

PROGRAMMIERANLEITUNG

Micro:bit KAUGUMMISPENDER

Dieses **Micro:bit-Experimentiermodell** enthält einen **Servo** (180°), dessen winkeltgenaue Rotation mit einem Micro:bit programmiert werden kann. Die grafischen Programme, die mit **Makecode©** erstellt werden, weisen einen einfachen bis fortgeschrittenen Schwierigkeitsgrad auf.

Grundlagen

1. Allgemeine Hinweise:

- Baue die Werkpackung **Micro:bit-Kaugummispender** laut beiliegender Anleitung zusammen. Befestige einen Micro:bit mit fünf Senkkopfschrauben M3 x 8 mm an den Messinghülsen und setze zwei neue **Mignon-Batterien** (je 1,5 V) in die Batteriebox (3 V).
Achtung: Schwache Batterien können Fehlfunktionen des Servos verursachen. Das Anlegen höherer Spannungen als 3 V kann zur Zerstörung des Micro:bit führen!
- Aufladbare Batterien (zB. NiMH, NiCD) haben eine Spannung von 1,2 Volt und sind daher nur bedingt einsetzbar. Ideal und nachhaltig wäre aber die Verwendung von **USB-Powerbanks**.
- Halte den Micro:bit von Feuchtigkeit fern und berühre ihn möglichst nicht an den Kontakten.



2. Anforderungen:

Um den Micro:bit in Betrieb zu nehmen, braucht man:

- einen Laptop oder PC mit Windows 10 (8, 7) oder Mac (OSX oder Linux)
- ein Micro-USB-Kabel zum Anschluss des Micro:bit an den Computer
- einen Internet-Zugang (Chrome, Edge, Firefox, ...) - **Aber:** Für den Betrieb ohne Internet gibt es eine **App**: <https://makecode.microbit.org/offline-app>

Der Micro:bit kann auch über eine App mit einem **Tablet / iPad oder Smartphone** via Bluetooth programmiert werden. Dazu muss aber der Micro:bit mit diesen Geräten **gekoppelt** werden. Eine **Video-Anleitung** dafür findet man auf der **Homepage** von microbit.org unter: <https://microbit.org/get-started/user-guide/mobile/#pair-your-micro:bit-with-the-app>

3. Den Micro:bit vorbereiten:

Schließe den Micro:bit über ein Micro-USB-Kabel an einen freien USB-Anschluss des PCs. Das Kabel dient sowohl zur Stromversorgung des Micro:bit als auch zur Datenübertragung. Der Micro:bit erscheint im Windows Explorer (PC) oder Filemanager (Mac) als **Laufwerk** mit dem Namen **[MICROBIT]** und einem Laufwerksbuchstaben, zB. **[E:]**. Der Micro:bit kann dann über dieses Laufwerk mit einer Programmdatei (*.hex) versorgt werden.

Bei neuen Micro:bits ist ein **Demo-Programm** vorinstalliert, das die Funktionen des Micro:bit zeigt und zu verschiedenen Aktivitäten aufruft, zB. Schütteln, Neigen, Taste drücken, usw. - Es wird später einfach durch eigene Programme überschrieben!

Bei Anschluss des Micro:bit an einen Computer muss die Batterieversorgung nicht getrennt werden, denn der Micro:bit schaltet automatisch auf USB-Versorgung um.

4. Der Makecode-Editor:

Zum Programmieren verwenden wir die grafische Programmierplattform **Makecode©** von Microsoft: <https://makecode.microbit.org/>. Eine grafische Programmierung ist ideal für Anfänger, die noch keine Programmiersprache kennen, denn sie ist intuitiv und leicht zu erlernen. **Makecode©** läuft im Browser, daher braucht kein eigenes Programm installiert zu werden.

Programmierungsumgebung

1. Programmstart:

- Schließe den Micro:bit über ein **Micro-USB-Kabel** am Computer an.
- Der Micro:bit wird im Explorer als **Laufwerk** (zB. MICROBIT [E:]) angezeigt.
- Öffne den Browser (Chrome, Edge, Firefox ...) und gib folgende **Programm-URL** ein:
<https://makecode.microbit.org/>
- Wähle die Schaltfläche [Neues Projekt] aus und gib dem Projekt einen Namen (zB. **Test1**).
Nun erscheint die **Programmoberfläche**:



2. Programmbeschreibung:

Die Programmieroberfläche von **Makecode**© besteht aus drei Bereichen:
SIMULATIONSBEREICH, BEFEHLSLEISTE, PROGRAMMIERFENSTER

Im **Simulationsbereich** ist ein Micro:bit abgebildet, der das laufende Programm abspielt.

In der **Befehlsleiste** befinden sich verschiedenfarbige **Register** mit **Blöcken zum Programmieren**. Nach Anklicken der Register erscheinen diverse Blöcke, die man mit der Maus per Drag&Drop ins **Programmierfenster** ziehen kann. Die Blöcke erscheinen im Programmierfenster zuerst grau und nehmen ihre Originalfarbe erst wieder an, wenn sie richtig im Programm verankert sind. Die Blöcke lassen sich mit einem Klick der rechten Maustaste **duplizieren und löschen** oder man schiebt sie wieder zurück in die Befehlsleiste. Die Blöcke sind so geformt, dass sie nur dann ineinander passen, wenn sie logisch zu den Programmbefehlen passen. Dadurch werden Programmierfehler stark reduziert. **Fortgeschrittene** können aber statt der grafischen **Block-Programmierung** auch **JavaScript** oder **Python** verwenden.

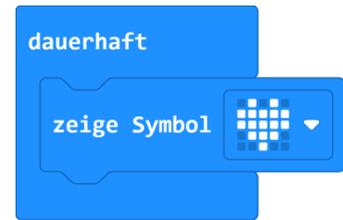
Nach einem Klick auf das **Zahnrad**symbol (rechts oben) können Einstellungen durchgeführt werden: zB. Sprache, Programme löschen, zusätzliche Blockregister einfügen usw.

Ein Klick auf das **Haus**-Symbol (oben) öffnet die **Startseite**.

3. Ein erstes Testprogramm speichern:

Lösche den **Start-Block** [beim Start] durch Verschieben in den Registerbereich. Ziehe vom Register [Grundlagen] den **Symbol-Block »Herz«** in die **Klammer** von **Block »dauerhaft«**.

Klicke unten auf das **Diskettensymbol** neben dem **Programmnamen** (Test1). Das Programm wird nun lokal auf dem Computer als **microbit-Test1.hex** gespeichert.



4. Das Testprogramm auf den Micro:bit übertragen:

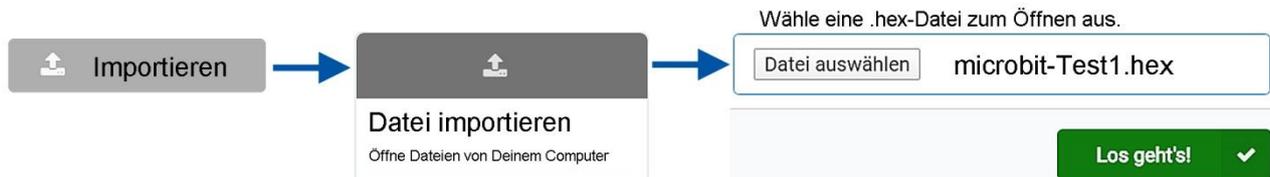
Die **Übertragung des Programms auf den Micro:bit** kann auf **zwei Arten** erfolgen:

- Öffne den Datei-Explorer und ziehe die Datei **microbit-Test1.hex** mit der Maus auf das Laufwerk [MICROBIT]. Bei der Übertragung blinkt zuerst ein gelbes Licht (Rückseite) und dann startet das Programm.
- Klicke erstmals im **Makecode-Editor** auf die Schaltfläche [Herunterladen], wähle das Laufwerk [MICROBIT] und klicke auf [Speichern]. Ab dem zweiten Mal kann jedes Programm durch einen einfachen Klick auf [Herunterladen] auf den Micro:bit übertragen werden.
- Mit der **Reset-Taste** auf der Rückseite des Micro:bit kann ein Programm neu gestartet werden.

5. Ein Programm (hex-Datei) importieren:

Um den **Programmcode einer hex-Datei** lesen und bearbeiten zu können, muss sie im Programmeditor **Makecode** geöffnet werden. Das kann auf **zwei Arten** erfolgen:

- Ziehe die entsprechende **hex-Datei** direkt **per Drag&Drop** vom Datei-Explorer auf das Programmierfenster von Makecode. Dort kann das Programm dann bearbeitet werden.
- Eine **hex-Datei** kann aber auch von der **Startseite von Makecode** importiert werden: Klicke auf die graue Schaltfläche [Importieren], dann auf [Datei importieren]. Über [Datei auswählen] kann im Datei-Explorer die gewünschte hex-Datei gewählt werden. Nach einem Klick auf [Los geht's] wird das Programm auf dem Makecode-Editor geöffnet.



6. Servo (180°):

Ein Servo (180°) ist ein elektronisch gesteuerter Getriebemotor mit einer Antriebsachse, die sich nur in einem Bereich von 180° vor- oder zurückdreht. Um diese Bewegung nutzen zu können, muss auf der Achse ein sogenanntes **Ruderhorn** oder eine runde **Ruderscheibe** befestigt werden.

Das Anschlusskabel besteht aus drei verschiedenfarbigen Drähten: **BRAUN** kommt an GND (-), **ROT** an 3 V (+) und **ORANGE** an **Pin 1** des Micro:bit. Über die orange **Signalleitung** an Pin 1 kann der Micro:bit den Servo winkeln genau von 0° - 180° steuern.

Obwohl Servos in der Regel mit 4.5 V bis 6 V betrieben werden, funktioniert der beiliegende Servo mit leicht verminderter Leistung schon ab 3 V.



Nach dieser kurzen Einführung in die Programmierumgebung von **Makecode©** folgen nun sehr einfache Programme für den Micro:bit und ab Seite 7 einfache bis leicht fortgeschrittene Programme zur Ansteuerung des Micro:bit-Kaugummispenders. Diese Programme sollen erste Programmierschritte festigen und als Basis für weiterführende, eigene Programmideen dienen.

7. Die wichtigsten Makecode-Blöcke für den Anfang:

- Aus dem Register **[Grundlagen]**:



Alle Blöcke (= Programme) innerhalb der **Start-Klammer** werden nur einmal beim **Start** ausgeführt.



Blöcke innerhalb dieser Klammer werden vom Micro:bit so lange als **Endlosschleife** ausgeführt, bis man den Strom abschaltet.



Dieser Block zeigt die eingefügte **Zahl** (hier „3“) auf der **LED-Matrix** an.



Dieser Block zeigt den eingefügten **Text** (hier „Hallo!“) am Micro:bit als **LED-Laufschrift** an.



Dieser Block zeigt das gewählte **Symbol** (hier „Herz“) als **LED-Symbol**. Mit der **Pfeilauswahl** erscheint eine Auswahl von 40 Symbolen.



Diese Funktion stellt die **Micro:bit LED-Anzeige** mit 25 LEDs dar. Durch Anklicken der dunkelblauen Felder kann man die einzelnen LEDs an- und ausschalten und damit eigene Symbole erstellen.



Durch das Einfügen eines **Pause-Blocks** wird der Programmablauf für eine bestimmte Zeit (hier 100 ms) verzögert. Die Angabe ist in **Millisekunden** (ms) → **1 Sekunde = 1000 ms**

- Aus dem Register **[Eingabe]**:



Bei einem Klick auf **Knopf A** des Micro:bit wird der Programmblock innerhalb der Klammer ausgeführt. Mit der Pfeilauswahl kann man weitere Knöpfe aktivieren: **B** und **A+B**

- Aus dem Register **[Schleifen]**:

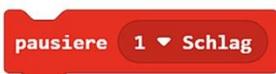


Alle Befehle (Blöcke) innerhalb des **Wiederhol-Blocks** werden in der gewählten Anzahl (hier 4-mal) wiederholt.

- Aus dem Register **[Musik]**:



Dieser Block erzeugt einen wählbaren **Ton** (hier „Mittleres C“) für eine **bestimmte Zeit** (hier 1 Schlag).



Dieser Block schaltet **Töne** für eine bestimmte Zeit (hier 1 Schlag) **aus**. An der **Pfeilauswahl** kann die „Schlagzeit“ verändert werden.

Micro:bit - Programme für Anfänger

- Programme mit dem Zusatz „**nur V2**“ laufen nur an einem **Micro:bit V2**. Mit einem kleinen, **hochohmigen Lautsprecher** (Buzzer) an **GND** und **PIN 0** des Micro:bit können aber Sound-Programme auch am **Micro:bit V1** abgespielt werden.
- Die vorgeschlagenen **Namen** der „**hex-Dateien**“ können natürlich geändert werden.

Programm 1: Begrüßung

Öffne den **Makecode-Editor** (<https://makecode.microbit.org/>), klicke auf [Neues Projekt] und gib ihm den Namen „Hallo1“.

Vorgabe: Nach dem Einschalten soll der Micro:bit einmal den Lauftext „**Hallo!**“ anzeigen und dann **dauerhaft** einen »**freundlichen Smiley**«

Programm-Code: ([microbit-Hallo1.hex](#))

Im linken **Simulationsbereich** des **Makecode-Editors** sieht man bereits eine Vorschau, was das Programm tut.

Speichere, wie auf Seite 3 beschrieben, das fertige Programm auf dem Computer.

Verbinde den Micro:bit über ein Micro-USB-Kabel mit dem Computer und übertrage das Programm ([microbit-Hallo1.hex](#)) auf den Micro:bit.



Der Lauftext „Hallo“ wird nur einmal angezeigt.



Solange der Micro:bit Strom hat, leuchtet der Smiley.

Weitere Aufgabe: Ändere den Text auf: „**Ich bin ein Microbit**“ und das Symbol auf ein „**Herz**“.

Programm 2: Herzklopfen

Vorgabe: Ein großes und ein kleines »**Herz-Symbol**« sollen jeweils **200 ms** lang abwechselnd aufleuchten.

Programm-Code: ([microbit-Herzklopfen1.hex](#))



Der **Pause-Block** lässt das „Herz“ 200 ms lang aufleuchten.

Weitere Aufgabe: Ändere die **Herzfrequenz** durch längere Pause-Zeiten (zB. 500 ms)

Programm 3: Wiederholung

Durch den Einsatz einer »**Wiederhol-Schleife**« aus dem **Register [Schleifen]** kann man die Anzahl der Wiederholungen von eingefügten Programmteilen genau festlegen.

Vorgabe: Ein kleines und ein großes »**Quadrat-Symbol**« sollen nach Betätigung von **Knopf A** **4-mal blinken**.

Programm-Code: ([microbit-Wiederholung1.hex](#))

Weitere Aufgaben:

- Ändere die **Anzahl der Wiederholungen**
- Setze an Stelle der Symbole **2 Zahlen** (zB. 0 / 1)
- Ändere die Blinkfrequenz durch Einfügen von »**Pause-Blöcken**« nach den Symbolen.

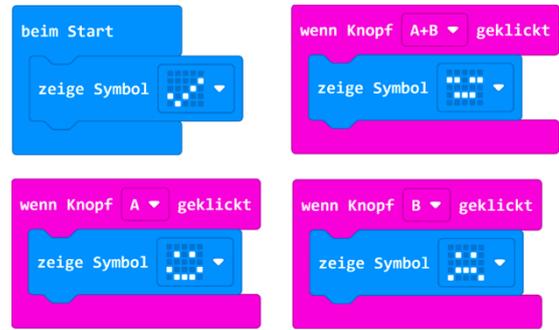


Programm 4: Knopf A und B

Vorgabe: Nach Drücken der **Knöpfe A, B und A+B** sollen verschiedene »Smileys« aufleuchten.

Programm-Code: ([microbit-KnopfAB-1.hex](#))

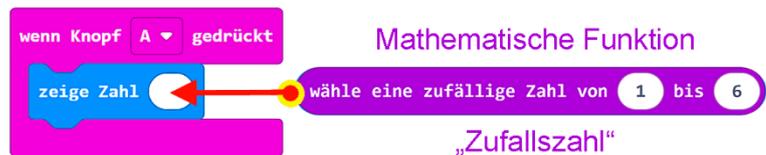
Weitere Aufgabe: Nach Drücken der **Knöpfe A, B und A+B** sollen die **Buchstaben „A“, „B“ und „C“** angezeigt werden.



Programm 5: Zahlenwürfel

Vorgabe: Nach Betätigung von **Knopf A** soll auf dem Micro:bit eine **zufällige Zahl von 1 - 6** angezeigt werden. Der Micro:bit wird dadurch zu einem praktischen **Zahlenwürfel**.

Die **mathematische Funktion »Zufallszahl«** befindet sich im **Register [Mathematik]**. Ziehe sie per Drag&Drop in das weiße Feld des **Blocks »zeige Zahl«**.



Programm-Code: ([microbit-Zahlenwürfel1.hex](#))

Programm 6: Pulston (nur V2)

Um **Töne mit Pausen** erzeugen zu können, benötigt man die **Blöcke »spiele Note ... für ...«** und **»pausiere für ...«** aus dem **Register [Musik]**.

Vorgabe: Beim Start soll ein **»X-Symbol«** aufleuchten. Nach Klicken von **Knopf A** soll die **»Note „Hohes C“** (1 Schlag) mit **»Pause«** (1 Schlag) in einer **»Wiederholschleife«** (6-mal) erklingen.

Programm-Code:
([microbit-Pulston1.hex](#))



Weitere Aufgaben:

Die Noten **„Mittleres G“** und **„Hohes C“** sollen in der **»Wiederholschleife«** ohne Pause **5-mal** ertönen. (Folgetonhorn von Einsatzfahrzeugen)

Programm 7: Klatschschalter (nur V2)

Der **Micro:bit V2** ist mit einem **Mikrofon** ausgestattet, das zur Steuerung von Programmen genutzt werden kann. Die **Ansprechempfindlichkeit (Schwellenwert)** reicht von **0 - 255**.

Vorgabe: Setze beim **»Start«** ein **»Symbol«** und stelle den **»Schwellenwert«** auf **„128“**. Der **Block »wenn laut«** aus dem Register **[Eingabe]** soll auf ein **Klatsch-Geräusch** reagieren und eine **»Melodie«** vom **Register [Musik]** im **Hintergrund** abspielen.

Programm-Code: ([microbit-Klatschschalter1.hex](#))



Micro:bit - Programme für den MB-Kaugummispender

- Für die Programmierung von Servos benötigt man das **Register [Servos]**. Wähle dazu in der Befehlsleiste das **Register [⊕ Erweiterungen]** und klicke auf das Bild „**Servo**“. In der Leiste erscheint in der Folge das **Register [Servos]**, das den **Block »setze Winkel von Servo an P ... auf ...°«** enthält.



Programm 8: Ausgaberad am Kaugummiautomat befestigen

Vor der endgültigen Positionierung des Ausgaberrades an der Servoachse muss der Winkel des Servos mit einem kleinen Programm auf einen **Winkel von 0°** eingestellt werden. Setze dann das Ausgaberad laut Bild auf die Servoachse und befestige die Plexiglasplatte mit vier Schrauben 3 x 12 mm. Befülle den Kaugummispender für erste Tests noch nicht mit Kaugummikugeln.



Vorgabe: Der »**Winkel von Servo (an P1)**« soll beim **Start** auf einen Winkel von „**0°**“ gesetzt werden.

Programm-Code: ([microbit-Rad1.hex](#))



Programm 9: Servotest

Wenn der Servo mit dem folgenden Testprogramm gut funktioniert, kann der Kaugummispender mit Kaugummikugeln befüllt werden.

Vorgabe: Setze den »**Winkel von Servo (an P2)**« mit **Knopf A** auf „**150°**“ und **Knopf B** auf „**0°**“.

Teste auch andere Winkel an **Knopf A:**
zB. 160°



Programm-Code:

([microbit-Servotest1.hex](#))

Programm 10: Kaugummispender 1

Vorgabe: Zeige ein »**Symbol**« und setze den »**Winkel von Servo**« (an P1) beim **Start** auf „**0°**“. Nach Drücken von **Knopf B** soll ein »**lächelnder Smiley**« erscheinen. Der Servo soll auf einen **Winkel** von „**150°**“ und nach einer kurzen »**Pause**« wieder auf „**0°**“ gesetzt werden:

Programm-Code: ([microbit-Spender1.hex](#))

Weitere Aufgaben: Ändere die vorgegebenen **Symbole** und **Pausezeiten!**



Programm 11: Kaugummispender mit Klatschschalter (nur V2)

Wenn man vor dem Micro:bit kräftig mit den Händen klatscht oder auf den Tisch klopft, soll der Kaugummispender eine Kaugummikugel ausgeben.

Vorgabe: Zeige beim **Start** ein »Symbol« und stelle den »Schwellenwert« auf „185“. Der **Block** »wenn laut« aus dem **Register [Eingabe]** soll auf ein **lautes Geräusch** reagieren, den **Ausruf „Hallo“** vom **Register [Musik]** abspielen und den **Servo** bewegen.

Programm-Code: ([microbit-Klatschspender1.hex](#))

Weitere Aufgaben: Ändere die vorgegebenen **Ansprechempfindlichkeit** (»Schwellenwert: 0 - 255«)

Programm 12: Kaugummispender 2 (nur V2)

Vorgabe: Zeige ein »Symbol« und setze den »Winkel von Servo« (an P1) beim »Start« auf „0“. Nach Betätigung von **Knopf A** soll ein »böser Smiley« erscheinen und der »Text „So nicht!“«. Nach Drücken von **Knopf B** soll ein »lächelnder Smiley« aufleuchten und nach kurzer »Pause« der **Servo** eine Kaugummikugel ausgeben. Bei gleichzeitigem Drücken der **Knöpfe A + B** soll zu einem neuen »Symbol« die **Melodie „Entertainer“** abgespielt werden.

Programm-Code: ([microbit-Spender2.hex](#))

Hinweis: Die Weitergabe und Vervielfältigung dieses Anleitungenheftes, auch auszugsweise, ist für den schulischen Gebrauch grundsätzlich gestattet. Eine Veröffentlichung, auch auszugsweise, oder eine entgeltliche Weitergabe bedarf der schriftlichen Genehmigung der Firma Winkler Schulbedarf.