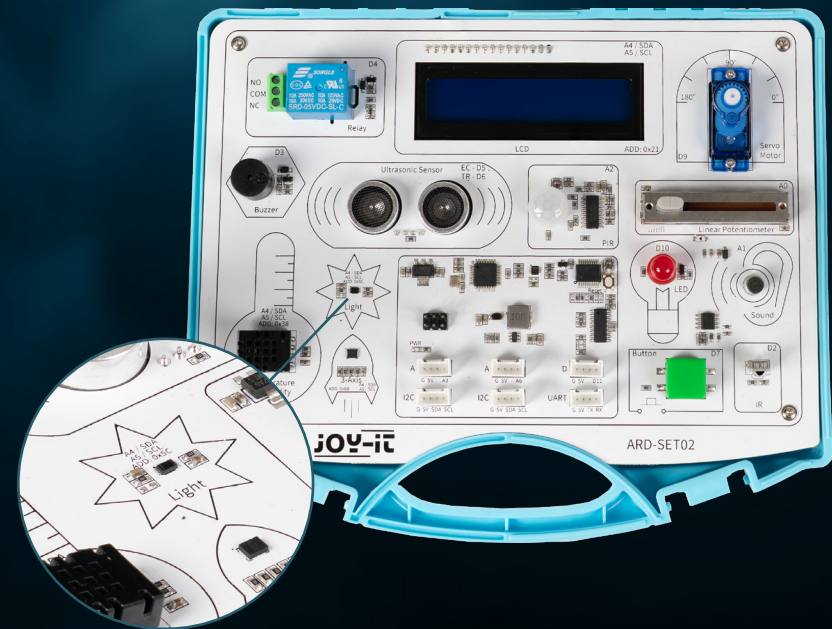
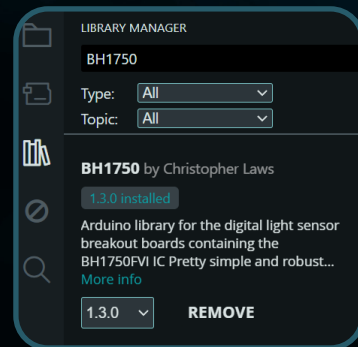
Der BH1750 ist ein digitaler Lichtsensor, mit dem die Umgebungshelligkeit sehr einfach gemessen werden kann. Der Messwert wird direkt in Lux ausgegeben, was ihn besonders anfangersfreundlich macht – es sind keine Umrechnungen nötig.

Der Sensor wird über den I²C-Bus mit dem Kit verbunden. Dazu finden Sie in Kapitel **4.15 INTERFACES** mehr. Der BH1750 eignet sich ideal für Projekte wie automatische Lichtsteuerungen, Display-Helligkeitsregelungen oder Umgebungslicht-Messungen.

Der Lichtsensor ist über I²C angeschlossen und kann mittels der Adresse 0x5C angesprochen werden.

Der Lichtsensor wird im Codebeispiel mittels der **BH1750**-Bibliothek angesteuert. Diese wurde von Christopher Laws erstellt und unter MIT-Lizenz veröffentlicht.

Sie können die Bibliothek im Library Manager unter BH1750 finden und installieren.



Im Codebeispiel wird die Kommunikation mit dem Lichtsensor über die Bibliothek aufgesetzt. Dazu werden der Modus der Kommunikation, die I²C-Adresse und die I²C-Kommunikation an die Bibliothek übergeben. Danach kann man über die Methode **readLightLevel()** den gemessenen Lux-Wert ausgeben lassen.

4.07. LED

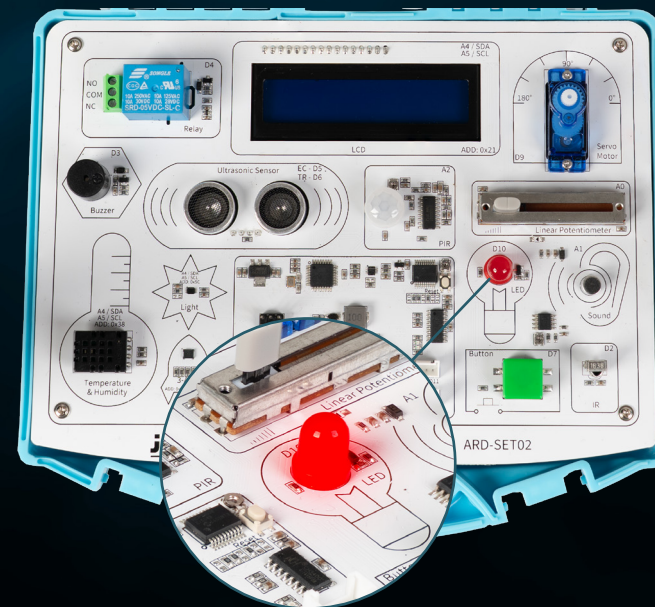
Eine rote LED ist eine der einfachsten Möglichkeiten, um Ausgaben des Kits sichtbar zu machen. Sie wird häufig verwendet, um Zustände anzuzeigen, zum Beispiel Ein/Aus, Statusmeldungen oder Warnhinweise. Eine LED eignet sich ideal für Experimente und hilft dabei, Programme und Schaltungen schnell und einfach zu testen.

Die LED kann digital oder via PWM angesteuert werden:

Wird der digitale Ausgang im Programm auf HIGH gesetzt, fließt Strom durch die LED und sie leuchtet. Bei LOW ist der Stromkreis unterbrochen und die LED erlischt.

Mit der **PWM-Steuerung (Pulsweitenmodulation)** kann die Helligkeit einer LED mit dem Kit stufenlos eingestellt werden. Obwohl ein digitaler Ausgang nur ein oder aus kennt, lässt sich so eine gleichmäßige Helligkeitsregelung realisieren. Das Kit schaltet dazu den Ausgang dabei sehr schnell ein und aus. Je länger die LED während eines Schaltzyklus eingeschaltet ist, desto heller erscheint sie. Ist sie nur kurz eingeschaltet, wirkt sie dunkler. Dies geschieht, da die Effektivspannung an der LED durch das PWM-Signal beeinflusst wird.

Die LED ist an Pin D10 angeschlossen.



Im Codebeispiel wird die LED digital an- und ausgeschaltet. Dazu wird der Pin der LED als Output deklariert.

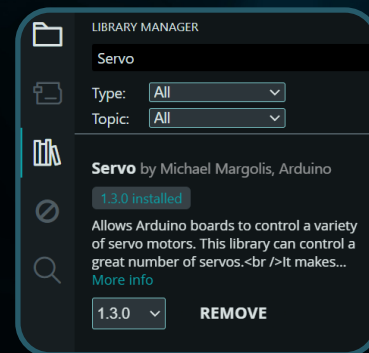
4.08. SERVOMOTOR

Ein Servo besteht aus einem Elektromotor mit Getriebe und Steuerelektronik. Auf der Ausgangsseite des Getriebes befindet sich ein Zahnrad, auf das das Servohorn montiert wird. Der Servo kann die Achse in einem Bereich von etwa 180° bewegen. Einsatzgebiete für Servos sind zum einen der Modellbau, um bei einem Flugzeug oder Schiff beispielsweise die Flügel- oder Ruderstellung zu kontrollieren. Auch im Automobilbau werden immer mehr Servos verbaut, um Türen automatisch zu schließen, für Fensterheber, Spiegel und andere verstellbare Elemente.

Ein Servomotor wird mit dem Kit über ein PWM-Signal gesteuert, um eine bestimmte Position anzufahren (0° bis 180°). Anders als bei der LED wird mit PWM hier nicht die Helligkeit, sondern der Winkel des Servos festgelegt.

Der Servomotor ist an Pin D10 angeschlossen.

Der Servomotor wird im Codebeispiel mittels der **Servo**-Bibliothek angesteuert. Diese wurde von Michael Margolis und Arduino erstellt und unter GNU Lesser General Public License v2.1 veröffentlicht. Sie können die Bibliothek im Library Manager unter Servo finden und installieren.



Im Codebeispiel wird der Servomotor über die Bibliothek angesteuert. Dazu wird der Pin an die Bibliothek gegeben. Im Beispiel dreht sich der Servomotor zwischen seinen Maximalwerten.

4.09. SOUNSENSOR

Ein Soundsensor wird verwendet, um Geräusche in der Umgebung zu erkennen, zum Beispiel Klatschen, Sprechen oder andere Schallereignisse. Er eignet sich gut für einfache Projekte wie Geräuschsteuerungen, Klatschschalter oder Lautstärkeerkennungen.

Der Sensor besteht aus einem Mikrofon und einer kleinen Auswerte-Schaltung. Es handelt sich dabei um einen analogen Sensor, das heißt, er liefert ein kontinuierliches Signal, das die Lautstärke des erfassten Geräuschs widerspiegelt. Je nach Intensität des Schalls verändert sich der ausgegebene Messwert entsprechend.

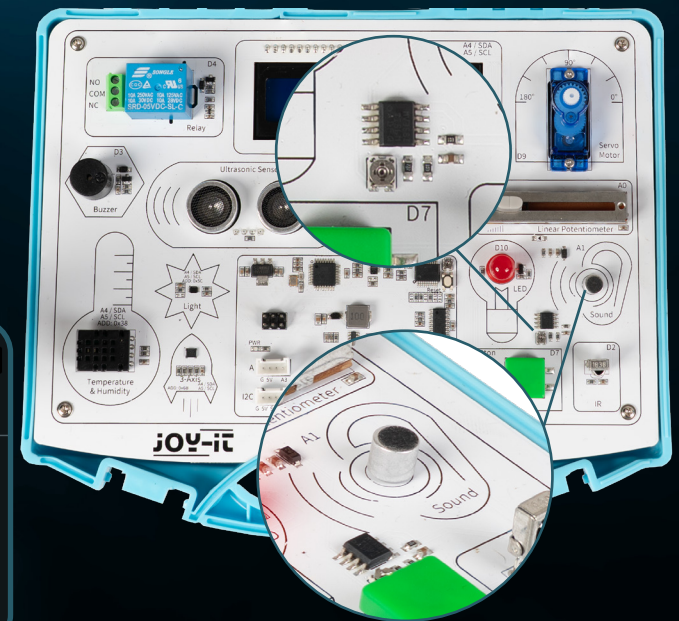
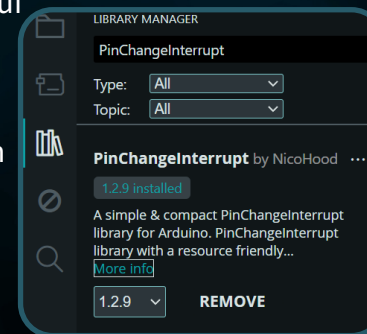
Zusätzlich verfügt das Modul über ein integriertes Potentiometer. Mit diesem kleinen Drehregler kann die Empfindlichkeit des Sensors eingestellt werden. So lässt sich festlegen, ab welcher Lautstärke der Sensor reagieren soll, was besonders bei unterschiedlichen Umgebungsgeräuschen hilfreich ist.

Der Soundsensor ist an Pin A1 angeschlossen.

Ein Soundsensor kann sehr kurze Geräuschimpulse erzeugen, zum Beispiel bei einem Klatschen oder einem Knacken. Würde das Kit den Sensor nur regelmäßig im Programm abfragen, könnten diese kurzen Signale übersehen werden.

Ein Interrupt löst sofort aus, sobald sich der Zustand eines Pins ändert (LOW $\leftrightarrow$ HIGH). Das Kit reagiert damit unabhängig vom aktuellen Programmablauf direkt auf das erkannte Geräusch.

Dazu wird die Bibliothek **PinChangeInterrupt** verwendet, da der Soundsensor nicht an einem spezifischen Pin für Interrupts angeschlossen ist. Diese Bibliothek ermöglicht es, einen Pin bei einem Port als Interrupt zu verwenden. Sie wurde von NicoHood erstellt und unter der MIT-Lizenz veröffentlicht.



Im Codebeispiel wird der Pin des Soundsensors als Input an einen Pin-Change-Interrupt gebunden. Wenn nun der Sensor ein Geräusch erkennt, führt der Code automatisch die Methode **soundRecognized()** aus.

Der Sensor selbst arbeitet weiterhin analog und liefert ein kontinuierliches Signal, das die aktuelle Lautstärke widerspiegelt. In diesem Beispiel wird das analoge Signal jedoch vom Kit nur als digitales Schaltsignal ausgewertet: Es wird also entweder High oder Low eingelesen. Überschreitet das Geräusch eine bestimmte, über das Potentiometer einstellbare Lautstärke-Schwelle, wechselt der Ausgangszustand – und genau dieser Wechsel löst den Interrupt aus.

4.10. BEWEGUNGSSENSOR

Ein PIR-Bewegungssensor (Passive Infrarot) wird verwendet, um Bewegungen von Menschen oder Tieren zu erkennen. Er reagiert dabei auf Wärmeänderungen in seinem Erfassungsbereich und eignet sich ideal für Bewegungsmelder, Alarmanlagen oder automatische Lichtsteuerungen.

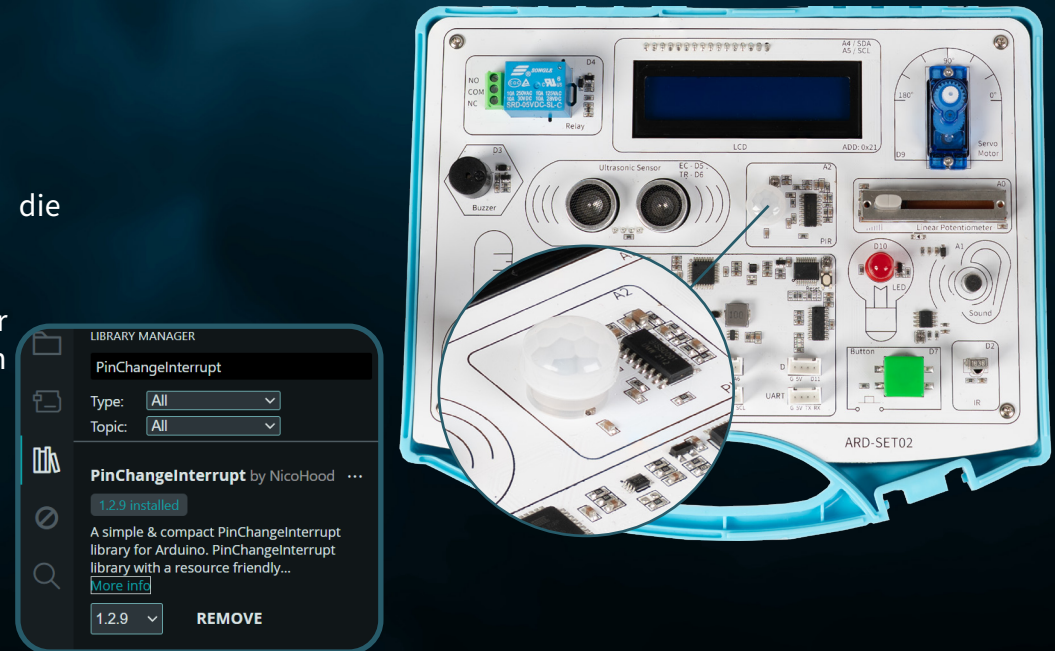
Der Sensor erkennt keine Bewegung an sich, sondern eine Veränderung der Infrarot-Wärmestrahlung, die entsteht, wenn sich ein warmes Objekt vor dem Sensor bewegt. Sobald eine Bewegung erkannt wird, schaltet der Signalausgang auf HIGH.

Beim Sensor befindet sich zusätzlich ein Potentiometer, mit dem sich die Empfindlichkeit des Sensors einstellen lässt..

Der Bewegungssensor ist an Pin A2 angeschlossen.

Aus den gleichen Gründen wie beim **SOUNDSENSOR** wird hier die **PinChangeInterrupt**-Bibliothek verwendet.

Wichtig! Der Sensor besitzt einen 2–3 Sekunden langen Cool-Down, bis er die nächste Bewegung erkennt. Zudem benötigt der PIR-Sensor nach dem Einschalten eine kurze Aufwärmzeit, bevor er zuverlässig Bewegungen erkennt.



Im Codebeispiel wird der Pin des Bewegungssensors als Input an einen Pin-Change-Interrupt gebunden. Wenn nun der Sensor eine Bewegung erkennt, führt der Code automatisch die Methode **movementRecognized()** aus.

Innerhalb dieser Methode wird der aktuelle Zustand des Pins abgefragt. Anhand dieses Zustands lässt sich erkennen, ob gerade eine Bewegung registriert wurde (Signal aktiv) oder ob der Sensor nach seiner sogenannten Cool-Down-Phase wieder in den Bereitschaftszustand zurückgekehrt ist. Während dieser kurzen Pause reagiert der Sensor nicht auf neue Bewegungen, um Fehlauslösungen zu vermeiden.

4.11. INFRAROTSSENSOR MIT FERNBEDIENUNG

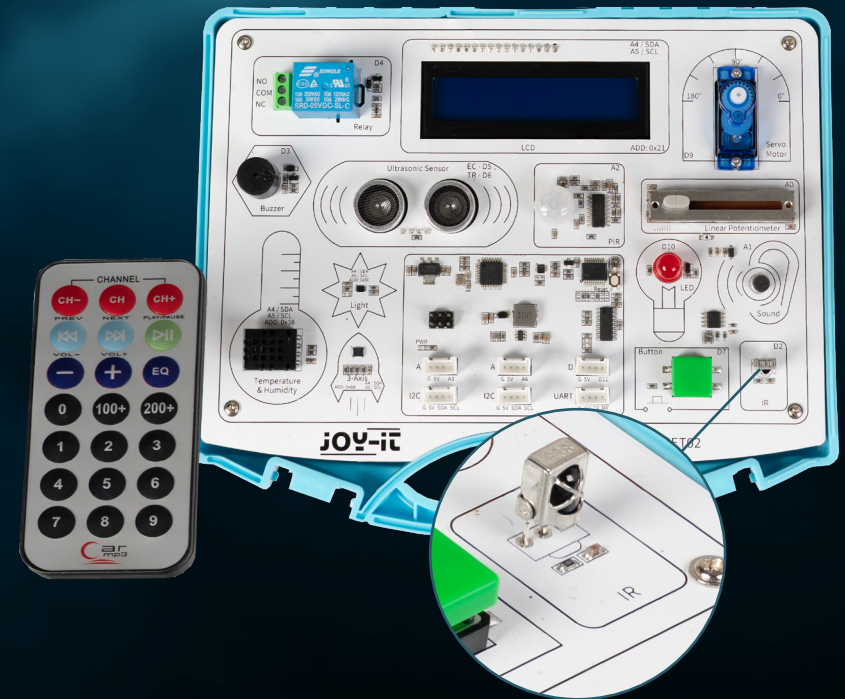
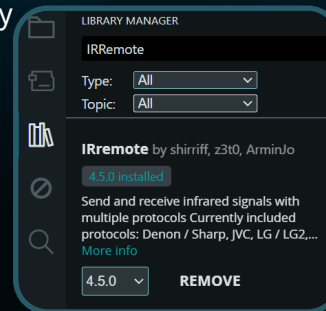
Ein Infrarotsensor (IR-Empfänger) wird verwendet, um Signale einer Fernbedienung zu empfangen und mit dem Kit auszuwerten. So lassen sich Projekte bequem kontaktlos steuern, zum Beispiel LEDs, Motoren oder Menüs.

Die Fernbedienung sendet beim Tastendruck Infrarot-Lichtimpulse mit einem bestimmten Code. Der IR-Empfänger erkennt diese Impulse und gibt ein digitales Signal an das Kit weiter. Jeder Knopf auf der Fernbedienung besitzt dabei einen eigenen Code, den das Kit unterscheiden kann.

Der Infrarotsensor ist an Pin D2 angeschlossen.

Hinweis! Um die Fernbedienung nutzen zu können, müssen Sie zunächst zwei AAA-1,5-V-Batterien in die Fernbedienung einlegen. Achten Sie dabei auf die korrekte Polarität.

Der Infrarotsensor wird in dem Codebeispiel mittels der **IRremote**-Bibliothek angesteuert. Diese wurde von shirriff, z3t0 und ArminJo erstellt und unter der MIT-Lizenz veröffentlicht. Sie können die Bibliothek im Library Manager unter IRremote finden und installieren.



Im Codebeispiel wird das Encoding der Fernbedienung mittels der Bibliothek definiert. Der Pin des Infrarotssensors wird als Input auch an die Bibliothek weitergegeben. Die Methode **decode()** gibt zurück, ob der Sensor ein Infrarotsignal empfangen hat. Über die Variable **.decodedIRData.decodedRawData** kann auf den empfangenen Wert zugegriffen werden. Die Methode **getKey()** wandelt den empfangenen Wert in den Namen der Taste der beiliegenden Fernbedienung um. Das weitere Einlesen wird durch die Methode **resume()** ermöglicht.

4.12. 16X2 LCD-BILDSCHIRM

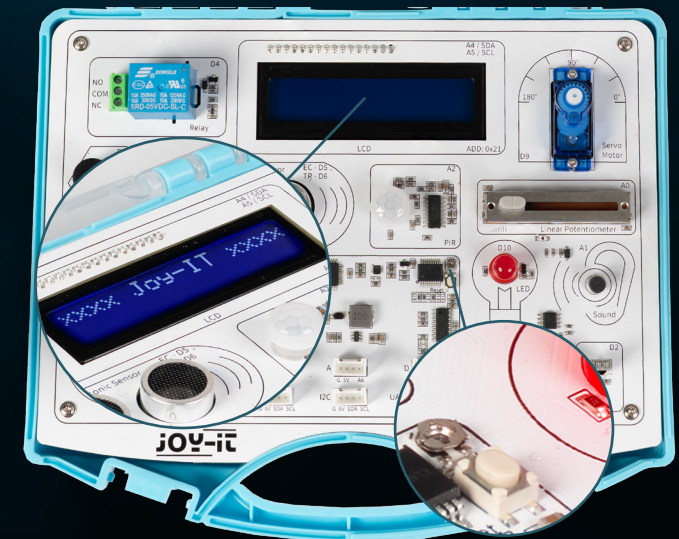
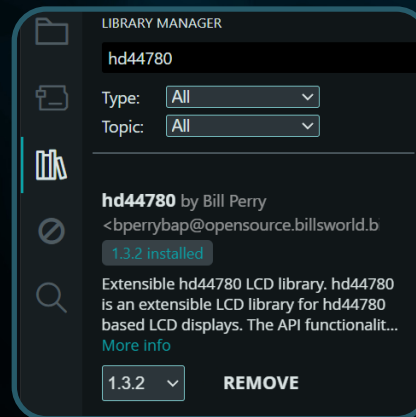
Ein 16×2 LCD-Bildschirm ist ein Textdisplay, das 16 Zeichen pro Zeile und 2 Zeilen darstellen kann. Es wird häufig verwendet, um Messwerte, Statusmeldungen oder Menüs übersichtlich anzuzeigen. Das Display basiert meist auf dem HD44780-Controller und kann direkt mit dem Kit angesteuert werden.

Die Kommunikation erfolgt über den I²C-Bus, bei dem das Display über eine feste I²C-Adresse angesprochen wird. Dazu finden Sie in Kapitel **4.15 INTERFACES** mehr.

Auf dem Kit befindet sich zudem ein kleines Potentiometer zur Kontrasteinstellung für den LCD-Bildschirm. Mit diesem Drehregler kann der Anzeige-Kontrast individuell angepasst werden. Ist der Kontrast zu niedrig eingestellt, sind die Zeichen nur schwer oder gar nicht zu erkennen; ist er zu hoch eingestellt, erscheinen dunkle Balken auf dem Display. Durch behutsames Drehen lässt sich die Anzeige optimal einstellen. Das Potentiometer ist links neben der roten LED auf dem Kit positioniert und somit leicht auffindbar.

Der 16x2 LCD-Bildschirm ist über I²C angeschlossen und kann mittels der Adresse 0x21 angesprochen werden.

Der 16×2 LCD-Bildschirm wird in dem Codebeispiel mittels der **hd44780**-Bibliothek angesteuert. Diese wurde von Bill Perry erstellt und unter GNU General Public License v3.0 veröffentlicht. Sie können die Bibliothek im Library Manager unter hd44780 finden und installieren.



Im Codebeispiel werden die I²C-Kommunikation und die I²C-Adresse des LCDs an die Bibliothek gegeben. Die Bibliothek hat viele vorgefertigte Methoden, um den Bildschirm ansteuern zu können. Im Codebeispiel finden Sie dazu mehr.

4.13. BESCHLEUNIGUNGSSENSOR

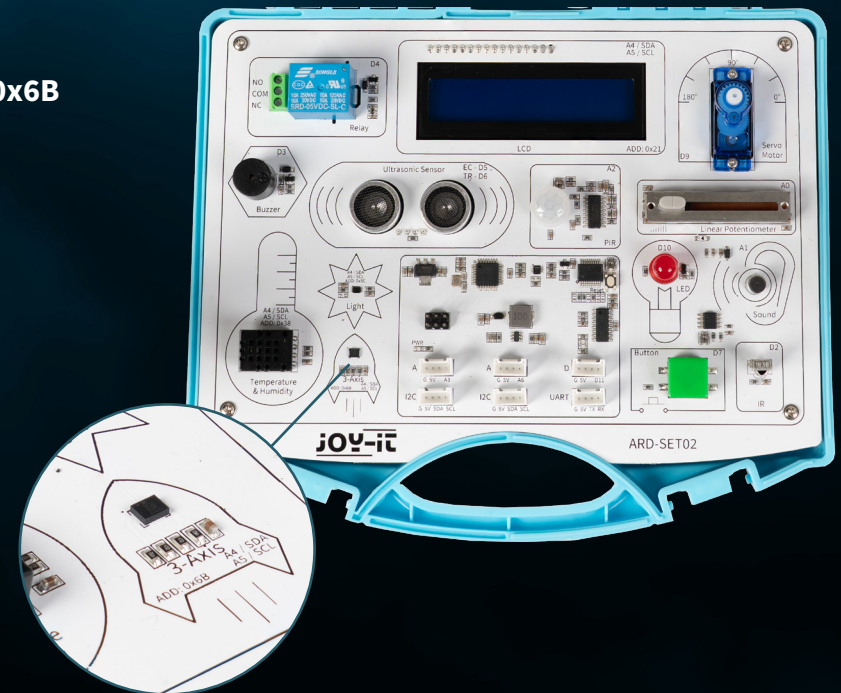
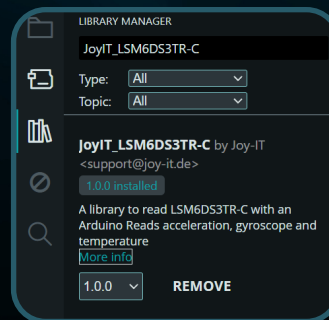
Der LSM6DS3TR-C ist ein kombinierter Beschleunigungs- und Gyroskopsensor. Er kann Bewegungen, Neigungen und Drehungen erfassen und eignet sich ideal für Projekte wie Lageerkennung, Schrittzähler, Bewegungssteuerungen oder Gestenerkennung. Der Sensor wird über den I²C-Bus mit dem Kit verbunden. Dazu finden Sie in Kapitel **4.15 INTERFACES** mehr.

Der Sensor misst die lineare Beschleunigung entlang der drei Raumachsen (x, y, z) und erfasst sowohl Bewegungen als auch die Wirkung der Erdanziehung. Dadurch lassen sich Neigungen, Lageänderungen, Erschütterungen und Stöße erkennen. Auch im Stillstand liefert der Sensor Messwerte, da die Gravitation erfasst wird.

Er misst auch die Drehgeschwindigkeit um die drei Achsen (x, y, z). Diese Werte werden in Grad pro Sekunde ausgegeben und geben Auskunft darüber, wie schnell sich der Sensor um eine Achse dreht. Damit können Drehbewegungen, schnelle Richtungsänderungen und Rotationen präzise erfasst werden. Außerdem kann der Sensor auch die Temperatur erfassen, mit der man die Temperaturabhängigkeiten der Beschleunigungs- und Gyroskopmessungen berücksichtigen kann.

Der Beschleunigungssensor ist über I²C angeschlossen und kann mittels der Adresse 0x6B angesprochen werden.

Der Beschleunigungssensor wird in dem Codebeispiel mittels der **JoyIT_LSM6DS3TR-C**-Bibliothek angesteuert. Diese wurde von uns erstellt und unter der MIT-Lizenz veröffentlicht. Sie können die Bibliothek im Library Manager unter JoyIT_LSM6DS3TR-C finden und installieren.



Im Codebeispiel wird der Beschleunigungssensor über die Bibliothek angesteuert. Die Bibliothek stellt auch Objekte bereit, in welchen die Messwerte gespeichert werden. Dabei dient **Acceleration** zur Speicherung der Beschleunigungswerte und **Gyroscope** zur Speicherung der Drehgeschwindigkeitswerte. In beiden Objekten kann man auf die Werte der einzelnen Achsen jeweils durch z.B. **.x** zugreifen. Die Temperatur wird in einem übergebenen **float** gespeichert.

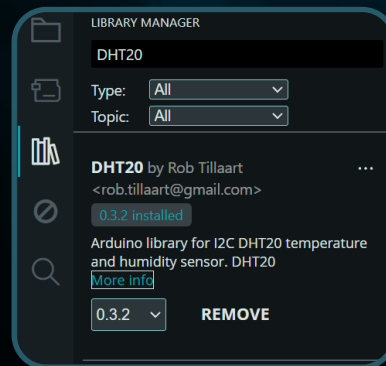
4.14. TEMPERATUR- & LUFTFEUCHTIGKEITSSENSOR

Der DHT20 ist ein digitaler Temperatur- und Luftfeuchtigkeitssensor. Er misst zuverlässig die Umgebungstemperatur sowie die relative Luftfeuchtigkeit und stellt die Messwerte digital zur Verfügung. Die Verbindung erfolgt über den I²C-Bus, wozu Sie in Kapitel **4.15 INTERFACES** mehr finden. Der DHT20 liefert die Temperatur typischerweise in Grad Celsius und die Luftfeuchtigkeit in Prozent (%) relativer Feuchte. Die Messwerte sind bereits intern aufbereitet und können direkt im Programm verwendet werden, zum Beispiel zur Anzeige auf einem Display oder zur Steuerung von Lüftern und Heizungen.

Durch seine kompakte Bauform und die einfache Ansteuerung eignet er sich ideal für Anwendungen wie Wetterstationen, Raumklimaüberwachung oder Smart-Home-Projekte.

Der Temperatur- und Luftfeuchtigkeitssensor ist über I²C angeschlossen und kann mittels der Adresse 0x38 angesprochen werden.

Der Temperatur- und Luftfeuchtigkeitssensor wird in dem Codebeispiel mittels der **DHT20**-Bibliothek angesteuert. Diese wurde von Rob Tillaart erstellt und unter der MIT-Lizenz veröffentlicht. Sie können die Bibliothek im Library Manager unter DHT20 finden und installieren.



Im Codebeispiel wird der DHT20 über die Bibliothek angesteuert. Dazu wird die I²C-Kommunikation an die Bibliothek weitergegeben. Durch den Aufruf der Methode **.read()** liest der Sensor intern die Daten aus. Mit den Methoden **.getHumidity()** und **.getTemperature()** werden dann die gemessene Feuchtigkeit und Temperatur zurückgegeben.

4.15. INTERFACES

Interface-Anschlüsse spielen eine entscheidende Rolle in der Welt der Elektronik, ähnlich wie Buttons in Benutzeroberflächen. Sie ermöglichen die Kommunikation und Stromversorgung zwischen verschiedenen elektronischen Bauteilen. Auf unserem Kit sind daher im Interface-Bereich die folgenden Anschlüsse zu finden:

I²C (Inter-Integrated Circuit): I²C ist ein serielles Zwei-Draht-Interface, das aus einer Datenleitung (SDA – Serial Data) und einer Taktleitung (SCL – Serial Clock) besteht. Über diese beiden Leitungen können mehrere Geräte miteinander kommunizieren. Dabei übernimmt ein sogenannter Master (z. B. ein Mikrocontroller wie das Lernkit) die Steuerung der Kommunikation und spricht gezielt einzelne Slave-Geräte (z.B. Sensoren & Aktoren, wie der Temperatur- und Luftfeuchtigkeitssensor) an.

Jedes angeschlossene Gerät besitzt eine eigene Adresse. Der Master sendet diese Adresse, um festzulegen, mit welchem Baustein gerade Daten ausgetauscht werden sollen. Dadurch können mehrere Sensoren, Displays oder andere Module parallel an denselben beiden Leitungen betrieben werden.

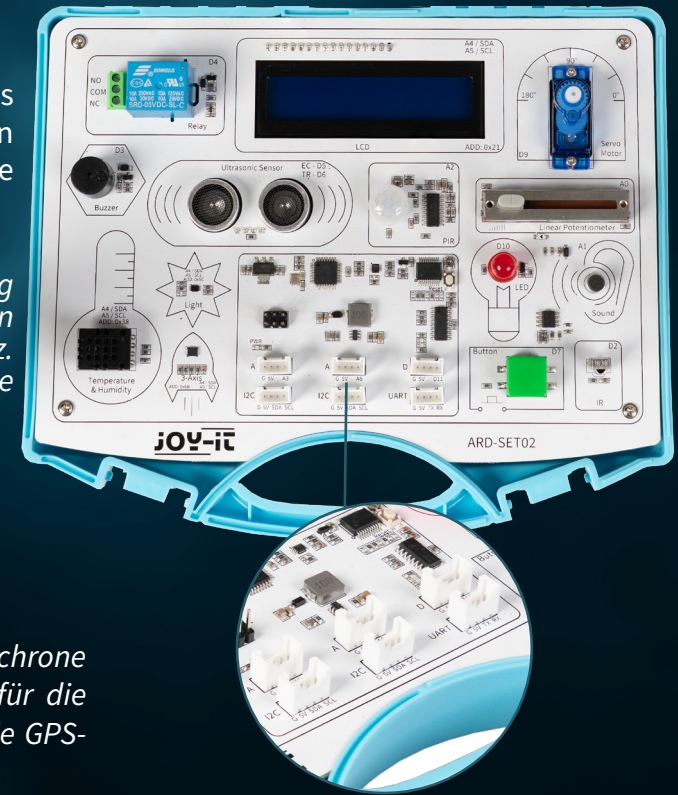
UART (Universal Asynchronous Receiver/Transmitter): Dieses Interface ermöglicht eine asynchrone serielle Kommunikation über zwei Leitungen: TX (Transmit) und RX (Receive). UART wird häufig für die Kommunikation zwischen Mikrocontrollern und Computern oder für die Anbindung von Modulen wie GPS-Empfängern oder Bluetooth-Modulen genutzt.

Weitere Anschlüsse: Es sind auch die Pins A3, A6 und D11 aus dem Kit herausgeführt worden, sodass externe Sensoren oder Aktoren auch angeschlossen werden können. Dabei sind A3 und A6 analoge Pins und können dementsprechend eine analoge Kommunikation verwenden. Sie können aber auch digital mit Sensoren oder Aktoren kommunizieren. Pin D11 ist ein digitaler Pin und kann daher nur digitale Kommunikation.

5 V Anschlüsse: Diese Anschlüsse versorgen externe elektronische Komponenten mit Strom. Das Lernkit hat einen Logikpegel von 5 V. Über diese Anschlüsse können beispielsweise Sensoren, Module oder kleine Aktoren mit Strom versorgt werden.

Das Lernkit arbeitet mit einem Logikpegel von 5 V. Das bedeutet, dass die digitalen Ein- und Ausgänge mit 5-V-Signalen arbeiten. Wenn ein Ausgang aktiviert ist (HIGH), liegt ein 5-V-Signal an, im deaktivierten Zustand (LOW) liegt ein 0-V-Signal an. Beim Anschluss externer Komponenten ist darauf zu achten, dass diese für 5-V-Logik geeignet sind.

Jeder dieser Anschlüsse hat seine spezifische Anwendung und Bedeutung in der Elektronik, ähnlich wie verschiedene Arten von Buttons in einer Benutzeroberfläche unterschiedliche Funktionen haben. Sie bieten die notwendige Flexibilität und Funktionalität für den Aufbau und die Erweiterung elektronischer Systeme.



5. PROJEKTE

Man kann die verschiedenen Sensoren und Aktoren miteinander kombinieren. Dazu stellen wir verschiedene Projektideen mit Beispielcode zur Verfügung. Es gibt jedoch auch noch viele weitere Möglichkeiten, diese Sensoren und Aktoren zusammen zu verwenden.

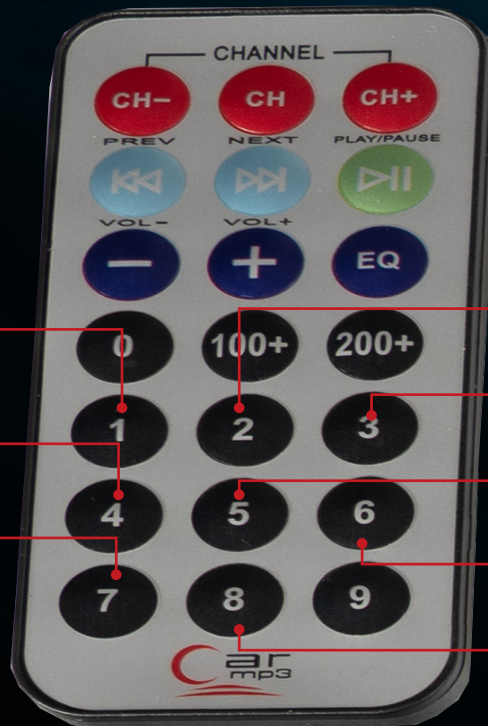


HIER können Sie sich alle Codebeispiele der Projekte herunterladen.

5.01. HAUPTPROJEKT

Dieses Codebeispiel kombiniert alle Sensoren und Aktoren des Kits in einem Code. Dabei kann man mittels der Fernbedienung über den Infrarotsensor die verschiedenen hier aufgeführten Projekte auswählen.

Dabei kann man mit den folgenden Tasten die Projekte auswählen.



Potentiometer & Servomotor
KAPITEL 5.05

Ultraschallsensor & Buzzer
KAPITEL 5.07

Temperatur- & Luftfeuchtigkeitssensor & 16x2 LCD-Bildschirm
KAPITEL 5.09

Lichtsensoren & LED
KAPITEL 5.06

Relais & Button
KAPITEL 5.03

Soundsensor & LED
KAPITEL 5.02

Bewegungssensor & Buzzer
KAPITEL 5.04

Beschleunigungssensor,
16x2 LCD-Bildschirm & Button
KAPITEL 5.10



Dieses Codebeispiel ist deutlich komplexer als die anderen Projekte, da es zu Konflikten mit den zur Verfügung stehenden Timern des Kits kam. Dazu wurde für den Infrarotsensor ein anderer Import verwendet als in dem einzelnen Codebeispiel. Dieses Projekt eignet sich gut zum Ausprobieren des Codes, jedoch ist es deutlich komplexer nachzuvollziehen.

5.02. SOUNSENSOR & LED

Bei diesem Projekt ändert man den Zustand der LED, sobald der Soundsensor ein Geräusch erkennt. Dies kann man zum Beispiel durch Klatschen triggern. Probieren Sie zunächst aus, dieses Projekt selbst zu programmieren, bevor Sie sich das Beispiel anschauen.

5.03. RELAIS & BUTTON

In diesem Projekt sollen Sie das Relais mittels des Buttons schalten, sodass Sie das Relais ausführen können, um weiteren Code zu schreiben. Probieren Sie zunächst aus, dieses Projekt selbst zu programmieren, bevor Sie sich das Beispiel anschauen.

In dem Codebeispiel wurde der Button mit der **PinChangeInterrupt**-Bibliothek angesteuert. Versuchen Sie dies auch in Ihrem eigenen Code.

5.04. BEWEGUNGSSENSOR & BUZZER

In diesem Projekt soll der Buzzer einen Ton abspielen, sobald der Bewegungssensor eine Bewegung erkennt. Dies ist zum Beispiel die Funktionsweise einer Alarmanlage. Probieren Sie zunächst aus, dieses Projekt selbst zu programmieren, bevor Sie sich das Beispiel anschauen.

5.05. POTENTIOMETER & SERVOMOTOR

In diesem Projekt soll das Potentiometer den Servomotor rotieren lassen. Das bedeutet, wenn das Potentiometer nach ganz links geschoben wird, befindet sich der Servomotor bei 0° und bei ganz rechts bei 180°. Probieren Sie zunächst aus, dieses Projekt selbst zu programmieren, bevor Sie sich das Beispiel anschauen.

5.06. LICHTSENSOR & LED

In diesem Projekt soll der Lichtsensor anhand der Helligkeit im Raum die LED heller oder dunkler leuchten lassen. Es soll wie eine Art intelligente Lichtquelle erkennen, dass, wenn es dunkel wird, das Licht heller leuchten soll, damit man noch etwas sehen kann. Probieren Sie zunächst aus, dieses Projekt selbst zu programmieren, bevor Sie sich das Beispiel anschauen.

5.07. ULTRASCHALLSENSOR & BUZZER

In diesem Projekt sollen Sie Musik über den Buzzer spielen. Dabei steuern Sie die Noten, die auf den Buzzer ertönen, anhand der gemessenen Distanz des Ultraschallsensors. Dies ist dann eine Art Theremin. Probieren Sie zunächst aus, dieses Projekt selbst zu programmieren, bevor Sie sich das Beispiel anschauen.

5.08. INFRAROTSENSOR & 16X2 LCD-BILDSCHIRM

In diesem Projekt lassen Sie sich die gedrückten Tasten der Fernbedienung, welche der Infrarotsensor empfangen hat, auf dem LCD-Bildschirm anzeigen. Probieren Sie zunächst aus, dieses Projekt selbst zu programmieren, bevor Sie sich das Beispiel anschauen.

5.09. TEMPERATUR-& LUFTFEUCHTIGUNGSSENSOR & 16X2 LCD-BILDSCHIRM

In diesem Projekt lassen Sie sich die gemessenen Werte des Temperatur- und Luftfeuchtigkeitssensors auf dem 16 × 2-LCD-Bildschirm anzeigen. Dadurch brauchen Sie keinen Computer mehr, um diese Werte einsehen zu können. Probieren Sie zunächst aus, dieses Projekt selbst zu programmieren, bevor Sie sich das Beispiel anschauen.

5.10. BESCHLEUNIGUNGSSENSOR, 16X2 LCD-BILDSCHIRM & BUTTON

In diesem Projekt lassen Sie sich die gemessenen Werte des Beschleunigungssensors auf dem 16 × 2 LCD-Bildschirm anzeigen. Dadurch brauchen Sie keinen Computer mehr, um diese Werte einsehen zu können. Da nicht alle Werte auf den Bildschirm passen, sollten Sie den Button verwenden, um durch die Beschleunigungswerte, die Drehgeschwindigkeitswerte und die Temperatur zu wechseln. Probieren Sie zunächst aus, dieses Projekt selbst zu programmieren, bevor Sie sich das Beispiel anschauen.

6. INFORMATIONS- & RÜCKNAHMEPFLICHTEN

UNSERE INFORMATIONS- UND RÜCKNAHMEPFLICHTEN NACH DEM ELEKTROGESETZ (ELEKTROG)

SYMBOL AUF ELEKTRO- UND ELEKTRONIKGERÄTEN:

Diese durchgestrichene Mülltonne bedeutet, dass Elektro- und Elektronikgeräte nicht in den Hausmüll gehören. Sie müssen die Altgeräte an einer Erfassungsstelle abgeben. Vor der Abgabe haben Sie Altbatterien und Altakkumulatoren, die nicht vom Altgerät umschlossen sind, von diesem zu trennen.

RÜCKGABEMÖGLICHKEITEN:

Als Endnutzer können Sie beim Kauf eines neuen Gerätes, Ihr Altgerät (das im Wesentlichen die gleiche Funktion wie das bei uns erworbene neue erfüllt) kostenlos zur Entsorgung abgeben. Kleingeräte bei denen keine äußere Abmessung größer als 25 cm sind, können unabhängig vom Kauf eines Neugerätes in haushaltsüblichen Mengen abgeben werden.

MÖGLICHKEIT RÜCKGABE AN UNSEREM FIRMENSTANDORT WÄHREND DER ÖFFNUNGSZEITEN:

SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

MÖGLICHKEIT RÜCKGABE IN IHRER NÄHE:

Wir senden Ihnen eine Paketmarke zu, mit der Sie das Gerät kostenlos an uns zurücksenden können. Hierzu wenden Sie sich bitte per E-Mail an service@joy-it.net oder per Telefon an uns.

INFORMATIONEN ZUR VERPACKUNG:

Verpacken Sie Ihr Altgerät bitte transportsicher, sollten Sie kein geeignetes Verpackungsmaterial haben oder kein eigenes nutzen möchten kontaktieren Sie uns, wir lassen Ihnen dann eine geeignete Verpackung zukommen.

7. SUPPORT

Wir sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen, stehen wir Ihnen auch per E-Mail, Telefon und Ticket-System zur Seite.

E-Mail: service@joy-it.net

Ticket-System: <http://support.joy-it.net>

Telefon: +49 (0)2845 9360 – 50

Für weitere Informationen besuchen Sie unsere Website:

[WWW.JOY-IT.NET](http://www.joy-it.net)