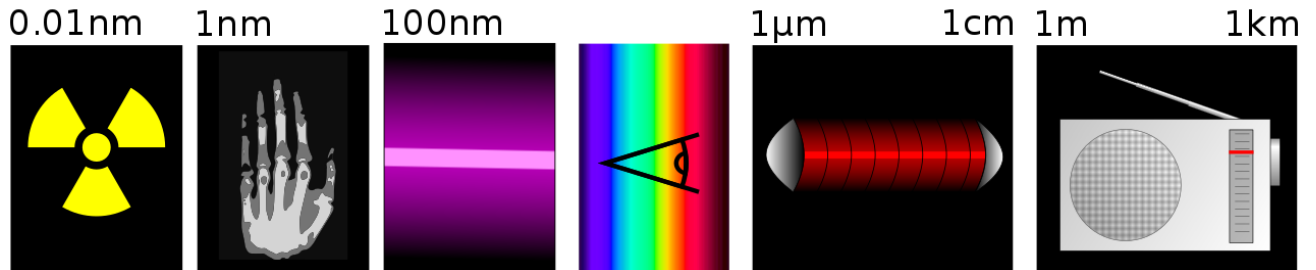


Was ist Funk?

Funk ist die Übertragung von Daten (Sprache, Musik, Bilder, Video) **ohne Kabel** – über Radiowellen. Das Wort „Funk“ kommt von „Funke“, weil vor 150 Jahren bei den ersten Funk-Experimenten künstliche Blitze die Funkwellen erzeugt haben. Damals konnte man nur Morsezeichen senden (kurze oder lange Pulse – „ein“ oder „aus“).

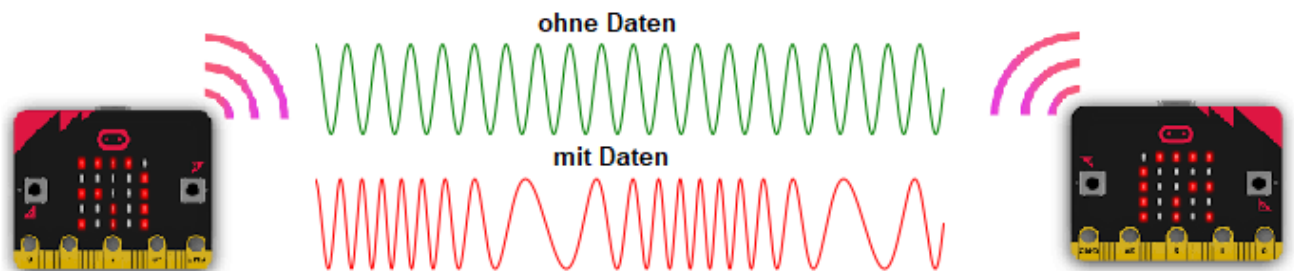
Funk ist elektromagnetische Strahlung, so wie radioaktive Strahlung, Röntgen und Licht. Diese unterscheiden sich durch ihre Wellenlänge bzw. Frequenz: **Frequenz bedeutet die Anzahl der Wellen pro Sekunde.**



So wie Licht breitet sich Funk mit der schnellsten möglichen Geschwindigkeit aus: der Lichtgeschwindigkeit (300.000km pro Sekunde – das ist ca. die Entfernung zwischen Erde und Mond). Die schnellste Rakete ist noch immer 20.000 mal langsamer und braucht 5 Stunden statt 1 Sekunde für 300.000km!

Für Funkübertragung benötigt man einen Sender und einen (oder mehrere) Empfänger. Der Sender schickt über eine Antenne Radiowellen mit einer bestimmten Frequenz. Heute werden die Informationen nicht mehr als „ein“ und „aus“ übertragen, sondern „moduliert“. Will der Sender Zahlen übertragen, dann ändert er die Frequenz ein wenig (macht sie um diese Zahl kleiner oder größer): die **Wellen werden kürzer oder länger.**

Der micro:bit kann (fast) gleichzeitig senden und empfangen: er schaltet ganz schnell hin- und her.



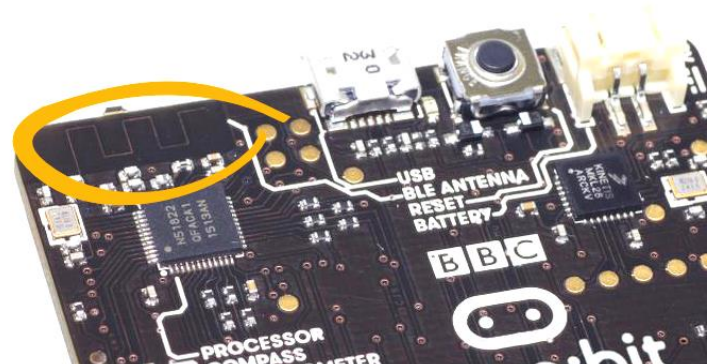
Die Frequenz, die der micro:bit zum Funken verwendet, ist sehr groß: es sind 2,4GHz (2400 Millionen Wellen pro Sekunde)

2 Wellenberge haben bei 2,4GHz einen Abstand von ca. 12cm.

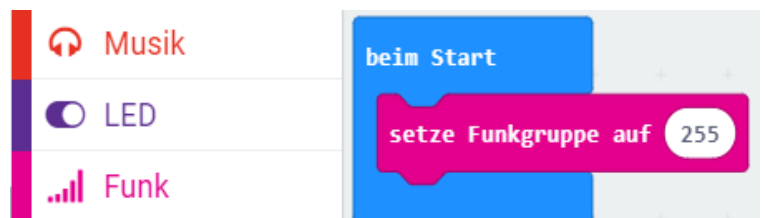
Damit die Antenne diese Wellen gut empfangen und senden kann, muss sie genau so lange wie ein Viertel davon sein:

$$12\text{cm} : 4 = 3\text{cm}$$

Auf dem micro:bit ist die Antenne als 3cm lange **Zick-Zack-Leitung** zu finden (im Bild orange eingekreist)



Damit sich die micro:bit beim Funken nicht gegenseitig stören, muss man zusammengehörige Sender und Empfänger in die **gleiche Funkgruppe** eintragen:



Zahlen über Funk schicken und empfangen: Wir programmieren das Spiel „Die heiße Kartoffel“!

Zwischen 2 micro:bit wird eine „heiße Kartoffel“ hin und her gegeben. Eine „Uhr“ zählt dabei auf 0 herunter (wie eine Sanduhr, bei der am Ende der Sand durchgelaufen ist).

Wer am Ende die heiße Kartoffel hat, hat verloren!

Statt einer echten heißen Kartoffel schicken wir eine Zahl zwischen den micro:bits in und her. Und es läuft auch keine Uhr, sondern bei jeder Weitergabe der „Kartoffel“ wird die Zahl um 1 kleiner, bis sie 0 und das Spiel zu Ende ist.

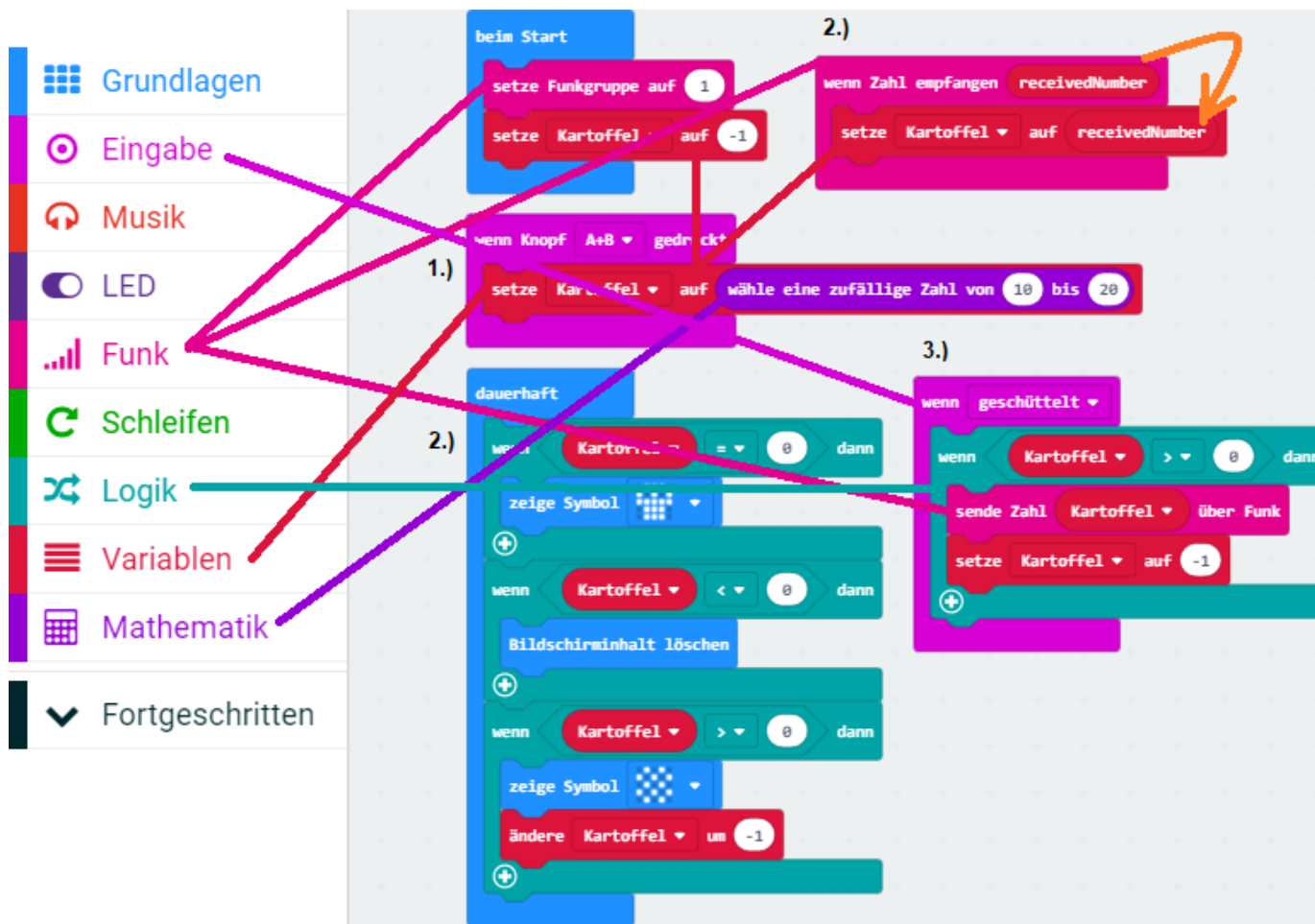


Wir brauchen also eine Variable „Kartoffel“:

- Wenn der Wert von „Kartoffel“ größer 0 ist, dann hat der Spieler die Kartoffel und der Wert bedeutet die restliche Zeit bis 0
- Wenn der Wert von „Kartoffel“ 0 ist, dann ist das Spiel zu Ende
- Ist „Kartoffel“ -1 (negative Zahl), dann bedeutet das, dass der Spieler die Kartoffel nicht hat

Was müssen wir noch programmieren?

1. Das Spiel startet, wenn auf einem micro:bit die Knöpfe A+B gleichzeitig gedrückt werden
2. Wenn man die „Kartoffel“ empfängt, dann zeigt der micro:bit ein Bild an
3. Wenn man den micro:bit schüttelt, dann wird die „Kartoffel“ zum anderen micro:bit gesendet



Erklärung „receivedNumber“: der micro:bit macht automatisch eine Variable für die empfangene Zahl, die einen englischen Namen hat. Man findet diese Variable leider auch nicht in der roten Bibliothek „Variablen“. Aber man kann mit der Maus den roten Block „receivedNumber“ (mit den runden Ecken) auf jedes Feld für Zahlen (runder Platzhalter) ziehen.

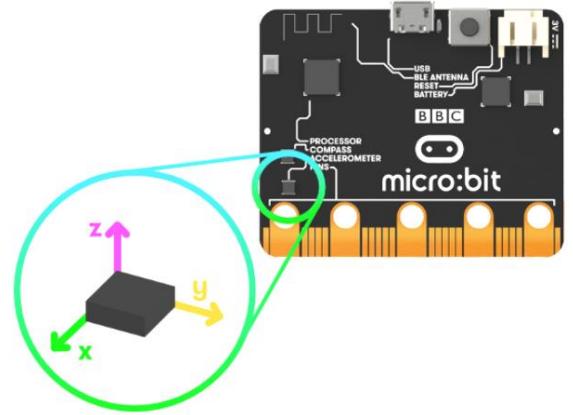
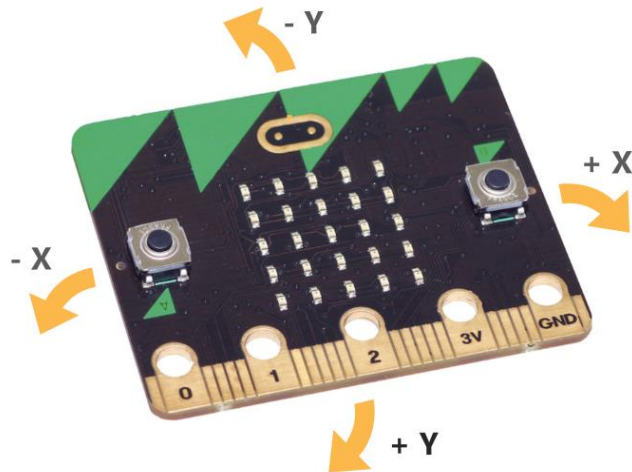


Verbesserungsvorschlag: Damit man kontrollieren kann, welche Funkgruppe auf dem micro:bit eingestellt ist, sollten wir im Block „beim Start“ auch den Befehl „zeige Zahl“ mit der Funkgruppe aufrufen. Das hilft, Fehler einzugrenzen (oft passen nämlich die Funkgruppen nicht zusammen – das sieht man sonst nicht so einfach).

Roboter MIK über Funk steuern:

Mit einem zweiten micro:bit können wir den Roboter wie mit einer Funkfernbedienung steuern. Statt der Knöpfe A und B verwenden wir den Beschleunigungssensor, der unserem Programm verschiedene Werte gibt, je nachdem wie wir den micro:bit halten:

Wenn wir ihn zur Seite kippen, ändert sich der Wert **x** und wenn wir ihn nach vorne und hinten neigen das **y**.



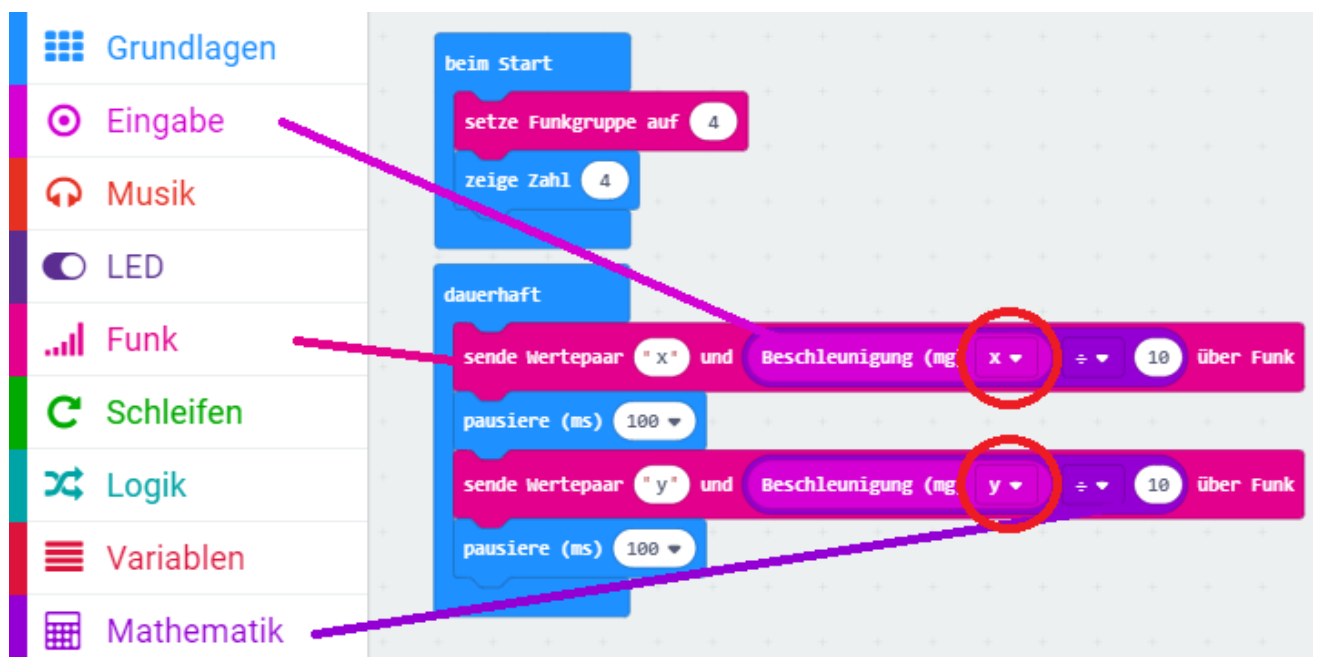
Der Beschleunigungssensor mit seinen 3 Achsen x, y, z

Je mehr wir den micro:bit kippen, desto größer werden die Zahlen. Wenn wir den micro:bit gerade halten (oder auf dem Tisch liegen lassen) dann sind beide Zahlen 0.

Im Programm für den Sender schicken wir in der Endlos-Schleife „dauerhaft“ immer diese beiden Werte x und y. Dabei machen wir jedes Mal eine Pause von 100ms, damit der micro:bit genug Zeit hat, diese Zahlen zu funken.

Weil wir 2 Zahlen schicken und der Empfänger diese unterscheiden können soll, schicken wir ein Wertepaar (einen Buchstaben und die Zahl), z.B. x = 50

Den x-Wert werden wir auf dem Roboter für das Lenken verwenden und den y-Wert zum Gas geben bzw. Bremsen.

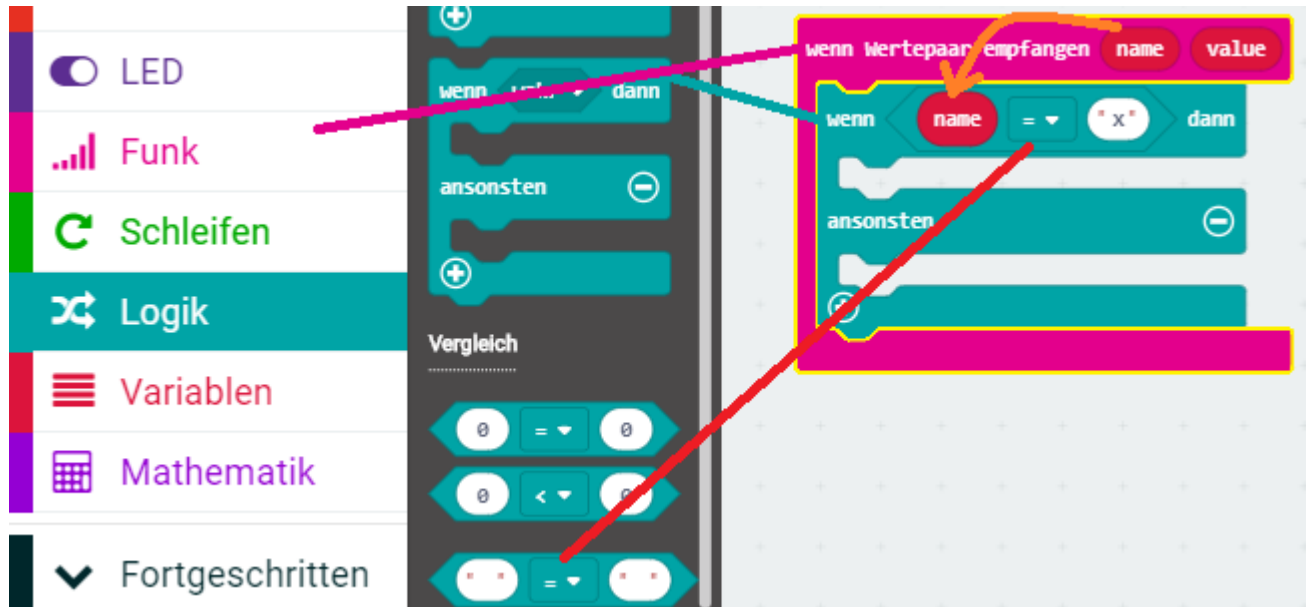


Wir programmieren jetzt den Empfang am Roboter: zuerst erstellen wir 2 Variablen x und y.

Dann brauchen wir den Block „wenn Wertepaar empfangen“ aus der Bibliothek Funk. In diesem Block müssen wir zuerst abfragen, ob wir x oder y empfangen haben.

Dafür brauchen wir den Vergleich mit den „Anführungszeichen“ aus der Bibliothek „Logik“: wir vergleichen den empfangenen Buchstaben (Variable „name“) mit „x“.

Wenn das gleich ist (=), dann setzen wir unsere Variable x auf den empfangenen Wert (Variable „value“):



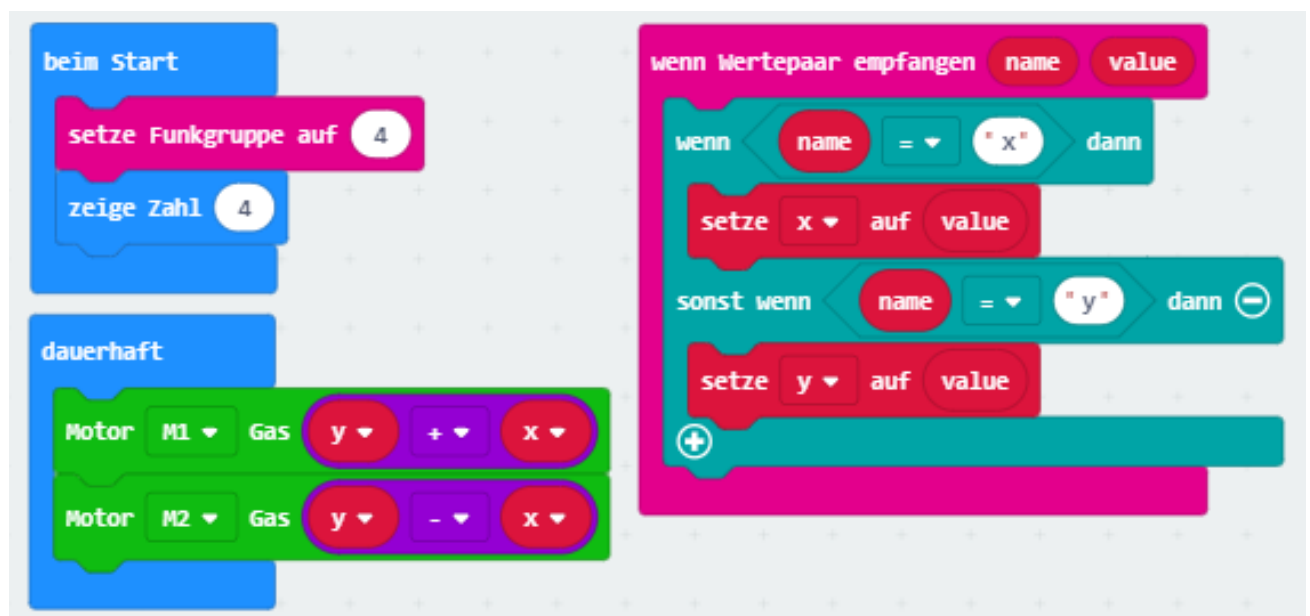
Wenn wir den Buchstaben („name“) = „y“ empfangen, dann setzen wir unsere Variable y.

Wir haben jetzt die Werte für x und y vom Sender empfangen und müssen überlegen, wie wir damit die Motoren ansteuern. Der y-Wert ist zum Gas geben, d.h. wir geben y auf beide Motoren. Ist y positiv, dann fährt der Roboter nach vorne und wenn y negativ ist, fährt er rückwärts.

Der x-Wert ist zum Lenken. Ist x positiv, dann soll der Roboter nach rechts fahren. Dazu müssen wir Motor M1 schneller drehen und M2 langsamer: für Motor M1 setzen wir Gas also auf $y + x$ und für M2 setzen wir Gas auf $y - x$.

Ist das empfangene x kleiner als 0 (negative Zahl, z.B. -10), dann fährt der Roboter nach links, weil Motor M1 jetzt die kleinere Zahl (z.B. $50 - 10 = 40$) und Motor M2 die größere Zahl ($50 - (-10) = 50 + 10 = 60$) als Gas hat.

In der Mathematik ist nämlich **zweimal Minus ein Plus**.




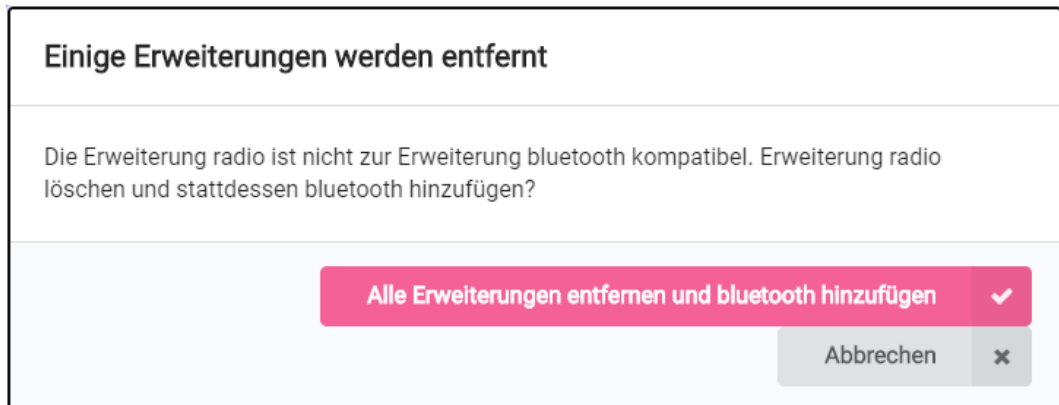
Was ist Bluetooth?



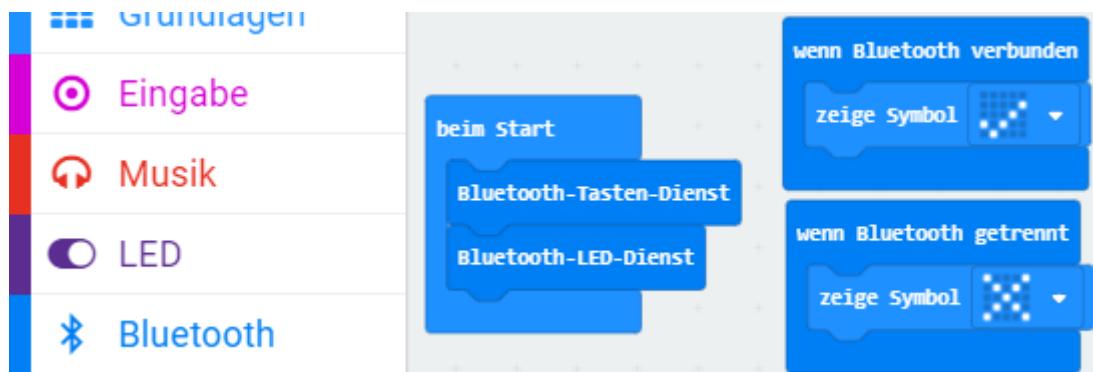
Mit dem micro:bit Funkmodul kann der micro:bit auch mit einem Handy funken. Dazu müssen aber die „Bluetooth services“ als Bibliothek genutzt werden.

Bluetooth besteht aus ziemlich komplizierten Funk-Regeln (einem „Protokoll“), die seit 20 Jahren immer wieder erweitert werden. Bevor Bluetooth-Geräte miteinander funken können, müssen sie zuerst ein geheimes Passwort (einen Schlüssel) austauschen. Man sagt dazu auch „Pairing“ (Koppeln).

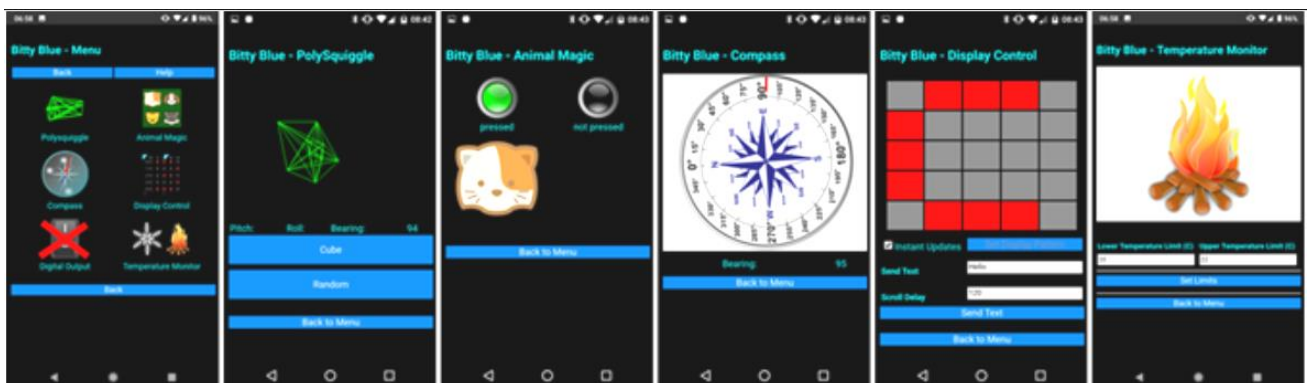
Wenn man die Erweiterung „Bluetooth“ zum Programmieren verwenden möchte, dann kommt folgende Meldung, die wir mit dem rosa Knopf  bestätigen müssen:



Unser Programm auf dem micro:bit ist sehr einfach. Wir starten den „Bluetooth-Tasten-Dienst“ und zeigen ein Häkchen wenn der micro:bit über Bluetooth verbunden ist und ein x, wenn er getrennt ist.



Mit der App „Bitty Blue“ können wir jetzt mit dem micro:bit das Handy steuern (und umgekehrt):



Es gibt viele Apps zur Steuerung des micro:bit. Damit die Steuerung funktioniert, müssen die App und das Programm auf dem micro:bit zusammenpassen!