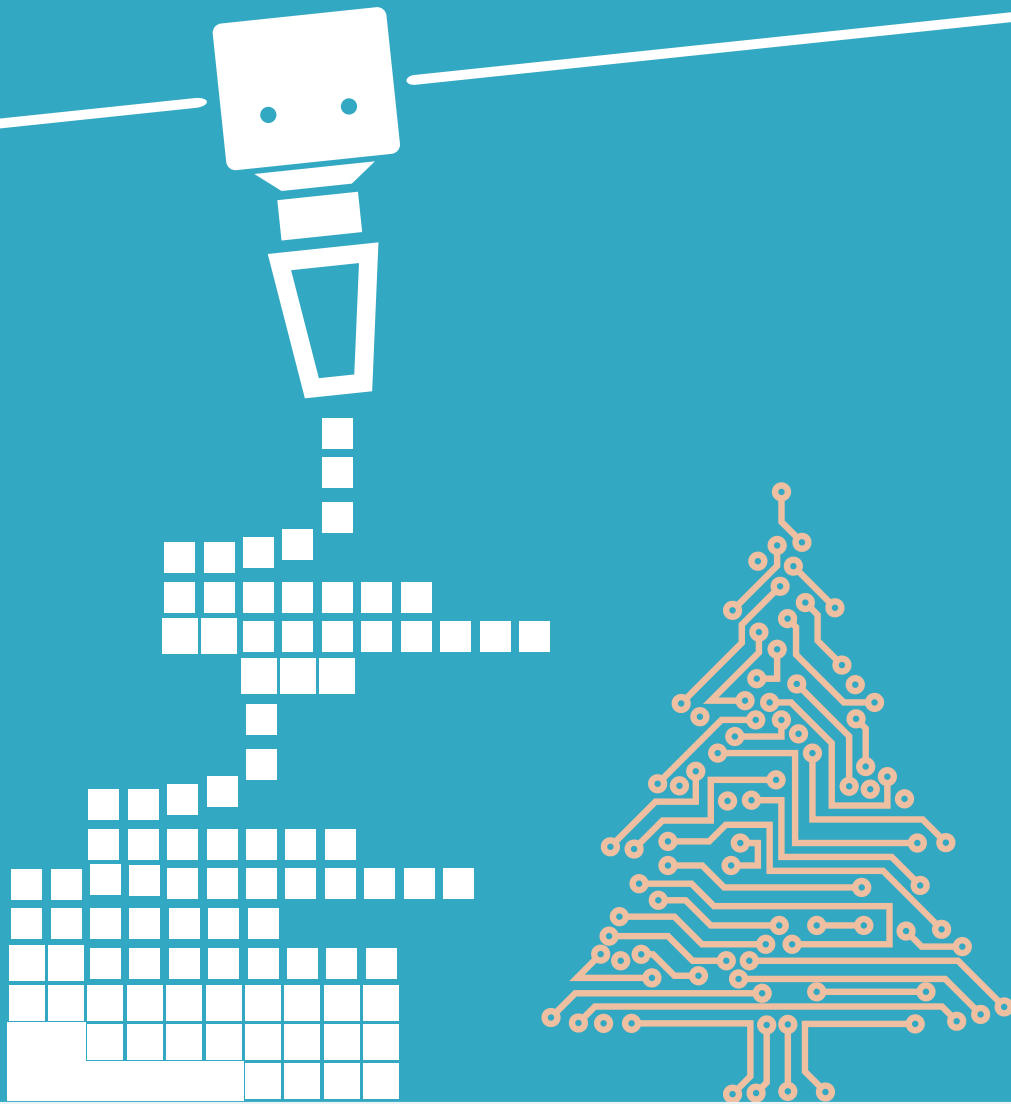


ADVENTSKALENDER

3D-DRUCK

HANDBUCH



FRANZIS

Impressum

© 2021 Franzis Verlag GmbH, Richard-Reitzner-Allee 2, 85540 Haar bei München
Änderungen, Innovationen und Druckfehler vorbehalten
www.franzis.de

2021/01

Autor: Christian Immler
Produktmanagement: Tobias Schärfl
Copy Editor: Claudia Fliedner
Coverdesign: Thomas Preischl
GTIN: 4019631672803

Besonderen Dank an Hr. K.-D. Krüger für die Mitarbeit an diesem Adventskalender!

Nicht für Kinder unter 14 Jahren geeignet.

Alle in diesem Buch vorgestellten Schaltungen und Programme wurden mit der größtmöglichen Sorgfalt entwickelt, geprüft und getestet. Trotzdem können Fehler im Buch und in der Software nicht vollständig ausgeschlossen werden. Verlag und Autor haften in Fällen des Vorsatzes oder der groben Fahrlässigkeit nach den gesetzlichen Bestimmungen. Im Übrigen haften Verlag und Autor nur nach dem Produkthaftungsgesetz wegen der Verletzung des Lebens, des Körpers oder der Gesundheit oder wegen der schuldhaften Verletzung wesentlicher Vertragspflichten. Der Schadensersatzanspruch für die Verletzung wesentlicher Vertragspflichten ist auf den vertragstypischen, vorhersehbaren Schaden begrenzt, soweit nicht ein Fall der zwingenden Haftung nach dem Produkthaftungsgesetz gegeben ist.

Alle Rechte vorbehalten, auch die der fotomechanischen Wiedergabe und der Speicherung in elektronischen Medien. Das Erstellen und Verbreiten von Kopien auf Papier, auf Datenträgern oder im Internet, insbesondere als PDF, ist nur mit ausdrücklicher Genehmigung des Verlags gestattet und wird widrigenfalls strafrechtlich verfolgt.

Die meisten Produktbezeichnungen von Hard- und Software sowie Firmennamen und Firmenlogos, die in diesem Werk genannt werden, sind in der Regel gleichzeitig auch eingetragene Warenzeichen und sollten als solche betrachtet werden. Der Verlag folgt bei den Produktbezeichnungen im Wesentlichen den Schreibweisen der Hersteller.

Liebe Kunden!



Dieses Produkt wurde in Übereinstimmung mit den geltenden europäischen Richtlinien hergestellt und trägt daher das CE-Zeichen. Der bestimmungsgemäße Gebrauch ist in der beiliegenden Anleitung beschrieben. Bei jeder anderen Nutzung oder Veränderung des Produktes sind allein Sie für die Einhaltung der geltenden Regeln verantwortlich. Bauen Sie die Schaltungen deshalb genau so auf, wie es in der Anleitung beschrieben wird. Das Produkt darf nur zusammen mit dieser Anleitung weitergegeben werden.



Das Symbol der durchkreuzten Mülltonne bedeutet, dass dieses Produkt getrennt vom Hausmüll als Elektroschrott dem Recycling zugeführt werden muss. Wo Sie die nächstgelegene kostenlose Annahmestelle finden, sagt Ihnen Ihre kommunale Verwaltung.

Warnung! Augenschutz und LEDs:

Blicken Sie nicht aus geringer Entfernung direkt in eine LED, denn ein direkter Blick kann Netzhautschäden verursachen! Dies gilt besonders für helle LEDs im klaren Gehäuse sowie in besonderem Maße für Power-LEDs. Bei weißen, blauen, violetten und ultravioletten LEDs gibt die scheinbare Helligkeit einen falschen Eindruck von der tatsächlichen Gefahr für Ihre Augen. Besondere Vorsicht ist bei der Verwendung von Sammellinsen geboten. Betreiben Sie die LEDs so wie in der Anleitung vorgesehen, nicht aber mit größeren Strömen.

Kurzschlüsse vermeiden!

Eine direkte Verbindung zwischen Minus- und Pluspol muss unbedingt vermieden werden, weil Drähte und Batterien heiß werden können und weil die Batterien sich dann schnell verbrauchen. Im Extremfall können Drähte glühend heiß werden, und die Batterie kann explodieren. Es besteht Brand- und Verletzungsgefahr. Verwenden Sie nach Möglichkeit nur normale Zink-Kohle-Batterien (6F20), die einen geringeren Kurzschlussstrom liefern und deshalb weniger gefährlich sind als Alkalibatterien (6RL61). Verwenden Sie keinesfalls Akkus!

Inhaltsverzeichnis

3D-Druck-Adventskalender für Maker..... 6

Tag 1 8

Das Konstruktionsprogramm 3D easy	8
So funktioniert 3D easy	8
Projekte.....	8
Quick Start Guide	8
Ihr Zugang zu 3D easy.....	8

Tag 2..... 9

Die Programmiersprache mBlock.....	9
Batteriekasten.....	9
mBlock 5.....	9
So einfach funktioniert das Programm	11
Konstruieren mit 3D easy.....	14

Tag 3 15

Die Blink-LED.....	15
Konstruieren mit 3D easy – die Grundplatte für das Weihnachtshaus	16

Tag 4 17

Das Nano-Board.....	17
Das Nano-Board – eine Arduino-kompatible Platine	17
Den Nano vorbereiten	18
Softwareinstallation in Kürze.....	18
Aufgabe	20
So funktioniert das Programm	20
Konstruieren mit 3D easy – die Grundplatte für das Weihnachtshaus, Teil 2	21

Tag 5 22

Zwei LEDs blinken abwechselnd	22
Aufgabe	22
Zwei LEDs blinken abwechselnd	23
So funktioniert 3D easy	23
Konstruieren mit 3D easy – das Grundgerüst für das Weihnachtshaus	23

Tag 6..... 24

Eine LED blinkt mit wechselnder Geschwindigkeit	24
Aufgabe	24
LED blinkt mit einstellbarer Geschwindigkeit	25
So funktioniert das Programm	26
Konstruieren mit 3D easy – das Dach des Weihnachtshauses.....	26

Tag 7 27

Einstellbare Wechselblink-LEDs	27
Aufgabe	27
So funktioniert das Programm	28
Konstruieren mit 3D easy – Der Schornstein des Weihnachtshauses	28

Tag 8 29

Dimmen von LEDs.....	29
Schalt draht.....	29
LEDs dimmen	29
Aufgabe	30
Das Programm	30
So funktioniert das Programm	31
Konstruieren mit 3D easy – Fenster und Türen des Weihnachtshauses	32

Tag 9 33

Erster Test für die Hausinstallation	33
Aufgabe	33
Das Programm	33
So funktioniert das Programm	35
Türklingel.....	35
Kamintaster	36
LED auf dem Dach	38
Konstruieren mit 3D easy – Halterung für das Batteriefach im Weihnachtshaus.....	38

Tag 10 39

Lauflichter.....	39
Aufgabe	39
Das Programm	39
So funktioniert das Programm	39
Konstruieren mit 3D easy – Halterung für den Piezo-Summer	40

Tag 11 41

Zufallsgenerator	41
Aufgabe	41
Das Programm	41
So funktioniert das Programm	42
Konstruieren mit 3D easy – weitere Arbeit am Weihnachtshaus	42

Tag 12	43
Widerstände und Taster	43
Widerstände und ihre Farbcodes	43
Aufgabe	44
Das Programm	45
So funktioniert das Programm	45
Konstruieren mit 3D easy – Fertigstellung des Weihnachtshauses.....	46
 Tag 13	 47
Klingel mit Piezo-Summer	47
Aufgabe	47
Das Programm	48
So funktioniert das Programm	48
Konstruieren mit 3D easy – der Tannenbaum, Teil 1.....	48
 Tag 14	 49
Nano mit dem PC steuern.....	49
Aufgabe	49
Das Programm	49
So funktioniert das Programm	51
Konstruieren mit 3D easy – der Tannenbaum, Teil 2.....	51
 Tag 15	 52
Die RGB-LED	52
Aufgabe	53
Das Programm	54
So funktioniert das Programm	54
Konstruieren mit 3D easy – der Tannenbaum, Teil 3.....	54
 Tag 16	 55
RGB-Farbmischung mit PWM	55
Aufgabe	55
Das Programm	56
So funktioniert das Programm	56
Konstruieren mit 3D easy – der Weihnachtsstern, Teil 1.....	56
 Tag 17	 57
RGB-LED leuchtet in Regenbogenfarben	57
Aufgabe	57
Das Programm	57
So funktioniert das Programm	60
Konstruieren mit 3D easy – der Weihnachtsstern, Teil 2.....	60
 Tag 18	 61
RGB-LED mit Taster.....	61
Aufgabe	61
Das Programm	61
So funktioniert das Programm	63
Konstruieren mit 3D easy – der Weihnachtsstern, Abschluss.....	63

Tag 19	64
Nano im Arduino-Modus	64
Nano mit Batterie betreiben	64
Aufgabe	64
Das Programm	65
So funktioniert das Programm	65
 Tag 20	 66
RGB-LEDs im Zufallsmodus.....	66
Aufgabe	66
Das Programm	66
So funktioniert das Programm	67
Konstruieren mit 3D easy – der Schneemann, Teil 1.....	67
 Tag 21	 68
Summer und LEDs Live-Modus.....	68
Aufgabe	68
Das Programm	69
So funktioniert das Programm	70
Konstruieren mit 3D easy – der Schneemann, Teil 2.....	71
 Tag 22	 72
Weihnachtslied auf dem PC-Lautsprecher	72
Aufgabe	72
Das Programm	73
So funktioniert das Programm	74
Konstruieren mit 3D-Easy – Der Schneemann Teil 3.....	74
 Tag 23	 76
Der Dämmerungsschalter	76
Fototransistor	76
Aufgabe	76
Weihnachtliche Lichteffekte, wenn es dunkel wird	77
So funktioniert das Programm	77
 Tag 24	 78
Das Häuschen fertig stellen	78
Aufgabe	78
Das Programm	79
So funktioniert das Programm	80
Programmblöcke für Licht und Blinklichter	80
Programmblöcke für RGB-Blinklicht.....	80
Programmblöcke für das Weihnachtslied.....	81

3D-Druck-Adventskalender für Maker

Dieser Adventskalender bietet jeden Tag ein neues Projekt aus den Themenbereichen 3D-Druck, Elektronik oder Programmieren.

Programmieren

Keine Angst: Programmieren kann jeder. Dank grafischer Programmiersprachen (hier *mBlock*) braucht man sich auch keine Syntaxregeln mehr zu merken. Programme werden einfach am Bildschirm aus vorgefertigten Bausteinen zusammengesetzt. Damit lassen sich auch anspruchsvolle Programmieraufgaben spielerisch lösen.

Elektronik

Sämtliche Bauteile, die Sie zum Thema Elektronik brauchen, finden Sie im Adventskalender. Sie können alle Experimente ohne Vorkenntnisse in der Elektronik zusammenbauen. Die technischen Zusammenhänge werden im Handbuch erklärt.

Sie brauchen nicht zu löten.

Was Sie eventuell noch benötigen:

Da die Beschriftungen auf dem Nano-Board recht klein sind, wäre eine Lupe sehr hilfreich. Das Einstecken der Drähte auf dem Steckbrett wird durch eine stabile Pinzette oder eine Feinmechanikerzange erleichtert. Außerdem benötigen Sie vier AAA-Batterien. Für den

Anschluss des Nano-Boards brauchen Sie ein Micro-USB-Kabel (wird bei den meisten Smartphones mitgeliefert).

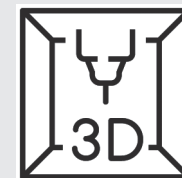
3D-Druck

Für die Projekte zum Thema 3D-Druck verwenden wir die Konstruktionssoftware 3D easy. Sie funktioniert direkt im Browser. Näheres dazu erfahren Sie an Tag 1 im Adventskalender.

Wir empfehlen als Browser MS Edge, Chrome oder Firefox. Auch wenn Sie keinen eigenen 3D-Drucker haben, können Sie die 3D-Objekte gestalten und sich ausdrucken und zuschicken lassen.

Für diejenigen, die einen eigenen 3D-Drucker besitzen, werden in den Konstruktionsbeschreibungen Tipps zu den Druckeinstellungen gegeben. Die Tipps beziehen sich auf die Slicer-Software *CURA*. Wenn Sie eine andere Software einsetzen, müssen Sie eventuell Anpassungen vornehmen oder die entsprechenden Einstellungen darin suchen.

Bitte beachten Sie, dass die Konstruktionsbeschreibungen und die Programmieranleitungen lediglich Vorschläge sind, die Sie natürlich Ihren Vorlieben anpassen können. Damit Sie gleich zu Anfang eine konkrete Vorstellung von dem Projekt haben, folgt hier eine kurze Darstellung des fertigen Objekts.



Vorsichtsmaßnahmen

Auf keinen Fall sollten Sie irgendwelche Pins des Nano miteinander verbinden und abwarten, was passiert. Nicht alle Pins lassen sich frei programmieren. Ein paar Pins sind für die Stromversorgung und andere Zwecke fest eingerichtet. Einige sind direkt mit Anschlüssen des Mikrocontrollers verbunden, ein Kurzschluss kann den Nano komplett zerstören – zumindest theoretisch. Die Arduino-kompatiblen Platinen sind erstaunlich stabil gegenüber Schaltungsfehlern. Verbindet man über eine LED zwei Pins miteinander, muss immer ein Vorwiderstand dazwischengeschaltet werden, wenn in der LED nicht bereits einer eingebaut ist.

Für Logiksignale benötigen einige Arduino-kompatible Platinen 3,3 V, andere 5 V. Das Arduino-kompatible Nano-Board in diesem Adventskalender verwendet ein +5-V-Signal als logisch *High* bzw. *wahr*.



Tag 1

DAS KONSTRUKTIONSPROGRAMM 3D EASY

● Im Kalender: Code für 3D easy

SO FUNKTIONIERT 3D EASY

● Konstruktion mit 3D easy

3D easy ist ein leicht zu bedienendes Konstruktionsprogramm für Einsteiger. Mit diesem Programm können Sie die Grundlagen des Konstruierens lernen und Ihr Leben mit selbst konstruierten Objekten bereichern.

Hochwertige und für den 3D-Druck optimierte Vorlagen bieten Ihnen die Möglichkeit, einfach und schnell zu guten Ergebnissen zu gelangen. Mit einer umfangreichen Auswahl an Gelenken ermöglicht 3D easy es

Ihnen auch, mehrfarbige und bewegliche Bauteile zu konstruieren.

Im Folgenden lernen Sie alle Funktionen kennen, die 3D easy zu bieten hat. Nutzen Sie dieses Programm nicht nur auf Ihrem PC oder Laptop, sondern auch auf Ihrem Tablet.

Um die Projekte aus dem FRANZIS-Adventskalender mit 3D easy PRO umzusetzen, können Sie auf zahlreiche Hilfen zurückgreifen.

PROJEKTE

Nach der Anmeldung in 3D easy können Sie im Hauptmenü auf den Bereich **Projekte** zugreifen. Dort finden Sie mehrere Beispielprojekte, mit denen Sie das Konstruieren einfacher Objekte erlernen können.

QUICK START GUIDE

Alternativ können Sie auch mit dem Quick Start Guide alle Funktionen von 3D easy Schritt für Schritt kennenlernen. Sie finden ihn als Download unter:

www.franzis.de/3d-druck-adventskalender

IHR ZUGANG ZU 3D EASY

Bitte geben Sie den Registriercode, den Sie hinter **Türchen Nummer eins** gefunden haben, hier ein:

<https://3deasy.3ddrucken24.de/Account/Register>

Damit erhalten Sie Zugang zu der Konstruktionssoftware.

Tag 2

DIE PROGRAMMIERSPRACHE MBLOCK

● Im Kalender: 1 Batteriefach

BATTERIEKASTEN

Heute befindet sich im Adventskalender für spätere Aufgaben ein Batteriekasten, der mit vier AAA-Batterien eine Spannung von 6 V liefert. Die Batterien sind nicht enthalten.

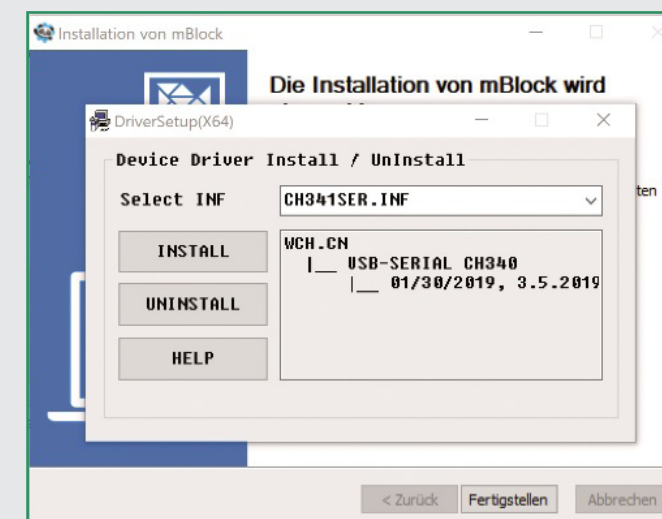
MBLOCK 5

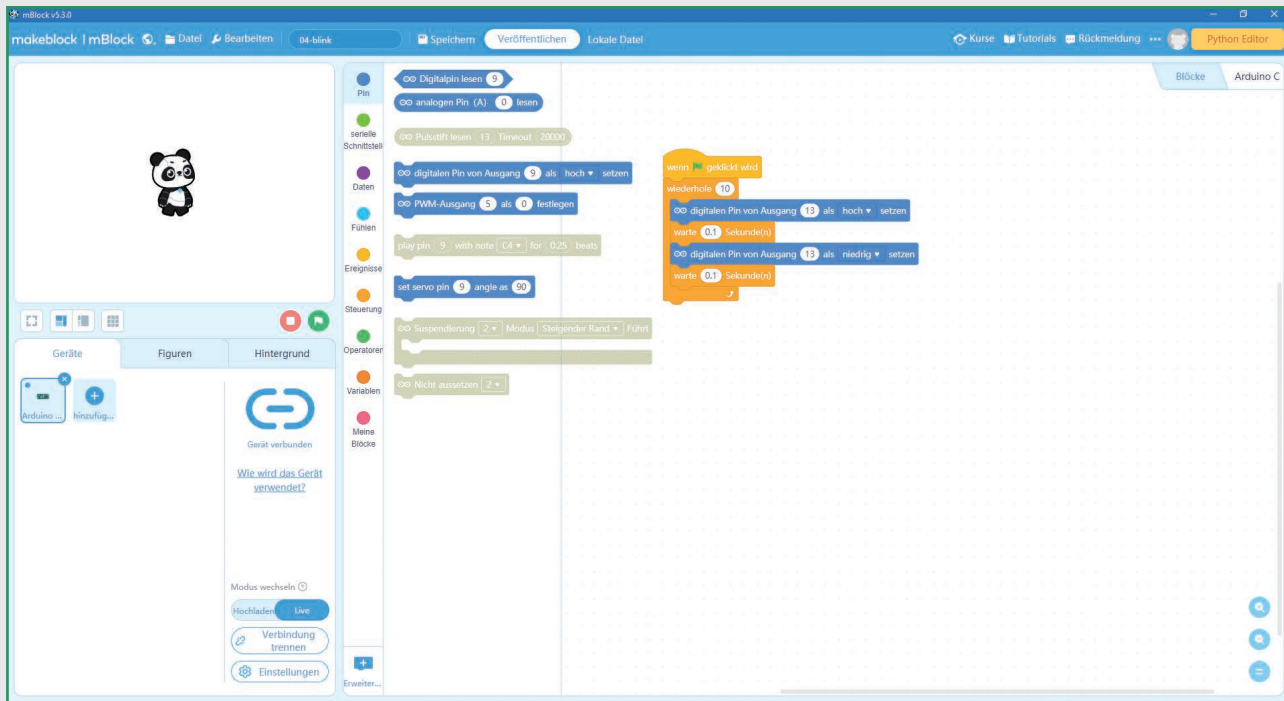
Für die Projekte im Adventskalender verwenden wir die einfach zu erlernende Programmiersprache *mBlock* 5. Die Software basiert auf dem populären Programm *Scratch*. Laden Sie sich die mBlock-5-PC-Version kostenlos bei www.mblock.cc/download herunter und installieren Sie sie. mBlock 5 funktioniert mit Windows 10, 8.1 und 7.

Ein Fenster informiert Sie über den Abschluss der Installation und bietet das Ausführen von mBlock an.

Starten Sie das Programm.

Jetzt erscheint die Programmieroberfläche mit dem M-Panda, der Symbolfigur von mBlock. Sollte mBlock 5 nicht automatisch auf Deutsch starten, fahren Sie auf der Menüleiste oben über das Weltkugelsymbol. Dort können Sie die Sprache einstellen. Klicken Sie links daneben, werden Sie auf eine Infoseite zu mBlock geleitet.





In mBlock 5 braucht man keinen Programmcode zu schreiben. Die Programme werden aus puzzlesteinartigen Blöcken zusammengesetzt. Diese befinden sich auf Blockpaletten, die farblich nach Themen geordnet sind:

Bereich	Geräte/Figuren	Farbe	Programmierbausteine (Beispiele)
Bewegung	Figuren	Blau	gehe, drehe, zeige in Richtung, gehe nach, setze X/Y ...
Pin	Geräte		Digitalpin lesen, analogen Pin lesen, digitalen Pin von Ausgang als setzen
Serielle Schnittstelle	Geräte	Hellgrün	an serielle Schnittstelle schreiben, lesbare Bytes von serieller Schnittstelle
Aussehen	Figuren	Lila	wechsle Objekt zu, gebe " " aus, wechsle Hintergrund, ändere Effekt, ändere Größe ...
Daten	Geräte		Karte von zu, Einschränkung zwischendurch, Konvertiert in
Klang	Figuren	Violett	spiele Sound, setze Lautstärke, setze Effekt ...
Ereignisse	Figuren/Geräte	Dunkelgelb	wenn Start, wenn Taste gedrückt, wenn ich angeklickt werde, sende " " an alle, wenn ich empfangen
Steuerung	Figuren/Geräte	Orange	warte, wiederhole, wenn " ", wenn " " sonst, warte bis, wiederhole bis, stoppe dieses Skript, stoppe alles
Fühlen	Figuren/Geräte	Hellblau	Maus X, Maus Y, Maustaste gedrückt, wurde " " berührt, wurde Farbe berührt, Stoppuhr, zu, frage " " und warte, Lautstärke, Tage seit 2000 ...
Operatoren	Figuren/Geräte	Hellgrün	+, -, *, /, Zufallszahl von bis, <, =, >, und, oder, nicht, mod, gerundet, Funktion von ...
Variablen/Listen	Figuren/Geräte	Orange/Rot	setze auf, ändere um, zeige, verstecke, füge zu hinzu ...
Meine Blöcke	Figuren/Geräte	Pink	hier können eigene Blöcke erstellt werden

Aufgabe

In einem einfachen Programm 02-mpanda.mblock soll die Figur auf der Bühne (Fenster links mit dem Panda) hin- und herlaufen und dabei ihre Farbe verändern. Dafür brauchen wir noch keine Elektronik.

Achten Sie darauf, dass in dem Fenster unter der Bühne das Register **Figuren** ausgewählt ist, da sonst nicht alle Funktionen in der Blockpalette aktiv sind.

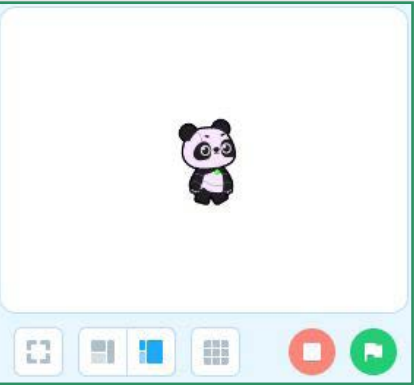


SO EINFACH FUNKTIONIERT DAS PROGRAMM

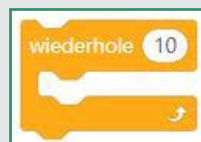
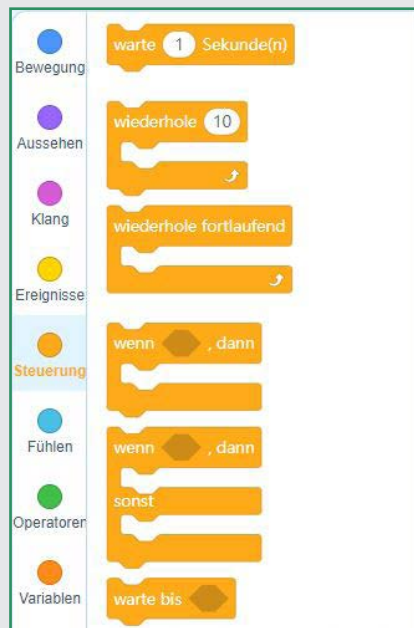
- Klicken Sie in der Blockpalette in der Mitte auf das dunkelgelbe Symbol **Ereignisse**. Damit werden die Blöcke angezeigt, die auf bestimmte Ereignisse reagieren.
- Unter der Bühne ist ein weißes Fähnchen auf grünem Grund zu sehen. Es dient üblicherweise dazu, ein Programm zu starten.



- Ziehen Sie den abgebildeten Block aus der Blockpalette in das graue Skriptfenster rechts.



- Der Block **wenn Fahne geklickt wird** sowie das Fähnchensymbol unter der Bühne bewirken, dass die folgenden Blöcke ausgeführt werden, wenn der Benutzer auf das Fähnchen klickt. Der Block ist oben rund, passt also unter keinen anderen Block. Er muss immer als Erstes gesetzt werden.
- Klicken Sie in der Blockpalette auf das gelbe Symbol **Steuerung**.



6. Die Bewegung der Figur wird aus einzelnen Schritten zusammengesetzt. Die Figur geht 16 Schritte in eine Richtung, dreht sich, geht wieder 16 Schritte zurück und so weiter. Ziehen Sie für die Wiederholungsschleife den abgebildeten Block in das Skriptfenster und docken Sie ihn unten an den dort bereits vorhandenen Block an. Wenn Sie zwei Blöcke nahe genug aneinanderziehen, verbinden sie sich automatisch.
7. Schreiben Sie in das Zahlenfeld im Block die Zahl 16.
8. Bis jetzt haben wir zwar die Wiederholung programmiert, aber noch keine wirkliche Bewegung. Um eine Bewegung zu programmieren, klicken Sie in der Blockpalette auf das blaue Symbol Bewegung.



9. Ziehen Sie den Block **gehe 10 Schritte** in das Skriptfenster und docken Sie ihn innerhalb der Schleife an.
10. Nachdem die Figur 16-mal einen 10er-Schritt gegangen ist, soll sie sich um 180 Grad rechts herum drehen und wieder zurückgehen. Ziehen Sie dazu einen Block **drehe dich um ... Grad** unter die Wiederholung. Ändern Sie den Zahlenwert auf 180.
11. Bis jetzt geht die Figur 16-mal einen 10er-Schritt, dreht sich und bleibt stehen. Sie soll aber immer wieder hin- und zurückgehen. Trennen Sie dazu zunächst die Anweisung **wenn Fahne geklickt wird** vom darunterliegenden Block, indem Sie auf **wiederhole 16** gehen und dies mit der Hand nach unten ziehen. Ziehen Sie dann aus der Blockpalette Steuerung den Block **wiederhole fortlaufend** unter den Block **wenn Fahne geklickt wird**. Anschließend ziehen Sie die **wiederhole 16**-Schleife in die Klammer der **wiederhole fortlaufend**-Schleife.



12. Probieren Sie aus, ob das Programm auch wie erwartet funktioniert. Klicken Sie dazu rechts oben in der Bühne auf das grüne Fähnchen. Die Figur geht hin und zurück, allerdings steht sie auf dem Rückweg auf dem Kopf, und die gewünschte Farbveränderung zeigt sich auch noch nicht. Halten Sie das Programm mit einem Klick auf das rote Stoppsymbol neben dem grünen Fähnchen wieder an.

13. Ziehen Sie aus der Blockpalette Bewegung den Block **setze den Rotationsstil auf links-rechts** über den Block **drehe dich um 180 Grad**. Anstatt eine kreisförmige Drehung zu vollführen, dreht sich die Figur jetzt nur noch nach links oder rechts.



14. Ziehen Sie aus der Blockpalette Aussehen den Block **ändere Effekt Farbe um 25** in die **wiederhole 16**-Schleife unter den Block **gehe 10 Schritte**, damit sich die Farbe bei jedem Schritt etwas verändert. Schreiben Sie in das Zahlenfeld des Blocks **ändere Effekt Farbe um 25** die Zahl „5“.

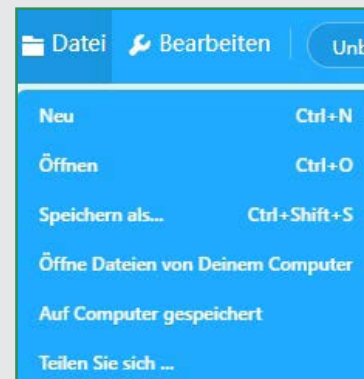


TIPP: Probieren Sie aus, was passiert, wenn Sie statt der Zahl 5 eine höhere oder niedrigere Zahl einsetzen.

HINWEIS: Sollte der Panda mal von der Bühne laufen, stoppen Sie das Programm und ziehen ihn wieder auf die richtige Position.

15. Starten Sie jetzt das Programm erneut mit einem Klick auf das grüne Fähnchen. Jetzt sollte alles wie erwartet laufen.

Sie können die Programme auch einfach jeden Tag anhand der Abbildungen selbst zusammenbauen.



Die Programme zum Adventskalender

Alle verwendeten Programme finden Sie in www.franzis.de/3d-druck-adventskalender. Entpacken Sie das Zip-Archiv in ein persönliches Verzeichnis auf der Festplatte. Die Programme sind nach Tagen nummeriert. Das Programm von heute finden Sie dort unter dem Namen 02-mpanda.mblock. Wählen Sie im Menü **Datei/Öffne Dateien von Deinem Computer**, um ein Programm zu öffnen. Auch wenn Sie mBlock gerade erst geöffnet und keine Aktion gestartet haben, erscheint eine Aufforderung zum Speichern, was mit der Möglichkeit zusammenhängt, Programme direkt auf dem mBlock-Server zu öffnen. Klicken Sie in dem Fall auf **Nicht Speichern** und laden Sie danach Ihr Programm.

KONSTRUIEREN MIT 3D EASY

Die Anleitungen für die Konstruktion Ihrer Projekte finden Sie gesammelt unter <https://www.franzis.de/3d-druck-adventskalender>. Wählen Sie einfach den zum jeweiligen Tag passenden Konstruktionsschritt aus.

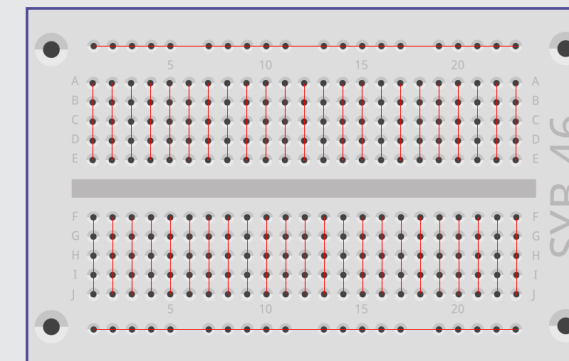
Tag 3

DIE BLINK-LED

● Im Kalender: 1 Steckbrett, 1 rote Blink-LED

Steckbrett

Für den schnellen Aufbau elektronischer Schaltungen ohne Löten befindet sich heute ein Steckbrett im Adventskalender. Hier können elektronische Bauteile direkt in ein Lochraster gesteckt werden. Bei diesem Steckbrett sind die äußeren Längsreihen mit Kontakten (X und Y) alle miteinander verbunden. Diese Kontaktreihen werden oft als Plus- und Minuspol zur Stromversorgung der Schaltungen genutzt. In den anderen Kontaktreihen sind jeweils fünf Kontakte (A bis E und F bis J) quer miteinander verbunden, wobei in der Mitte der Platine eine Lücke gelassen wurde. Dort können größere Bauelemente eingesteckt und nach außen hin verdrahtet werden.



Die Verbindungen auf dem Steckbrett.

LEDs

LEDs (zu Deutsch Leuchtdioden) leuchten, wenn Strom in Durchflussrichtung durch sie fließt. LEDs werden in Schaltungen mit einem pfeilförmigen Dreieckssymbol dargestellt, das die Flussrichtung vom Pluspol zum Minuspol oder zur Masseleitung anzeigt. Eine LED lässt in Durchflussrichtung nahezu beliebig viel Strom durch, sie hat nur einen sehr geringen Widerstand. Um den Durchflussstrom zu begrenzen und damit ein Durchbrennen der LED zu verhindern, muss üblicherweise zwischen dem verwendeten Anschlusspin und der Anode der LED – oder zwischen Kathode und Massepin – ein 220-Ohm-Vorwiderstand eingebaut werden. Dieser Vorwiderstand schützt auch den Ausgang des Nano vor zu hohen Stromstärken. Die LEDs im Adventskalender haben den Vorwiderstand bereits eingebaut und können daher direkt an die Pins angeschlossen werden.

Der Widerstand kann auf Basis des Ohm'schen Gesetzes

$$R = \frac{U}{I}$$

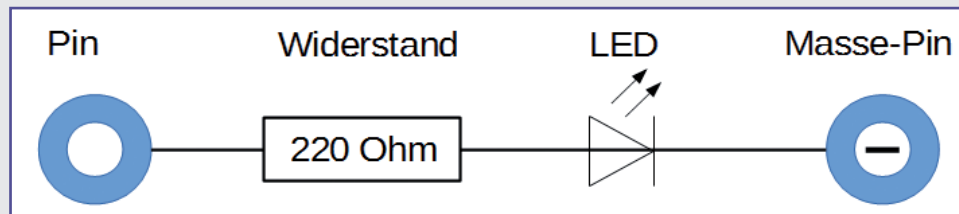
leicht berechnet werden.

Für unser Beispiel: Der Vorwiderstand hat einen Wert von 220 Ohm. Die Spannung der vier in Reihe geschalteten AAA-Batterien beträgt maximal 6 V (Volt). Nach dem Ohm'schen Gesetz fließt dann eine Stromstärke

LED in welcher Richtung anschließen?

Die beiden Anschlussdrähte einer LED sind unterschiedlich lang. Der längere ist der Pluspol, die Anode, der kürzere der Minuspol, die Kathode. Einfach zu merken: Das Pluszeichen hat einen Strich mehr als das Minuszeichen und macht damit den Draht quasi optisch etwas länger. Außerdem sind die meisten LEDs auf der Minusseite abgeflacht (haben also eine Kante), vergleichbar mit einem Minuszeichen. Leicht zu merken: Kathode = kurz = Kante.

von 0,0273 A (Ampere) bzw. 27,3 mA (Milliampere)
durch die LED.



Schaltplan einer LED mit Vorwiderstand.

Blink-LEDs

Die Blink-LED hinter dem heutigen Türchen des Adventskalenders blinkt, ohne dass ein Programm läuft. Sie muss einfach nur mit Strom versorgt werden.

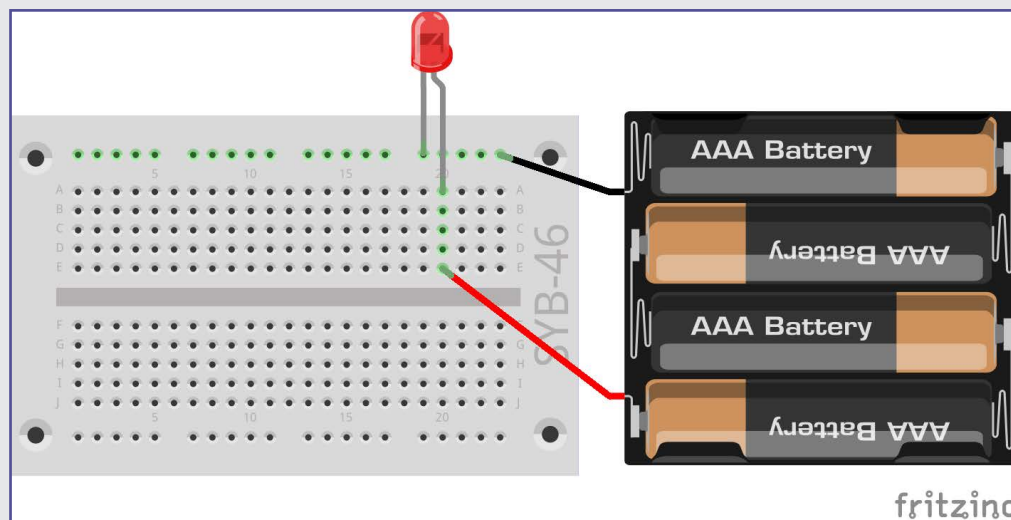
Bauteile

1 Steckbrett (SYB 46)

1 Blink-LED rot mit Vorwiderstand

1 Batteriekasten

TIPP: Markieren Sie sich die LED, da sie sonst mit anderen roten LEDs verwechselt werden kann und diese Schaltungen dann nicht wie gewünscht funktionieren.



KONSTRUIEREN MIT 3D EASY – DIE GRUNDPLATTE FÜR DAS WEIHNACHTSHAUS

Heute konstruieren Sie die Grundplatte für Ihr Weihnachtshaus. Lesen Sie dazu die Anleitung zu Tag 3, die Sie bereits heruntergeladen haben. Falls nicht, zur Erinnerung hier der Link:

<https://www.franzis.de/3d-druck-adventskalender>

Tag 4

DAS NANO-BOARD

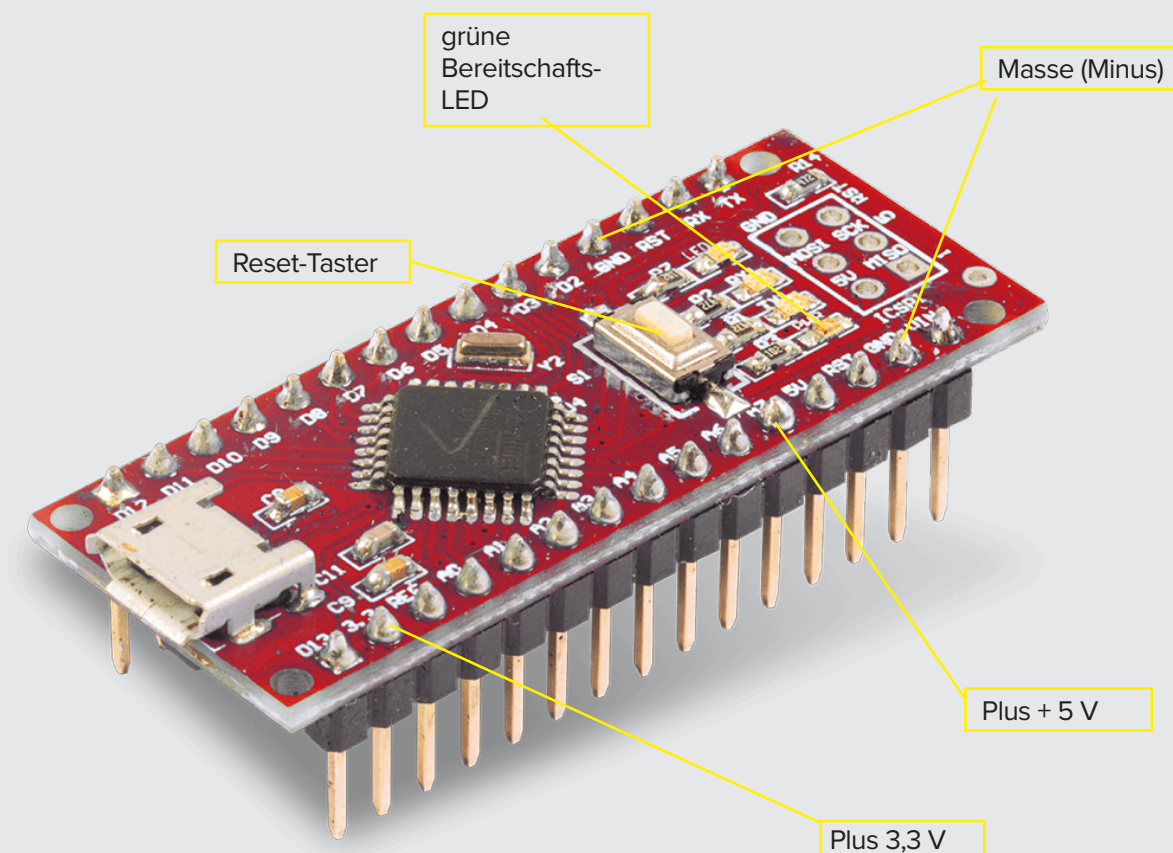
Im Kalender: Nano-Board

Das Programmieren von Mikrocontrollern war früher nur etwas für Ingenieure und Informatiker. Heute ermöglicht die Arduino-Plattform dank übersichtlicher Hardware und einfach zu verstehender Software jedem den Einstieg in die Mikrocontrollertechnik.

DAS NANO-BOARD – EINE ARDUINO-KOMPATIBLE PLATINE

Die Arduino-Plattform bietet mittlerweile eine größere Vielfalt an Platinen für unterschiedliche Anwendungszwecke. Dieser Adventskalender enthält eine mit dem Arduino-Nano-Standard kompatible Platine. Sie kann später direkt auf ein Steckbrett gesteckt werden, um weitere Elektronik anzuschließen. Die mit D2 bis D12 beschrifteten Pins sind digitale Ein- oder Ausgänge, die

die Logikwerte 0 oder 1 – oder, anders geschrieben, *Low* oder *High* (in mBlock **hoch** bzw. **niedrig**) – annehmen können. Damit lassen sich zum Beispiel LEDs ein- und ausschalten. Die mit A0 bis A7 beschrifteten Pins sind analoge Eingänge. Hier kann eine Spannung zwischen 0 und 5 V angelegt werden. Der Nano wertet sie aus und kann mit bestimmten Aktionen antworten



DEN NANO VORBEREITEN

Um den Nano in Betrieb zu nehmen, braucht man:

- PC mit Windows 10, 8.1, 7
- Micro-USB-Kabel
- Treiber

Die Verbindung zwischen PC und Nano erfolgt über ein Micro-USB-Kabel. Sie brauchen sich nicht extra ein sol-

ches Kabel zu besorgen, die meisten (älteren) Smartphones verwenden diesen Steckertyp. Das Kabel wird zugleich zur Stromversorgung und zur Datenübertragung verwendet.

Schließen Sie das Kabel nach Möglichkeit an einen USB-2.0-Anschluss Ihres PCs an, da es an USB-3.0-Anschlüssen eher zu Verbindungsproblemen kommen kann. Zur besseren Unterscheidung sind USB-3.0-Anschlüsse innen meistens blau.

SOFTWAREINSTALLATION IN KÜRZE

Starten Sie dazu mBlock 5 und verbinden Sie den Nano mit einer USB-Schnittstelle:

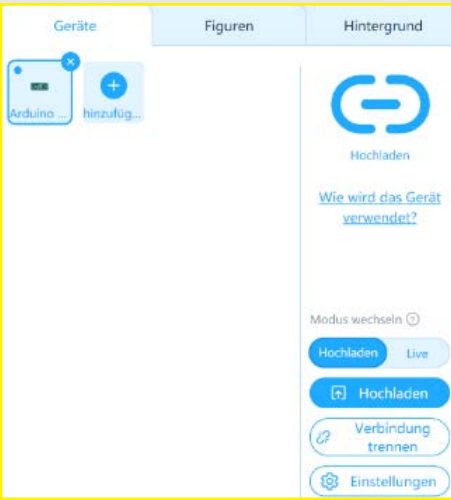
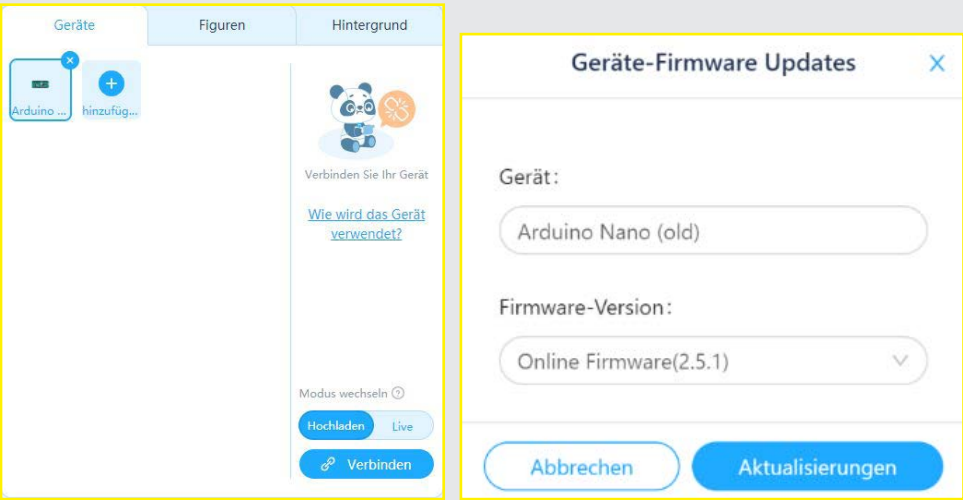
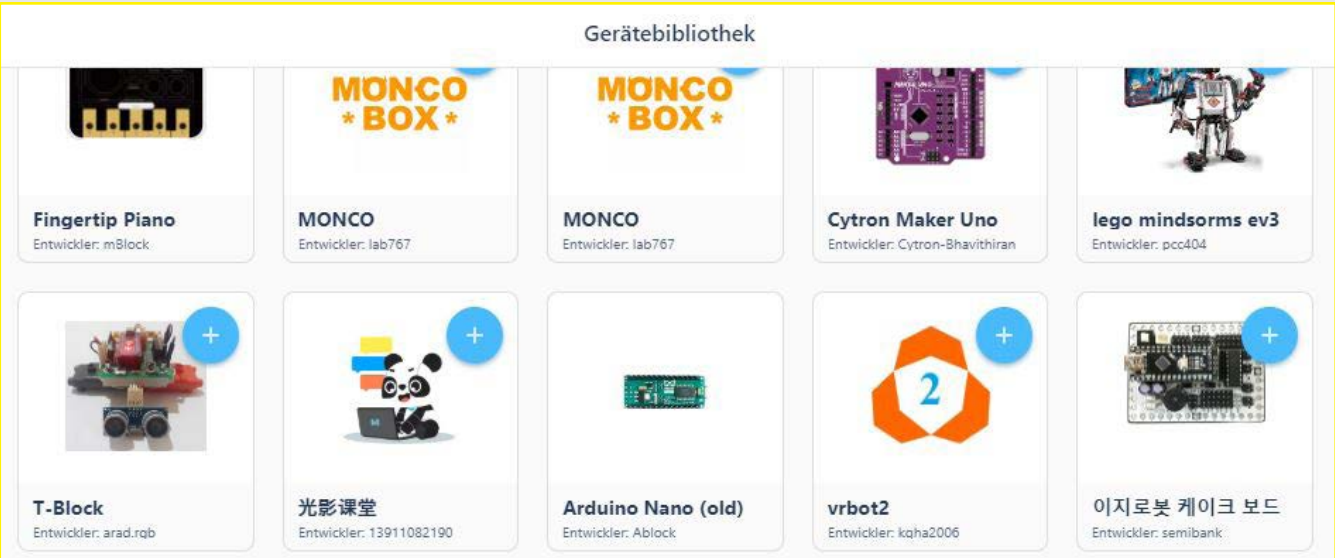
1. Unter **Geräte** im linken unteren Fenster ist ein Gerät **CyberPi** vorinstalliert. Entfernen Sie es mit einem Klick auf das x-Symbol.

2. Klicken Sie auf **Hinzufügen**. In der nun erscheinenden Gerätebibliothek scrollen Sie weiter nach unten, bis Sie in der Mitte **Arduino Nano (old)** finden. Ignorieren Sie das weiter oben befindliche Bild für den **Arduino Nano**.
3. Klicken Sie auf das Pluszeichen, und die Software dafür wird heruntergeladen. Erst danach können Sie unten auf **OK** klicken.

4. Der Arduino Nano erscheint nun im **Geräte**-Fenster.

5. Klicken Sie auf **Verbinden**.

6. Wichtig: Setzen Sie einen Haken bei **Alle anschlussfähigen Geräte anzeigen**. Eine Schnittstelle wird vorgeschlagen. Solange nur ein COM-Port angezeigt wird, wählen Sie diesen aus. Werden mehrere Ports angezeigt, ist in den meisten Fällen der mit der höheren Nummer der richtige. Klicken Sie dann auf **Verbindung**.



7. Der Nano wird nun als **verbunden** angezeigt.



8. Damit ein Programm aus mBlock den Nano steuern kann, muss noch eine spezielle Firmware installiert werden. Dazu erscheint ein orangefarbener Button **Aktualisieren**. Klicken Sie darauf.
9. Kontrollieren Sie noch einmal, ob für das Gerät auch wirklich **Arduino Nano (old)** angezeigt wird, und wählen Sie dann **Aktualisierungen**. Dabei wird eine spezielle Software zur Echtzeitkommunikation auf dem Nano installiert, was einige Sekunden dauert.

Währenddessen blinkt die eine LED auf dem Nano schnell.

Sollte die Aktualisierung mit einer Fehlermeldung abbrechen, gehen Sie in der Menüleiste von mBlock oben rechts auf die drei Punkte. In dem Pull-down-Menü wählen Sie **Schnellinstallation des Serial-Treibers**. Deinstallieren Sie ihn erst und installieren Sie ihn anschließend wieder. Dann sollte es funktionieren.

10. Nun müssen Sie den Nano erneut verbinden. Jetzt kann das Nano-Board im interaktiven Modus von mBlock verwendet werden, um elektronische Bauteile vom PC aus zu steuern

Fehlerbehebung

Die Verbindung mit dem Nano könnte von der Windows-Defender-Firewall blockiert werden. Hier benötigt mBlock 5 eine Berechtigung, die Sie mit einem Klick auf **Zugriff zulassen** in dem Dialogfeld, das automatisch erscheint, erteilen müssen.

Falls Sie eine andere Firewall auf dem PC installiert haben, schalten Sie diese aus. Viele externe Firewalls blockieren Verbindungen über USB-Kabel und bieten keine Möglichkeit, diese freizugeben, ohne die Firewall komplett abzuschalten.

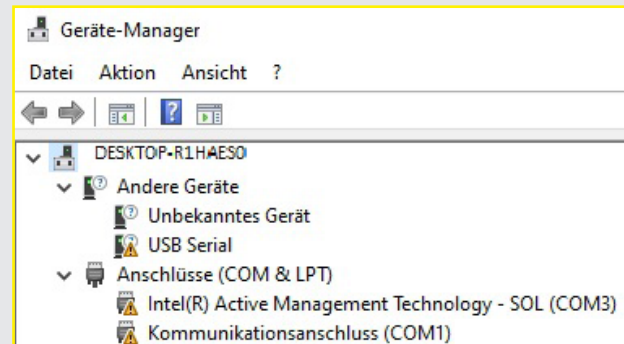
AUFGABE

Die mit „LED“ bezeichnete LED auf dem Nano soll zehnmal blinken. Starten Sie dazu mBlock 5.

SO FUNKTIONIERT DAS PROGRAMM

Achten Sie darauf, dass der Modus unten links im Verbindungsfenster auf **Live** gesetzt ist, um die Programme sofort ausführen zu können.

Das Programm 04-blink.mblock startet wie die meisten Programme in mBlock 5 mit dem Block **wenn Fahne geklickt wird**. Danach folgt die bereits bekannte Schleife **wiederhole 10**. Sie sorgt dafür, dass die darin enthaltenen Blöcke zehnmal hintereinander ausgeführt werden.



Sollte der Nano nicht gefunden werden, kontrollieren Sie bitte in der **Systemsteuerung** unter **Geräte-Manager**, ob und an welcher Schnittstelle er registriert ist.

Ist er nicht aufgelistet und führte die Installation des Treibers wie unter Punkt 7 beschrieben auch nicht zum Erfolg, entpacken Sie das Zip-Archiv des Nano-Treibers aus dem Download in einen eigenen Ordner. Starten Sie dann die Installation der CH341SER.EXE. Klicken Sie im Installationsdialog auf **Install** und warten Sie, bis eine Bestätigung dazu erscheint, dass der Treiber installiert wurde. Starten Sie anschließend mBlock wieder und beginnen Sie erneut mit Schritt 1.

Weitere Informationen und Dokumentationen zum Nano-Board finden Sie hier:

<https://www.franzis.de/maker/raspberry-pi-arduino-und-mehr/nano-board-platine>



Der Block **digitalen Pin von Ausgang 13 als hoch setzen** von der Blockpalette **Pin** setzt einen der digitalen Pins des Nano-Boards auf einen Logikwert **hoch** (an) oder **niedrig** (aus). Die Werte können mit einem Klick auf den kleinen Pfeil rechts im Feld ausgewählt werden.

Nachdem die LED an Pin 13 eingeschaltet ist, wird durch den Block **warte ... Sekunden** 0,1 Sekunden gewartet. So lange leuchtet die LED. Danach wird die LED an Pin 13 wieder ausgeschaltet. Jetzt wartet das Programm erneut 0,1 Sekunden. Danach beginnt das Ganze von vorn.

Die LED an Pin 13

Für Statusanzeigen ohne Zusatzhardware hat der Nano eine eigene eingebaute LED, die an Pin 13 gesteuert wird.

Testen Sie das Programm, indem Sie auf das grüne Fähnchen klicken.

Das Programm ist als 04-blink.mblock in der Zip-Datei enthalten, die Sie an Tag 2 heruntergeladen haben. Wenn Sie das Programm selbst erstellt haben, speichern Sie es im Menü **Datei/Auf Computer gespeichert** (siehe Tag 2), nicht unter **Speichern als...** Das würde Sie auf einen Cloudspeicher leiten.

Dezimalpunkt statt Komma

mBlock verwendet, wie viele chinesische oder amerikanische Programme, den Punkt als Dezimaltrennzeichen, nicht das in Deutschland übliche Komma. Bedenken Sie das bei der Eingabe von Zahlenwerten.

Nano ausschalten

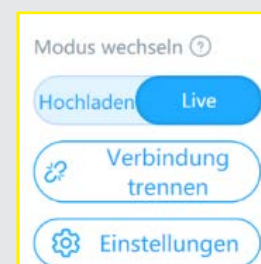
Wollen Sie den Nano ausschalten, trennen Sie einfach die Stromversorgung, und er schaltet sich ab. Beim nächsten Einschalten startet automatisch das zuletzt gespeicherte Programm. Das Gleiche passiert, wenn Sie den Taster auf dem Nano drücken.

Der Name Arduino

Der Arduino kommt aus Italien, sein Name stammt vom italienischen König Arduino, der bis ins Jahr 1005 in Ivrea herrschte, dem Firmensitz des Herstellers. Nach König Arduino ist auch die Lieblingsbar der Arduino-Entwickler Massimo Banzi und David Cuartielles benannt.

KONSTRUIEREN MIT 3D EASY – DIE GRUNDPLATTE FÜR DAS WEIHNACHTSHAUS, TEIL 2

Heute stellen Sie die Grundplatte fertig. Lesen Sie dazu die Anleitung zu Tag 4.



Tag 5

ZWEI LEDS BLINKEN ABWECHSELND

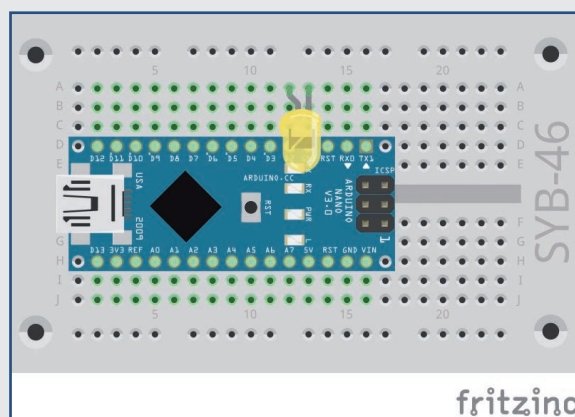
● Im Kalender: LED gelb mit Vorwiderstand

AUFGABE

Zwei LEDs sollen abwechselnd mit einstellbarer Geschwindigkeit blinken, die eingebaute LED und eine extern angeschlossene.

Bauteile

- 1 Steckbrett
- 1 Nano-Board
- 1 LED gelb mit Vorwiderstand



Stecken Sie das Nano-Board wie in der Abbildung auf das Steckbrett. Auf einem neuen Steckbrett lässt sich das Nano-Board oft nur schwer einstecken. Wenden Sie etwas gleichmäßigen Druck an, aber benutzen Sie kein Werkzeug wie etwa einen Hammer oder Ähnliches, das könnte die Elektronik beschädigen.

Achten Sie beim Aufbau der Schaltung darauf, dass die Kathode (kurzer Draht) der LED mit dem GND-Pin verbunden ist und die Anode (langer Draht) mit dem D2-Pin.

Die Pins auf dem Nano

Alle Pins, die mit D beginnen, sind digitale Ein- oder Ausgänge, die die Werte *wahr* oder *falsch* (ein oder aus) annehmen können. Die Pins beginnend mit A sind analoge Eingänge. GND-Pins sind Masseleitungen (also Minus). Arduino-kompatible Platinen arbeiten mit unterschiedlichen Spannungen und haben dazu standardmäßig zwei verschiedene Pluspins. An Pin 3.3 liegen +3,3 V Spannung an. An Pin 5 V liegen +5 V Spannung an. Der Nano im Adventskalender benötigt für ein logisches *wahr*-Signal +5 V, manche anderen Platinen nur +3,3 V.

ZWEI LEDS BLINKEN ABWECHSELND

Das Programm 05-1ed2blink.mblock ähnelt dem Programm des 4. Tags bis auf die zusätzlichen Blöcke für die zweite LED. Sie sind gegenläufig. Wird die eine LED eingeschaltet, wird die andere ausgeschaltet, damit beide LEDs abwechselnd blinken.



SO FUNKTIONIERT 3D EASY

Eine **wiederhole fortlaufend**-Schleife sorgt dafür, dass die beiden LEDs abwechselnd endlos blinken. Nachdem die LED an Pin 2 eingeschaltet wurde, wird die eingebaute LED an Pin 13 ausgeschaltet. Jetzt wartet das Programm die eingestellte Zeit. Danach wird auf die gleiche Weise die LED an Pin 13 ein- und die an Pin 2 ausgeschaltet. Nach einer weiteren Wartezeit beginnt der Zyklus von vorn.

KONSTRUIEREN MIT 3D EASY – DAS GRUNDGERÜST FÜR DAS WEIHNACHTSHAUS

Heute beginnen Sie mit der Konstruktion des Weihnachtshauses. Lesen Sie dazu die Anleitung für das Grundgerüst an Tag 5.

EINE LED BLINKT MIT WECHSELNDER GESCHWINDIGKEIT

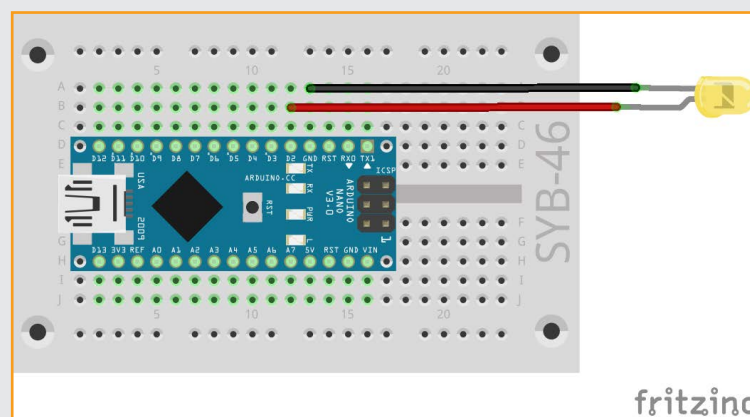
● Im Kalender: Verbindungskabelpaar

Verbindungskabel

Heute befindet sich ein Verbindungskabelpaar im Adventskalender. Diese Leitungen werden verwendet, um LEDs, die auf einem 3D-gedruckten Objekt stecken, mit dem Steckbrett zu verbinden. Die Verbindungskabel haben an einem Ende kleine Drahtstecker, mit denen sie sich leicht in das Steckbrett stecken lassen. Am anderen Ende befinden sich Klemmen, in die die Drähte einer LED gesteckt werden können.

Bauteile
1 Nano-Board
1 Steckbrett
1 LED gelb mit Vorwiderstand
1 Verbindungskabelpaar

Die LED ist über die Verbindungskabel mit dem Steckbrett verbunden. Achten Sie beim Aufbau der Schaltung darauf, dass die Kathode (kurzer Draht) der LED mit dem GND-Pin des Nano-Boards verbunden ist, die Anode (langer Draht) ist in dieser Schaltung mit Pin D2 auf dem Nano verbunden.



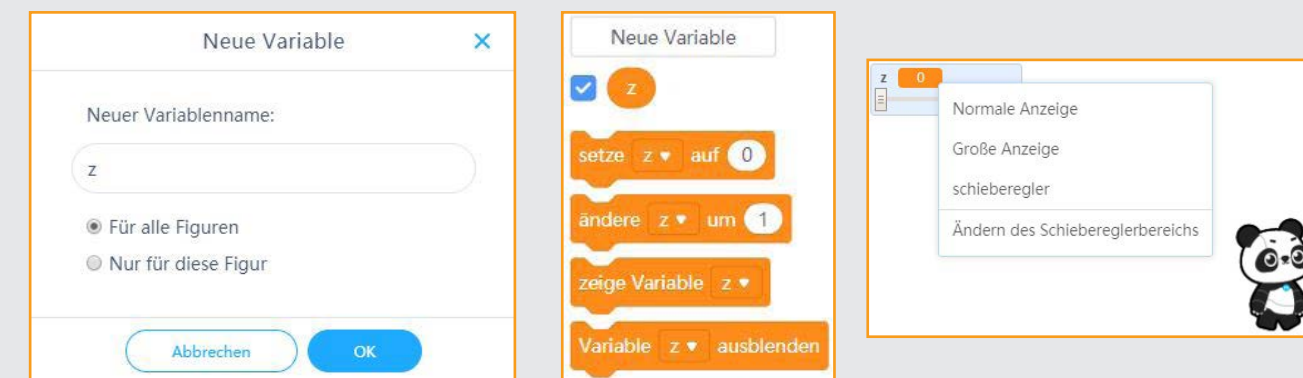
AUFGABE

Eine LED soll mit einstellbarer Geschwindigkeit blinken.

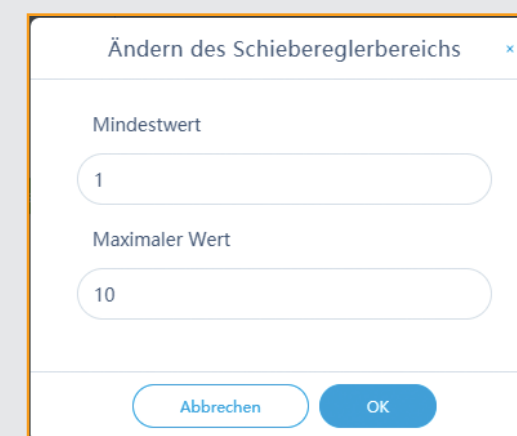
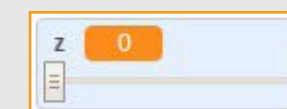
LED BLINKT MIT EINSTELLBARER GESCHWINDIGKEIT

Das Programm 06-ledblink.mblock des 6. Tags lässt wieder eine LED blinken. Allerdings kann die Geschwindigkeit mit einer Variablen im Programm eingestellt werden.

1. Legen Sie zuerst auf der Blockpalette **Variablen** eine neue Variable **z** an, die die Wartezeit beim Blinken festlegt.



2. Solange das blaue Häkchen links neben dem Variablensymbol auf der Blockpalette aktiviert ist, erscheint die Variable auf der Bühne und zeigt dort immer den aktuellen Wert. Klicken Sie mit der rechten Maustaste auf die Variable auf der Bühne und wählen Sie im Kontextmenü **schieberegler**. Damit erscheint ein Schieberegler, mit dem der Variablenwert, während das Programm läuft, eingestellt werden kann.
3. Klicken Sie noch einmal mit der rechten Maustaste auf die Variable auf der Bühne und wählen Sie diesmal im Kontextmenü **Ändern des Schiebereglerbereichs**. Tragen Sie hier die Werte „1“ und „10“ ein und bestätigen Sie mit **OK**. mBlock 5 rechnet zwar mit Fließkommazahlen, die Schieberegler funktionieren aber nur mit ganzzahligen Werten. Deshalb wird der eingestellte Wert anschließend mit dem Divisionsblock **.../...** aus der Blockpalette **Operatoren** durch 10 geteilt. Je nach eingestelltem Wert soll die LED zwischen 0,1 Sekunden und einer Sekunde leuchten und danach genauso lange ausgeschaltet bleiben.



4. Bauen Sie jetzt das Programm wie in der Abbildung gezeigt zusammen.



5. Beachten Sie: In dem Block **warte ... Sekunden** können Zahlen nur mit der Tastatur eingetragen werden. Um die Variable **z** einzutragen, ziehen Sie das Symbol der Variablen **z** aus der Blockpalette **Variablen** einfach in das Zahlenfeld.

HINWEIS: Sollte ein Programm trotz richtiger Programmierung und angezeigter Verbindung nicht laufen und stattdessen der Button **Aktualisieren** orange unterlegt sein, kontrollieren Sie bitte in dem Fenster zur Aktualisierung der Firmware, ob dort wirklich **Arduino Nano (old)** angezeigt wird. Mit dem anderen Nano-Board können Sie keine Firmware installieren und auch kein Programm starten. Leider sehen die Symbole in dem **Geräte**-Fenster für beide Boards gleich aus.

SO FUNKTIONIERT DAS PROGRAMM

Nach Klick auf das grüne Fähnchen wird die Variable **z** zunächst auf einen mittleren Wert von 5 gesetzt, der dann im Programm auf 0,5 Sekunden umgerechnet wird. Die **wiederhole fortlaufend**-Schleife lässt die LED endlos blinken. Statt einer vom Programm fest vorgegebenen Zeit zwischen dem Umschalten wird die Variable **z** verwendet. Diese können Sie, während das Programm läuft, jederzeit mit dem Schieberegler verändern. Die LED blinkt entsprechend schneller oder langsamer.

KONSTRUIEREN MIT 3D EASY – DAS DACH DES WEIHNACHTSHAUSES

Heute machen Sie mit dem Dach des Weihnachtshauses weiter. Lesen Sie dazu die Anleitung zu Tag 6.

Tag 7

EINSTELLBARE WECHSELBLINK-LEDS

- Im Kalender: Verbindungskabelpaar, LED grün

AUFGABE

Zwei LEDs auf dem Hausdach sollen abwechselnd mit einstellbarer Geschwindigkeit blinken.

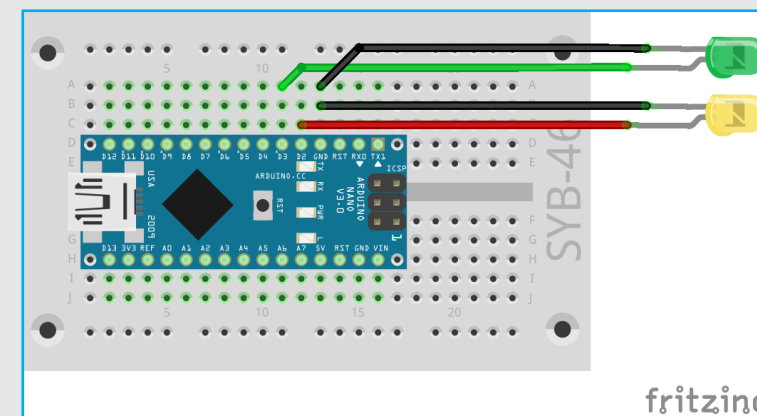
Das Programm 07-led2varioblink.mblock ähnelt dem Programm des 5. Tags bis auf den Einsatz der Variablen **z**.

Achten Sie beim Aufbau der Schaltung darauf, dass die Kathoden (kurzer Draht) der LEDs mit dem GND-Pin

verbunden sind.

Mit den Verbindungskabelpaaren werden später die LEDs in bzw. auf dem Häuschen und dem Tannenbaum installiert.

Setzen Sie das Programm wie abgebildet zusammen. Installieren Sie den Schieberegler wie am vorherigen Tag beschrieben.



Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 1 LED gelb mit Vorwiderstand
- 1 LED grün mit Vorwiderstand
- 2 Verbindungskabelpaare



SO FUNKTIONIERT DAS PROGRAMM

Der Ablauf ist grundlegend der gleiche wie im Programm des 5. Tags. Neu ist, dass Sie die Blinkzeit jetzt individuell einstellen können.

KONSTRUIEREN MIT 3D EASY – DER SCHORNSTEIN DES WEIHNACHTSHAUSES

Weiter geht es mit dem Schornstein. Lesen Sie dazu die Anleitung zu Tag 7.

Tag 8

DIMMEN VON LEDS

● Im Kalender: Schaltdraht

SCHALTDRAHT

Heute ist Schaltdraht im Adventskalender enthalten. Damit können Sie kurze Verbindungsbrücken herstellen, mit denen dicht nebeneinanderliegende Kontaktreihen auf der Steckplatine verbunden werden. Solche Drahtbrücken liegen flach auf dem Steckbrett und stören beim Aufbau der anderen Bauteile weniger als lange Verbindungskabel.

Schneiden Sie den Draht mit einem kleinen Seitenschneider je nach Experiment auf die passenden Längen ab. Um die Drähte besser in die Steckplatine stecken zu können, empfiehlt es sich, sie leicht schräg abzuschneiden, sodass eine Art Keil entsteht. Entfernen Sie an beiden Enden mit einem scharfen Messer auf einer Länge von etwa einem halben Zentimeter die Isolierung.

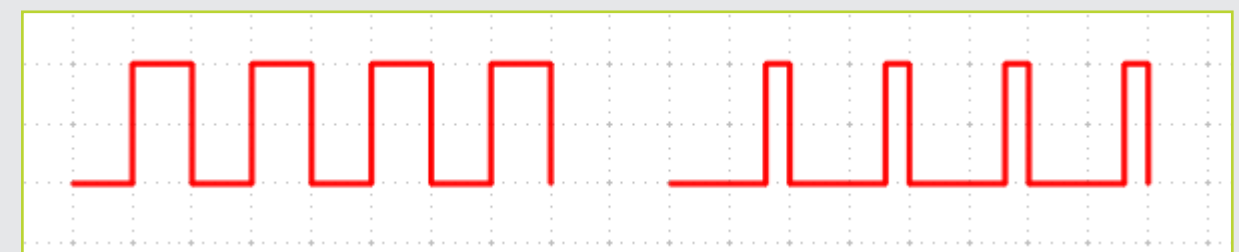
Bauteile

1 Nano-Board
1 Steckbrett
2 LEDs verschiedener Farbe mit Vorwiderständen
2 Verbindungskabelpaare
1 Drahtbrücke

LEDS DIMMEN

LEDs sind typische Bauteile zur Ausgabe von Signalen in der Digitalelektronik. Sie können zwei verschiedene Zustände annehmen, ein und aus, 0 und 1 oder *Low* (**tief**) und *High* (**hoch**). Das Gleiche gilt für die als Ausgänge definierten digitalen Pins. Demnach wäre es theoretisch nicht möglich, eine LED zu dimmen. Mit einem Trick gelingt es dennoch, die Helligkeit einer LED an einem digitalen Pin zu regeln. Lässt man eine LED schnell genug blinken, nimmt das menschliche Auge das nicht mehr als Blinken wahr. Die als *Pulsweitenmodulation* (PWM) bezeichnete Technik erzeugt ein pulsierendes Signal, das die LED in sehr kurzen Abständen ein- und wieder ausschaltet. Die Spannung des Signals bleibt immer gleich, nur das Verhältnis zwischen Level *Low* (0 V) und Level *High* (+5 V) wird verändert. Das Tastverhältnis gibt das Verhältnis der Länge des eingeschalteten Zustands zur Gesamtdauer eines Schaltzyklus an.

Je kleiner das Tastverhältnis, desto kürzer ist die Leuchtzeit der LED innerhalb eines Schaltzyklus. Dadurch wirkt die LED dunkler als eine permanent eingeschaltete LED.



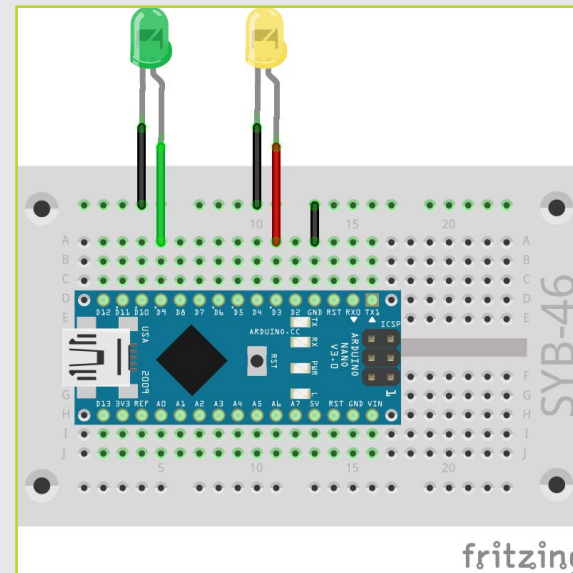
Links: Tastverhältnis 50 % High (eingeschaltet) – rechts: 20 % High (eingeschaltet). Tastverhältnis 20 %.

AUFGABE

Zwei LEDs sollen gleichmäßig heller und dunkler werden. Der gleichmäßige Zyklus zeigt deutlich, dass ein Mensch die Helligkeit von Licht bei verschiedenen Farben unterschiedlich wahrnimmt.

Pins für PWM-Signale

Die Pins 3, 5, 6, 9, 10, 11 sind auf den Schaltbildern mit einem „-Symbol gekennzeichnet. Nur diese Pins können für die PWM verwendet werden.



DAS PROGRAMM

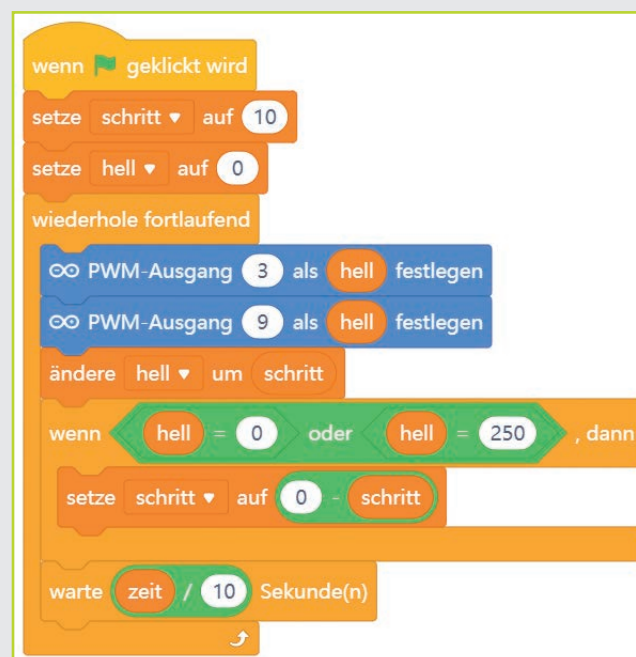
Das Programm 08-ledpwm.mblock dimmt die LEDs an den Pins D3 und D9 gemeinsam zyklisch heller und dunkler. Dazu werden einige neue Programmblöcke erstellt. Links: Tastverhältnis 50 % – rechts:

Am Anfang werden drei Variablen definiert:

- **hell** bezeichnet den PWM-Wert für die Helligkeit der LEDs.
- **schritt** gibt die Schrittweite beim Dimmen an.
- **zeit** legt die Zeit für einen Schritt fest, sie kann über einen Schieberegler eingestellt werden.

Aus der **Steuerung**-Blockpalette fügen Sie darunter eine **wiederhole fortlaufend**-Schleife an. Da hinein setzen Sie aus der Blockpalette **Pin** zwei Blöcke **PWM-Ausgang ... als ... festlegen**. Tragen Sie die entsprechenden Pins ein und ziehen Sie aus der Blockpalette **Variablen** die Variable **hell** in die anderen Felder.

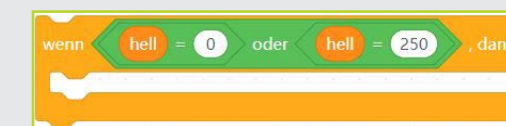
Unter die beiden blauen Blöcke hängen Sie aus der Blockpalette **Variablen** einen **ändere ... um ...**-Block, wählen im Auswahlfeld **hell** und ziehen die Variable **schritt** in das andere Feld.



Jetzt holen Sie aus der Blockpalette **Steuerung** den Block **wenn ..., dann**. In die Raute fügen Sie aus der Blockpalette **Operatoren** einen **oder**-Block ein.

Zum Einfügen nähern Sie den **oder**-Block an die Raute an, bis ein weißer Rand erscheint. Wenn Sie den Block loslassen, fügt er sich automatisch ein.

Ergänzen Sie anschließend wie abgebildet aus der grünen Blockpalette **Operatoren** wieder zwei **<... = ...>**-Blöcke und nehmen Sie die entsprechenden Eintragungen vor.



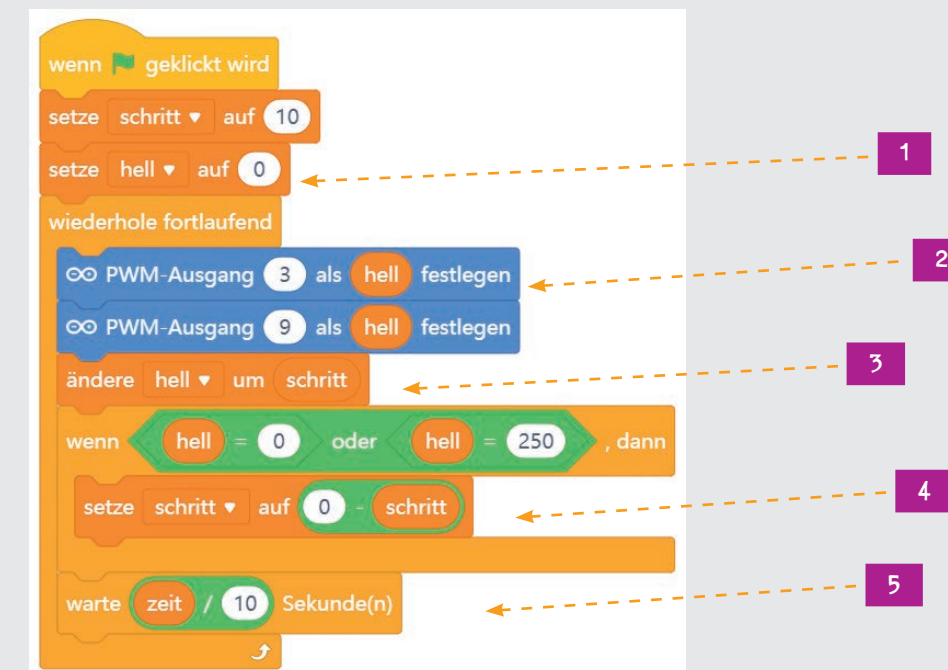
Fügen Sie in der Lücke im **oder**-Block aus der Blockpalette **Variablen** einen **setze ... auf ...**-Block mit der Variablen **schritt** ein. In das Zahlenfeld setzen Sie aus der Blockpalette **Operatoren** einen **(... - ...)**-Block und füllen die Felder mit den Variablen wie abgebildet aus.



Setzen Sie anschließend diese Gruppe in die **wiederhole fortlaufend**-Schleife ein und fügen Sie aus der Blockpalette **Steuerung** noch einen **warte ... Sekunde(n)**-Block mit der Variablen **zeit** ein, die mit einem **... / ...**-Block durch 10 geteilt wird.

Das Programm sollte jetzt wie weiter oben abgebildet aussehen.

SO FUNKTIONIERT DAS PROGRAMM



1. Zuerst werden die Variablen festgelegt: Die Schrittweite wird auf „10“ gesetzt, **hell** wird auf „0“, also LED aus, gesetzt, und die Zeit für einen Schritt wird später im Programm aus dem Schieberegler ausgelesen.
2. Als Erstes wird bei jedem Schleifendurchlauf der EndlosschleifederaktuelleWertderVariablen**hell**alsPWM-

Wert auf den Pins D3 und D9 ausgegeben. Zu Anfang ist dieser Wert 0 (LED aus).

3. Anschließend wird der Wert der Variablen **hell** um den Wert **schritt** erhöht, in diesem Fall um den Wert 10.

4. Im nächsten Schritt wird überprüft, ob der Wert von **hell** die Grenzen 0 oder 250 erreicht hat. Wenn mindestens eine Bedingung, also entweder 0 oder 250, erreicht ist, gibt der **oder**-Block den Wert **wahr** zurück und der Inhalt **dann** wird ausgeführt. Da mBlock 5 keine Möglichkeit bietet, das Vorzeichen einer Variablen umzukehren, verwenden wir den Operator '-' und subtrahieren den Wert der Variablen von 0, was das gleiche Ergebnis liefert.
5. Danach wartet das Programm in jedem Schleifendurchlauf die in der Variablen **zeit** gespeicherte Zeit. Anschließend startet die Endlosschleife neu und lie-

fert der LED einen neuen PWM-Wert. Die Variable **zeit** kann über einen Schieberegler verändert werden, um die LEDs schneller oder langsamer hell und dunkel werden zu lassen.

Einen ähnlichen Effekt können Sie erzielen, wenn Sie den Wert **schrift** erhöhen oder verringern. Wird er erhöht, wird der Wert 250 für **hell** schneller erreicht. Die LEDs scheinen in der Helligkeit zu „ruckeln“, und der Dimm-Effekt lässt nach. Wird er verringert, wird der Helligkeitswert erst über mehr Schritte erreicht, aber der Dimm-Effekt ist deutlicher.

KONSTRUIEREN MIT 3D EASY – FENSTER UND TÜREN DES WEIHNACHTSHAUSES

Heute konstruieren Sie die Fenster und Türen. Lesen Sie dazu die Anleitungen für Tag 8.

Tag 9

ERSTER TEST FÜR DIE HAUSINSTALLATION

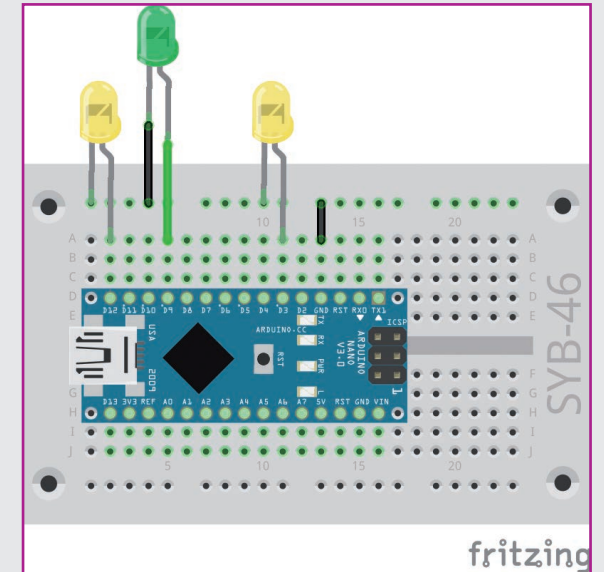
- Im Kalender: LED gelb mit Vorwiderstand

AUFGABE

Heute agieren erstmals Figuren und Nano-Board gemeinsam. Drei LEDs werden über ein Steuerpult auf dem PC gesteuert. Beim Klick auf das linke Fenster leuchtet die linke gelbe LED an Pin D12 für eine einstellbare Zeit. Beim Klick auf das rechte Fenster flackert die rechte gelbe LED an Pin D3 mithilfe eines PWM-Signals. Die Flackergeschwindigkeit lässt sich einstellen. Beim Klick auf die LED auf dem Dach blinkt die grüne LED an Pin D9. Dabei lassen sich Dauer und Blinkzeit einstellen.

Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 2 LEDs gelb mit Vorwiderständen
- 1 LED grün mit Vorwiderstand
- 1 Verbindungskabelpaar
- 1 Drahtbrücke



Die erste gelbe LED gehört zur Türklingel, die zweite gelbe LED zum Kamintaster, und die grüne LED auf dem „Dach“ könnte per Verbindungskabel angeschlossen werden.

DAS PROGRAMM

Das Programm zeigt auf der Bühne ein Haus als Steuerpult für die Beleuchtung. Es enthält drei Figuren: die beiden Fenster sowie einen Kreis für die LED auf dem Dach. Jede Figur hat zwei Kostüme, die den ausgeschalteten und den eingeschalteten Zustand der LED kennzeichnen. Bei jeder Figur werden auf der Bühne die zugehörigen Variablen angezeigt. Überall dort, wo

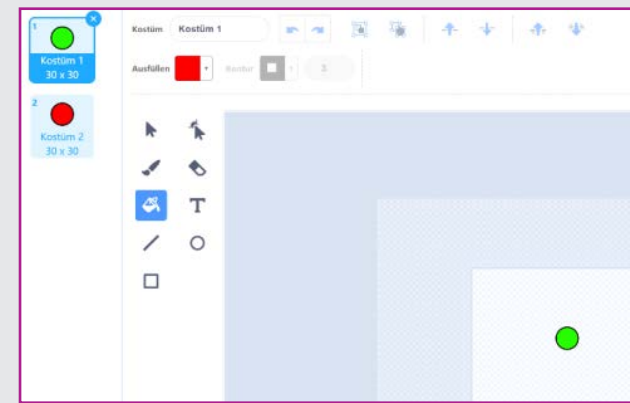
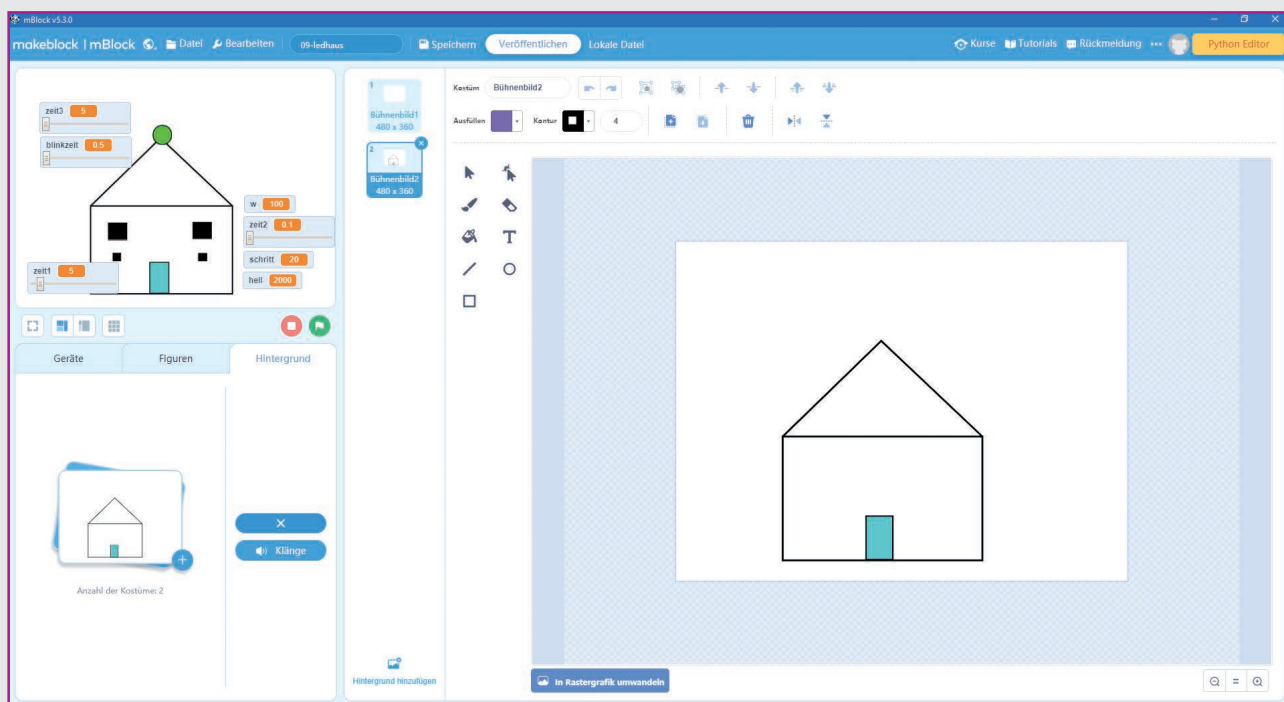
es sinnvoll ist, haben diese Variablen Schieberegler, um zum Beispiel Blinkzeiten einzustellen.

Öffnen Sie das Programm 09-ledhaus.mblock aus der Zip-Datei, die Sie an Tag 2 heruntergeladen haben, und klicken Sie auf die Registerkarte **Figuren** unterhalb der Bühne.



Links oben befindet sich die Bühne mit dem **Hintergrund**. In unserem Fall ist es der Umriss eines Häuschens mit zwei Fenstern, einer Tür, einer Türklingel (unter dem linken Fenster), einem Taster (unter dem rechten Fenster) und der LED auf dem Dach. Wenn Sie unter der Bühne auf die Registerkarte **Hintergrund** klicken, erscheint ein Button **Kostüme**. Klicken Sie darauf, können Sie verschiedene Hintergründe selbst entwerfen. Probieren Sie es einfach mal aus. Sie können alles wieder rückgängig machen. Im Notfall laden Sie einfach die Programmdatei neu.

Die Bühne wird durch Figuren (siehe Tag 2) belebt. Hier sind es geometrische Formen, die gezeichnet wurden. Sie können auch eigene Figuren auf die Bühne stellen, indem Sie einfach auf der Registerkarte **Figuren** auf **Hinzufügen** klicken. Sie können die Figuren frei auf der Bühne platzieren. In dem **Figuren**-Fenster unten befindet sich der Button **Kostüme**. Klicken Sie darauf, öffnet sich ein Fenster, in dem die Figuren mit Kostümen ausgestattet werden. Das Programm von heute wechselt beim Anklicken der jeweiligen Figur das Kostüm und signalisiert so den aktiven Zustand, beispielsweise hier für die LED auf dem Dach.



Verlassen Sie die Ansicht **Kostüme** mit einem Klick auf das blaue x-Symbol ganz unten.

Jeder der drei Figuren ist ein Programmblock zugeordnet. Er wird sichtbar, wenn Sie auf die jeweilige Figur auf der Registerkarte **Figuren** klicken.

SO FUNKTIONIERT DAS PROGRAMM

Die Programme werden diesmal nicht über das grüne Fähnchen gestartet, sondern indem die Figuren am Häuschen direkt angeklickt werden. Alle Blöcke in mBlock sind immer einer Figur oder einem Gerät zugeordnet. Die verschiedenen Figuren und Geräte können über sogenannte Nachrichten miteinander kommuni-

zieren. Dazu sendet ein Programmblock eine Nachricht an alle Objekte (Figuren und Geräte). Ein anderes Objekt kann diese Nachricht beim Empfang auswerten oder einfach ignorieren. Das Programm von heute nutzt diese Methode, um mit einer Figur, die angeklickt wird, das Gerät Nano zu steuern.

TÜRKLINGEL

Klicken Sie jetzt auf der Registerkarte **Figuren** auf **Türklingel**.

Im **Figuren**-Fenster sehen Sie das linke Programm und im Fenster **Geräte** das rechte.

Das Programm im linken Fenster steuert den Nano mit der Figur **Türklingel**.

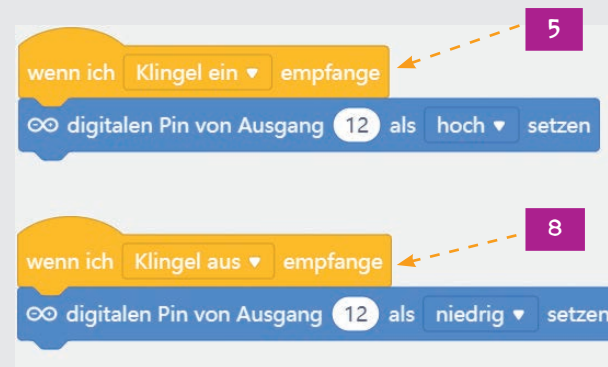
Fenster **Figuren**



Wenn das linke Fenster angeklickt wird, passiert Folgendes:

1. Die Figur wechselt zum Kostüm **Licht an**, das „Licht im Fenster geht an“, das Fenster erscheint also gelb.
2. Die Figur sendet das Signal **Klingel ein** an alle verbundenen Geräte (und Figuren).
3. Der Nano empfängt das Signal und schaltet die LED an Pin 12 ein.

Fenster **Geräte**



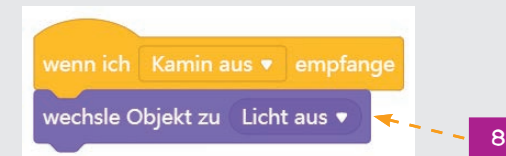
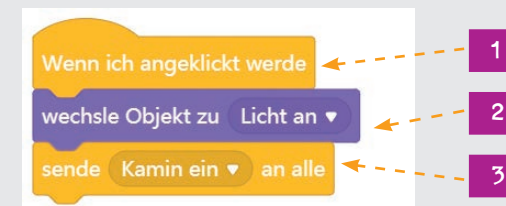
4. Die eingestellte Zeit wird gewartet. Die Zeit kann jederzeit über den Schieberegler **zeit1** verändert werden. Er hat einen Wertebereich von 1 bis 60 (Sekunden).
5. Danach wird das Signal **Klingel aus** gesendet.
6. Der Nano empfängt das Signal und schaltet die LED an Pin 12 wieder aus.
7. Anschließend wird auch das Kostüm auf **Licht aus** gesetzt. Das Fenster erscheint wieder dunkel.

KAMINTASTER

Beim Klick auf den Kamintaster wird eine Schleife gestartet, die ein PWM-Signal zyklisch verändert. Der Programmaufbau ähnelt im Wesentlichen dem Programm des 8. Tags.

Die Variable **zeit2**, die die Wartezeit zwischen zwei Veränderungen des PWM-Signals in Zehntelsekunden und damit die Flackergeschwindigkeit festlegt, kann über einen Schieberegler eingestellt werden, während das Programm läuft. Der Schieberegler hat einen Wertebereich von 1 bis 10. Der Wert wird im Programm durch 10 geteilt.

Fenster **Figuren**



Wenn das rechte Fenster angeklickt wird, passiert Folgendes:

1. Die Figur wechselt zum Kostüm **Licht an**, das „Licht im Fenster geht an“, das Fenster erscheint also gelb.
2. Die Figur sendet das Signal **Kamin ein** an alle verbundenen Geräte (und Figuren).
3. Der Nano empfängt das Signal und startet ein PWM-Signal. Dazu werden folgende Variablen verwendet:

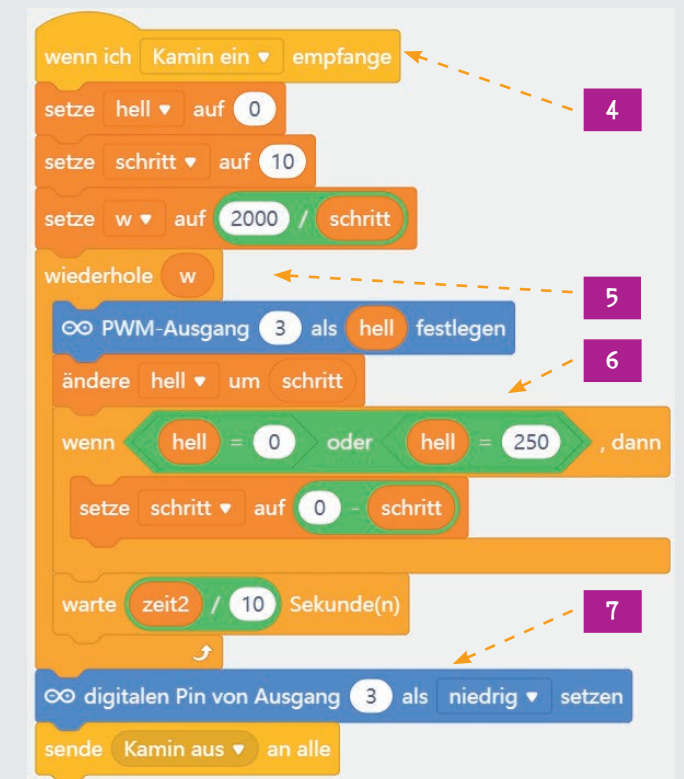
zeit2 = Zeit zwischen den Schritten – über Schieberegler einstellbar.

hell = Helligkeit der LED (durch PWM), anfangs 0 = LED aus.

schritt = Schrittweite – der PWM-Wert wird in einem Schleifendurchlauf um 10 Schritte verändert.

w = Anzahl der Wiederholungen der Schleife, hier sollen insgesamt 2.000 Schritte gemacht werden.

Fenster **Geräte**



4. Die LED wird schrittweise heller.
5. Wenn die Helligkeit 250 beträgt, wird durch die Wenn-dann-Abfrage die Helligkeit wieder auf 0 gesetzt. Nach der eingestellten Zeit beginnt der Zyklus von Neuem, bis die in **w** festgelegte Anzahl an Wiederholungen erreicht ist.
6. Anschließend wird die LED ausgeschaltet und ein Signal **Kamin aus** gesendet.
7. Wenn die Figur das Signal empfangen hat, wird der Ausgangszustand – „Licht im Fenster aus“ – wiederhergestellt. Die Figur wechselt zum Kostüm **Licht aus**.

Wie oft der Kamin flackern soll, lässt sich über die Schrittzahl einstellen. In diesem Fall sind es 2.000 Schritte. Das heißt, die LED flackert 20-mal (2000 / 10). Der Ablauf lässt sich gut in der Variablen **hell**, die auf der Bühne rechts neben dem Häuschen angezeigt wird, verfolgen.

LED AUF DEM DACH

Ein Klick auf die LED auf dem Dach lässt sie blinken. Dabei lassen sich die Gesamtzeit **zeit3** und auch, wie lange die LED bei jedem Blinkvorgang eingeschaltet ist, über **blinkzeit** einstellen.

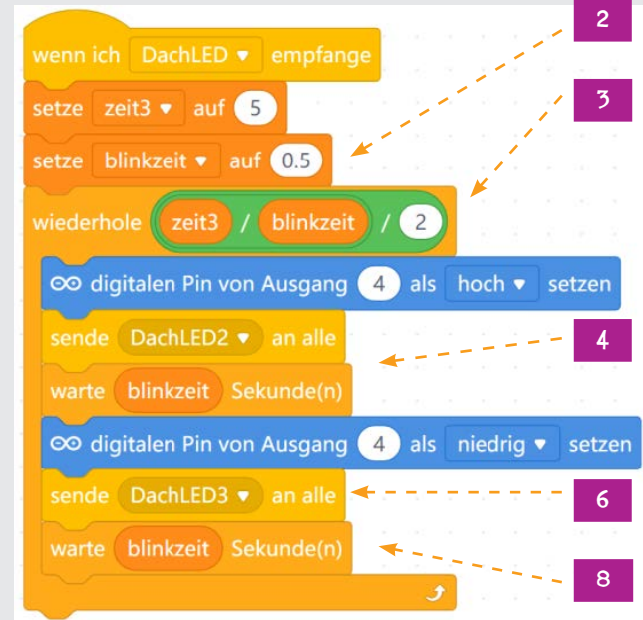
Um die Anzahl der Wiederholungen zu berechnen, wird zunächst die Gesamtzeit durch die Blinkzeit geteilt. Das

Ergebnis wird noch einmal mit 5 multipliziert, da die Blinkzeit in Zehntelsekunden gerechnet wird und jeder Schleifendurchlauf zwei Blinkzeiten dauert ($10 / 2 = 5$). Während der ersten Blinkzeit ist die LED an Pin 9 eingeschaltet, während der zweiten ausgeschaltet.

Fenster **Figuren**



Fenster **Geräte**



1.

Wenn Sie die Figur **DachLED** anklicken, wird ein Signal **DachLED** gesandt.
2.

Der Nano empfängt das Signal und errechnet die Anzahl an Wiederholungen für die Blinkschleife.
3.

In jedem Schleifendurchlauf wird die Dach-LED eingeschaltet und ein Signal **DachLED ein** gesendet.
4.

Die Figur **DachLED** empfängt das Signal und wechselt auf das grüne Kostüm **Licht an**.
5.

Nach der Blinkzeit wird die Dach-LED ausgeschaltet und das Signal **DachLED aus** gesendet.
6.

Die Figur **DachLED** empfängt das Signal und wechselt auf das schwarze Kostüm **Licht aus**.
7.

Wieder wird die Blinkzeit abgewartet, und der Zyklus beginnt erneut.

KONSTRUIEREN MIT 3D EASY – HALTERUNG FÜR DAS BATTERIEFACH IM WEIHNACHTSHAUS

Heute arbeiten Sie weiter an Ihrem Weihnachtshaus. Sie konstruieren die Halterung für das Batteriefach. Lesen Sie dazu die Anleitung zu Tag 9.

Tag 10

LAUFLICHTER

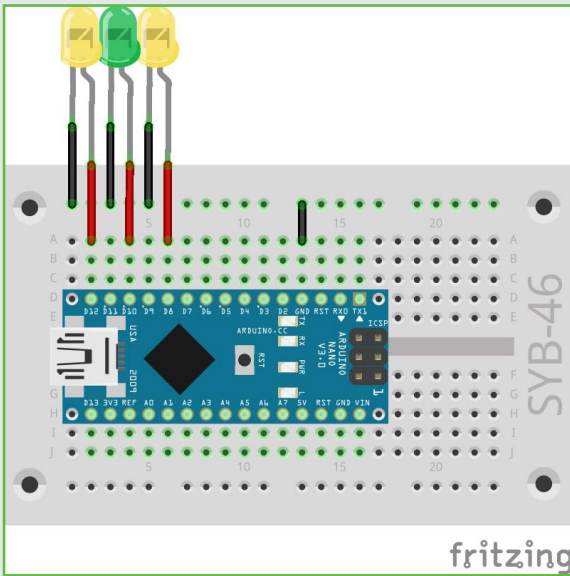
● Im Kalender: Verbindungskabelpaar

AUFGABE

Lauflichter sind immer wieder beliebte Effekte, nicht nur für Werbung und Partyräume. Drei LEDs sollen als Lauflicht nacheinander aufleuchten. Über eine Variable steuert man die Geschwindigkeit des Lauflichts.

Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 2 LEDs gelb mit Vorwiderständen
- 1 LED grün mit Vorwiderstand
- 3 Verbindungskabelpaare
- 1 Drahtbrücke



DAS PROGRAMM

Das Programm lässt die drei LEDs zyklisch als Lauflicht aufleuchten. Die Pinnummern haben jeweils einen Abstand von 2 voneinander. So können sie einfach in einer Schleife hochgezählt werden.

SO FUNKTIONIERT DAS PROGRAMM

Bei der Verwendung von Variablen kann mBlock 5 nicht gleich am Anfang erkennen, welche Pins als Ausgänge verwendet werden sollen, und initialisiert sie erst bei ihrer ersten Verwendung. Daher kann man von keinem eindeutig definierten Zustand der LEDs zum Programmstart ausgehen. Um das zu vermeiden, schaltet das Programm am Anfang alle LEDs mit den Blöcken **digitalen Pin von Ausgang ... als niedrig setzen** aus und initialisiert sie damit.

Tag 11

ZUFALLSGENERATOR

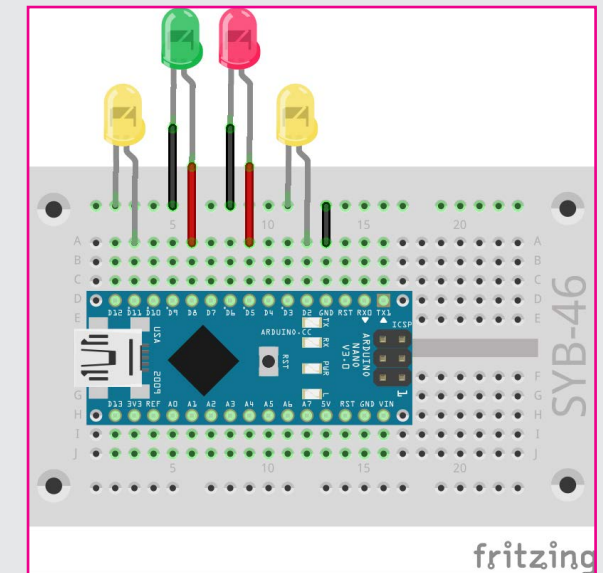
● Im Kalender: LED rot mit Vorwiderstand

AUFGABE

Vier LEDs sollen in zufälliger Reihenfolge blinken.

Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 2 LEDs gelb mit Vorwiderständen
- 1 LED grün mit Vorwiderstand
- 1 LED rot mit Vorwiderstand
- 2 Verbindungskabelpaare
- 1 Drahtbrücke

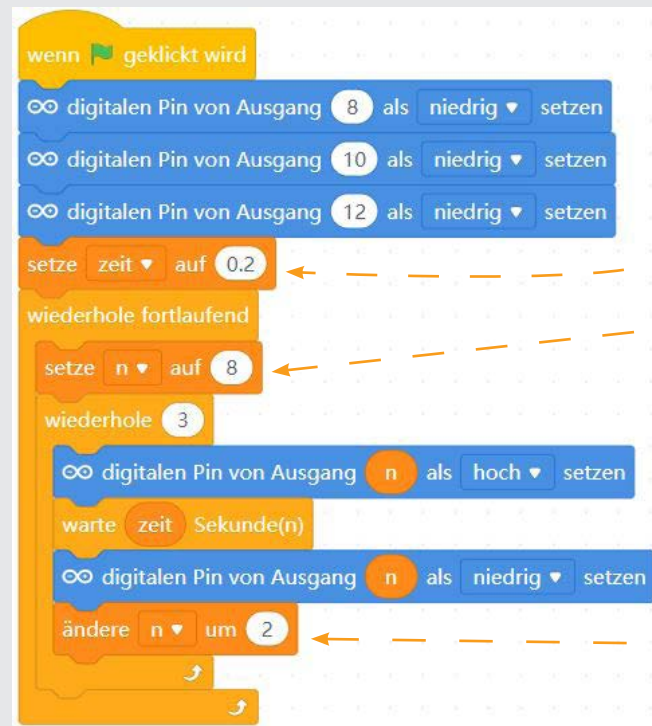


DAS PROGRAMM

Das Programm 11-led4zufall.mblock lässt vier LEDs zufällig abwechselnd blinken. Die Pinnummern haben zur einfacheren Berechnung wieder alle die gleichen Abstände.

Wie entstehen Zufallszahlen?

Gemeinhin denkt man, in einem Programm könne nichts zufällig geschehen – wie also kann ein Programm dann in der Lage sein, zufällige Zahlen zu generieren? Teilt man eine große Primzahl durch irgendeinen Wert, ergeben sich ab der x-ten Nachkommastelle Zahlen, die kaum noch vorhersehbar sind. Sie ändern sich auch ohne jede Regelmäßigkeit, wenn man den Divisor regelmäßig erhöht. Dieses Ergebnis ist zwar scheinbar zufällig, lässt sich aber durch ein identisches Programm oder den mehrfachen Aufruf des gleichen Programms jederzeit reproduzieren. Nimmt man aber eine aus einigen dieser Ziffern zusammengebaute Zahl und teilt sie wiederum durch eine Zahl, die sich aus der aktuellen Uhrzeitsekunde oder dem Inhalt einer beliebigen Speicherstelle des Computers ergibt, kommt ein Ergebnis heraus, das sich nicht reproduzieren lässt und daher als Zufallszahl bezeichnet wird.



Zeit, in der jede LED an ist.

Zuerst wird LED an Pin 8 angesprochen

LED ist an für 0,2 Sekunden ...

und wird danach ausgeschaltet.

Anschließend wird n um 2 erhöht sowie LED an Pin 10 wird angesprochen und das Programm wiederholt sich 3-mal in einer fortlaufenden Schleife.

Wenn Pin 12 „durch“ ist, wird wieder mit Pin 8 angefangen, da es keinen höheren als Pin 12 gibt.

Die Leuchtzeit der einzelnen LEDs wird über die Variable **zeit** geregelt. Diese Variable kann über einen Schieberegler von 1 bis 20 eingestellt werden. Der Wert legt fest, wie lange in Zehntelsekunden jede einzelne LED leuchtet. Deshalb wird die Variable **zeit** jedes Mal durch 10 geteilt.

KONSTRUIEREN MIT 3D EASY – HALTERUNG FÜR DEN PIEZO-SUMMER

Am heutigen Tag fertigen Sie die Halterung für den Piezo-Summer an. Lesen Sie dazu die Anleitung für Tag 10.

WIDERSTÄNDE UND TASTER

● Im Kalender: Taster und Widerstand 10 kOhm (braun – schwarz – orange)

WIDERSTÄNDE UND IHRE FARBCODES

Widerstände werden unter anderem zur Strombegrenzung an empfindlichen elektronischen Bauteilen sowie als Vorwiderstände für LEDs verwendet. Die Maßeinheit für Widerstände ist Ohm. 1.000 Ohm entsprechen einem Kiloohm, abgekürzt kOhm. 1.000 kOhm entsprechen einem Megaohm, abgekürzt MOhm. Oft wird für die Einheit Ohm auch das Omega-Zeichen Ω verwendet.

Die farbigen Ringe auf den Widerständen geben den Widerstandswert an. Mit etwas Übung sind sie deutlich leichter zu erkennen als die winzig kleinen Zahlen, die man nur noch auf ganz alten Widerständen findet.

Die meisten Widerstände haben vier solcher Farbringe. Die ersten beiden Farbringe stehen für die Ziffern, der dritte bezeichnet einen Multiplikator und der vierte die Toleranz. Dieser Toleranzring ist meistens gold- oder silberfarben – Farben, die auf den ersten Ringen nicht vorkommen. Dadurch ist die Leserichtung immer eindeutig. Der Toleranzwert selbst spielt in der Digitalelektronik kaum eine Rolle. Die Tabelle zeigt die Bedeutung der farbigen Ringe auf Widerständen.

In welcher Richtung ein Widerstand eingebaut wird, ist egal. Bei LEDs dagegen spielt die Einbaurichtung eine wichtige Rolle.

Farbe	Widerstandswert in Ohm			
	1. Ring (Zehner)	2. Ring (Einer)	3. Ring (Multiplikator)	4. Ring (Toleranz)
Silber			$10^{-2} = 0,01$	$\pm 10 \%$
Gold			$10^{-1} = 0,1$	$\pm 5 \%$
Schwarz		0	$10^0 = 1$	
Braun	1	1	$10^1 = 10$	$\pm 1 \%$
Rot	2	2	$10^2 = 100$	$\pm 2 \%$
Orange	3	3	$10^3 = 1.000$	
Gelb	4	4	$10^4 = 10.000$	
Grün	5	5	$10^5 = 100.000$	$\pm 0,5 \%$
Blau	6	6	$10^6 = 1.000.000$	$\pm 0,25 \%$
Violett	7	7	$10^7 = 10.000.000$	$\pm 0,1 \%$
Grau	8	8	$10^8 = 100.000.000$	$\pm 0,05 \%$
Weiß	9	9	$10^9 = 1.000.000.000$	



SO FUNKTIONIERT DAS PROGRAMM

Am Anfang werden die vier Pins der LEDs ausgeschaltet und dabei gleichzeitig initialisiert. Danach startet eine Endlosschleife.

Im ersten Schritt jedes Durchlaufs der Endlosschleife wird eine Zufallszahl benötigt, die einen der vier verwendeten Pins D2, D5, D8 oder D11 bezeichnet. Da der Block **wähle eine zufällige Zahl zwischen ...und ...** nur Zufallszahlen in einem begrenzten, fortlaufend nummerierten Bereich erzeugen kann, brauchen wir eine Umrechnungsformel: Zunächst wird die Variable **n** auf eine Zufallszahl zwischen 0 und 3 gesetzt. Diese Zahl wird mit drei multipliziert, anschließend wird zwei addiert. Das Ergebnis ist eine der Zahlen 2, 5, 8 oder 11 und gibt die Pinnummer der einzuschaltenden LED an.

Die Geschwindigkeit des Farbwechsels wird wie im letzten Programm über die einstellbare Variable **zeit** gesteuert, die in jedem Schleifendurchlauf einmal abgefragt und durch 10 geteilt wird, um Zeiten in Zehntelsekunden einstellen zu können.

Die zufällig gewählte LED wird für die eingestellte Zeit eingeschaltet und danach wieder ausgeschaltet. Im nächsten Schleifendurchlauf wird ein weiteres Mal eine neue LED zufällig gewählt. Dabei kann es durchaus passieren, dass mehrmals hintereinander die gleiche LED aufleuchtet.

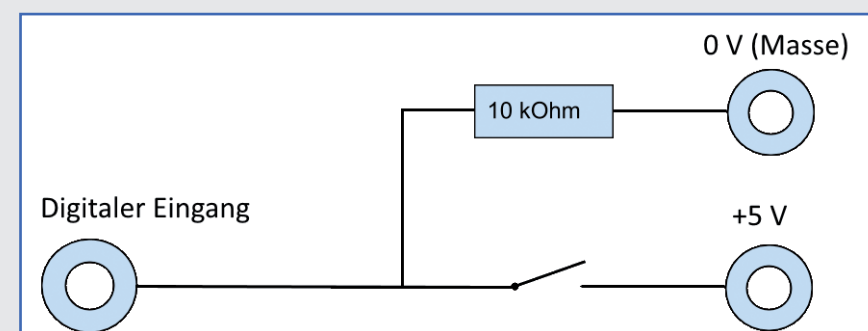
KONSTRUIEREN MIT 3D EASY – WEITERE ARBEIT AM WEIHNACHTSHAUS

Am heutigen Tag konstruieren Sie die Öffnungen für den Schall des Summers und zur Bedienung des Nano. Lesen Sie dazu die Anleitung zu Tag 11.

Taster

Digitale Pins können nicht nur Daten ausgeben, zum Beispiel über LEDs, sondern auch zur Dateneingabe verwendet werden. Zur Eingabe nutzen wir im heutigen Projekt einen Taster, der als Türklingelknopf in die Giebelwand neben der Tür gesteckt und von innen wie die LEDs mit einem Verbindungskabelpaar am Steckbrett angeschlossen wird. Solange die Taste gedrückt wird, sind beide Anschlüsse miteinander verbunden. Im Gegensatz zu einem Schalter rastet ein Taster nicht ein. Die Verbindung wird beim Loslassen sofort wieder getrennt.

Liegt auf einem digitalen Eingang ein +5-V-Signal an, wird es als logisch *wahr* ausgewertet.

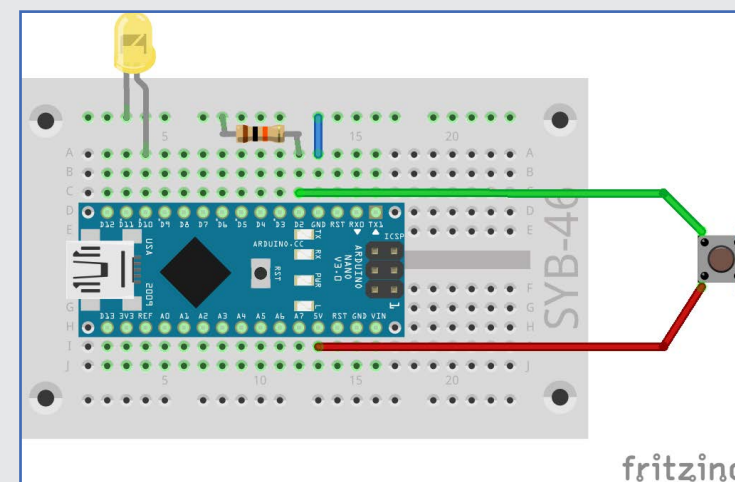


Schaltschema eines Tasters mit Pull-down-Widerstand.

AUFGABE

Beim Drücken eines Tasters soll das Licht einige Sekunden lang leuchten.

Beachten Sie, dass der Taster auf der „unteren Seite“ des Nano an den 5-V-Pin angeschlossen wird.



Sollte der Taster zu locker in den Verbindungskabeln stecken, isolieren Sie ein Stück Schalt-draht ab, stecken es zu dem Taster in das Verbindungskabel und schneiden es danach pass-send zu.

Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 1 LED gelb mit Vorwiderstand
- 1 Taster
- 1 10-kOhm-Widerstand (braun – schwarz – orange; siehe Tabelle: 10×10^3)
- 1 Verbindungskabelpaar
- 1 Drahtbrücke

DAS PROGRAMM

Zuerst muss das grüne Fähnchen angeklickt werden. Erst dann reagiert das Programm 12-taster.mblock auf den Tastendruck. Beim Drücken der Taste am digitalen Pin D2 leuchtet die LED während der eingestell-

ten Zeit. Das Fenster verändert dabei auch sein Kostüm („Licht an im Fenster“). Beim Anklicken des Fensters in der Figur passiert in diesem Programm nichts.

SO FUNKTIONIERT DAS PROGRAMM

Fenster **Figuren**

Fenster **Geräte**



Die Funktionen der Blöcke sind im Wesentlichen aus den vorherigen Programmen bekannt.

Neu ist der Block **Digitalpin lesen** in der **wenn ... dann**-Abfrage.

Das Drücken eines Tasters, genauer der Zustand eines digitalen Eingangs, kann nicht als Ereignis abgefragt werden, wie zum Beispiel das Anklicken eines Objekts.

Es gibt keinen Block in mBlock 5, der beim Drücken eines Tasters automatisch startet. Daher muss zu Anfang immer der Block mit dem grünen Fähnchen angeklickt

werden. Aus diesem Grund verwenden wir auch eine Endlosschleife. Sie prüft in jedem Durchlauf, ob der Taster gedrückt wird. Nach einer Zehntelsekunde erfolgt der nächste Prüfdurchlauf.

Der Block **wenn ... dann** von der Blockpalette **Steuerung** führt alle Blöcke innerhalb der Klammer aus, wenn die Bedingung *wahr* ist, also der Taster gedrückt wurde.

Die Blöcke innerhalb der Klammer des **wenn ... dann**-Blocks sind bereits bekannt. Sie schalten das Kostüm des Fensters um und die LED ein. Nach der eingestellten Zeit wird beides wieder zurückgesetzt.

KONSTRUIEREN MIT 3D EASY – FERTIGSTELLUNG DES WEIHNACHTSHAUSES

Heute verpassen Sie Ihrem Weihnachtshaus den letzten Schliff und schließen damit auch das größte Projekt des Adventskalenders ab. Lesen Sie dazu die Anleitung zu Tag 12.

Tag 13

KLINGEL MIT PIEZO-SUMMER

● Im Kalender: Piezo-Summer

Piezo-Summer

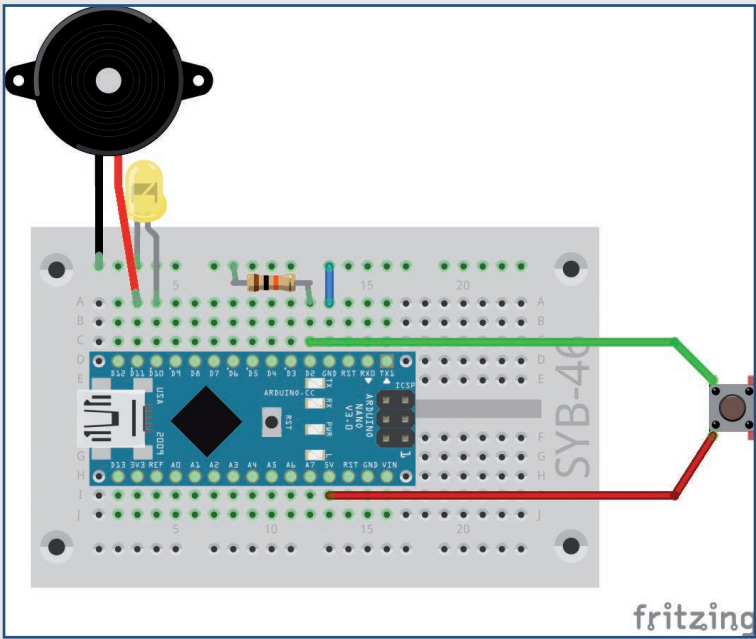
Der heute im Adventskalender enthaltene Piezo-Summer macht elektrische Schwingungen hörbar. Legt man eine pulsierende Gleichspannung zwischen die beiden Pole des Summers, wird er in Schwingung versetzt. Je nach Frequenz sind einzelne Klicks oder ein durchgän-

giger Ton zu hören. Frequenzen von wenigen Hertz (Hz, Schwingungen pro Sekunde) nimmt das menschliche Ohr noch als einzelne Töne wahr, Frequenzen zwischen etwa 20 Hz und 16 kHz werden als durchgehender Ton unterschiedlicher Höhe wahrgenommen.

AUFGABE

Der Taster ist jetzt die Türklingel am Haus. Stecken Sie die beiden Drähte von außen durch die beiden Löcher neben der Haustür und schließen Sie den Taster über ein Verbindungskabelpaar im Haus am Steckbrett an. Drückt jemand auf die Klingel, geht im Haus das Licht an, und die Klingel ertönt.

- Bauteile
- 1 Nano-Board
 - 1 Steckbrett
 - 1 LED gelb mit Vorwiderstand
 - 1 Taster
 - 1 10-kOhm-Widerstand (braun – schwarz – orange; siehe Tabelle: 10×10^3)
 - 1 Verbindungskabelpaar
 - 1 Drahtbrücke



DAS PROGRAMM

Wie im Programm von gestern leuchtet auch im Programm 13-*klengel.mblock* beim Druck auf den Taster die LED eine Zeitlang. Zusätzlich ertönen auf dem Piezo-Summer einige Töne, wobei Tondauer und Wiederholung über Schieberegler einstellbar sind.



SO FUNKTIONIERT DAS PROGRAMM

Das Programm wartet – wie das Programm von gestern – in einer Endlosschleife darauf, dass der Taster gedrückt wird.

Nachdem die LED eingeschaltet ist, startet eine **wiederhole**-Schleife, die auf dem Piezo-Summer Pieptöne ausgibt. Wie viele Töne das sein sollen, können Sie über den Schieberegler der neuen Variablen **klengel** festlegen.

In jedem Schleifendurchlauf wird am in der Variablen **summer** gespeicherten Pin ein PWM-Signal ausgege-

ben. Die Dauer wird mit dem Schieberegler der Variablen **zeit** in Zehntelsekunden festgelegt und deshalb wieder einmal durch 10 geteilt. Nach dem Ton wird das PWM-Signal ausgeschaltet. Das Programm wartet noch einmal die gleiche Zeit, bevor die Schleife den nächsten Ton abspielt.

Zum Schluss wird die LED ausgeschaltet, und die Hauptschleife wartet darauf, dass der Taster erneut gedrückt wird.

KONSTRUIEREN MIT 3D EASY – DER TANNENBAUM, TEIL 1

Heute starten Sie ein neues Projekt: einen Tannenbaum. Beginnen Sie mit seinem Grundgerüst. Lesen Sie dazu die Anleitung zu Tag 13.

Tag 14

NANO MIT DEM PC STEUERN

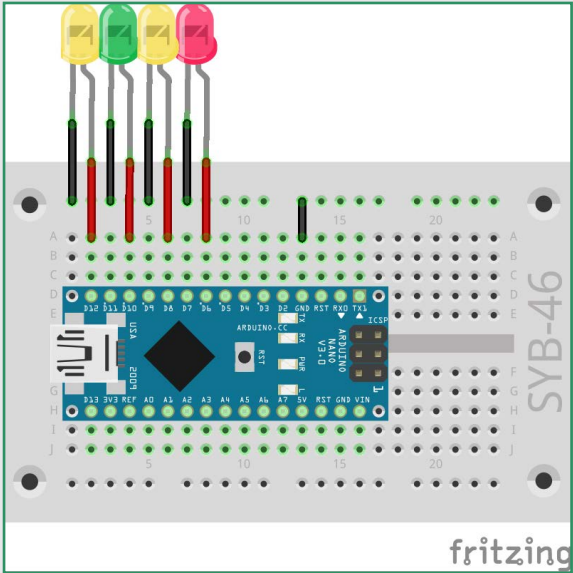
Im Kalender: Verbindungskabelpaar

AUFGABE

Beim Druck auf die Pfeiltaste nach rechts auf der Tastatur sollen vier LEDs nacheinander als Lauflicht leuchten. Die Pfeiltaste nach links lässt das Lauflicht in umgekehrter Richtung laufen. Beim Drücken der Leertaste blinken alle vier LEDs gleichzeitig.

Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 2 LEDs gelb mit Vorwiderständen
- 1 LED rot mit Vorwiderstand
- 1 LED grün mit Vorwiderstand
- 1 Verbindungskabelpaar
- 1 Drahtbrücke



Die Verbindungskabelpaare können zur späteren Installation in dem Häuschen genutzt werden.

DAS PROGRAMM

Das Programm 14-*lauflichtpc.mblock* besteht aus vier voneinander unabhängigen Programmblöcken. Das Hauptprogramm schaltet beim Klick auf das grüne Fähnchen alle vier LEDs aus. Die anderen drei Blöcke reagieren auf verschiedene Tasten der Tastatur.



SO FUNKTIONIERT DAS PROGRAMM

Beim Drücken auf das grüne Fähnchen wird die Variable **typ** auf 0 gesetzt. Dieser Typ legt eines von drei möglichen Blinkmustern fest. Ein Druck auf die Leertaste, den Pfeil nach rechts oder den Pfeil nach links setzt die Variable **typ** auf 0, 1 oder 2.

In einer Endlosschleife wird diese Variable ständig abgefragt, und abhängig davon läuft einer der Lauflichttypen genau einmal ab.

Wurde keine andere Taste gedrückt, läuft dasselbe Lauflicht ein weiteres Mal.

Hat der Benutzer eine der drei Tasten gedrückt, startet ein anderer Lauflichttyp.

Alle LEDs blinken

Nach Drücken der Leertaste werden die vier LEDs eingeschaltet. Bedingt durch die langsame Kommunikation mit dem Arduino, sieht man die LEDs nacheinander aufleuchten. Nach Ablauf der in der Variablen **zeit** festgelegten Zeit werden die LEDs wieder ausgeschaltet. Diese Variable kann über einen Schieberegler interaktiv in Zehntelsekunden eingestellt werden, während das Programm läuft. Nach einer weiteren Wartezeit wie-

derholt sich die Hauptschleife mit der Abfrage der Variablen **typ** wieder.

Lauflicht nach links

Beim Drücken der Pfeiltaste nach links läuft das Lauflicht nach links in Richtung aufsteigender Pinnummern von der rechten LED an Pin D6 über D8 und D10 bis zur linken LED an Pin D12. Dazu läuft in der Endlosschleife eine weitere Schleife, die viermal wiederholt wird und in jedem Durchlauf die in der Variablen **n** gespeicherte Pinnummer um zwei erhöht, nachdem die betreffende LED kurz ein- und wieder ausgeschaltet wurde. Nach vier Durchläufen der inneren Schleife setzt die Endlosschleife die Pinnummer **n** wieder auf 6. Danach wiederholt sich die Hauptschleife mit der Abfrage der Variablen **typ**.

Lauflicht nach rechts

Das Lauflicht nach rechts funktioniert fast genauso. Es startet aber mit der LED ganz links an Pin D12. Dann wird in jedem Durchlauf die in der Variablen **n** gespeicherte Pinnummer um zwei verringert bis zur LED an Pin D6 ganz rechts.

KONSTRUIEREN MIT 3D EASY – DER TANNENBAUM, TEIL 2

Gestern haben Sie bereits das Grundgerüst des Tannenbaums konstruiert. Heute erstellen Sie die Aussparungen für die LEDs. Lesen Sie dazu die Anleitung für Tag 14.

DIE RGB-LED

● Im Kalender: RGB-LED mit Vorwiderstand

RGB-LEDs

Eine normale LED leuchtet immer nur in einer Farbe. Die im Adventskalender verwendeten RGB-LEDs können wahlweise in mehreren Farben leuchten. Hier wurden im Prinzip drei LEDs in verschiedenen Farben in ein transparentes Gehäuse eingebaut. Jede dieser drei LEDs hat eine eigene Anode, über die sie mit einem GPIO-Pin verbunden wird. Die Kathode, die mit der Masseleitung verbunden wird, ist nur einmal vorhanden. Deshalb hat eine RGB-LED vier Anschlussdrähte.

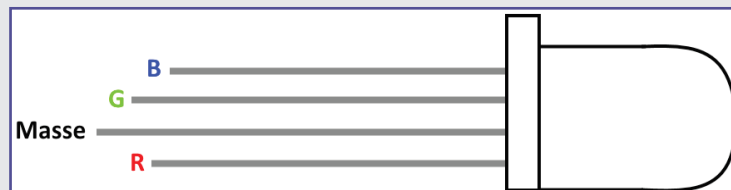
Die Anschlussdrähte der RGB-LEDs sind unterschiedlich lang, um sie eindeutig kenntlich zu machen. Im Gegensatz zu einer normalen LED ist die Kathode hier der längste Draht.

RGB-LEDs funktionieren wie drei einzelne LEDs und brauchen deshalb auch drei 220-Ohm-Vorwiderstände (rot – rot – braun). In den RGB-LEDs in diesem Adventskalender sind die Vorwiderstände bereits eingebaut.

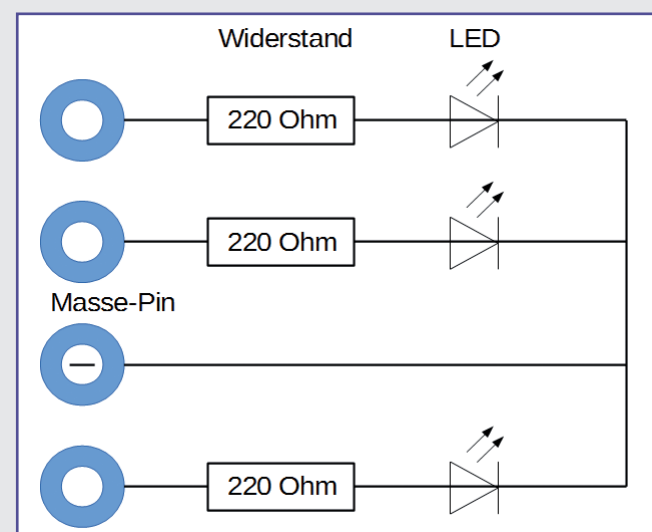
Auf dem Dach des Hauses sind Positionen mit vier Löchern für RGB-LEDs vorgesehen.

Bauteile

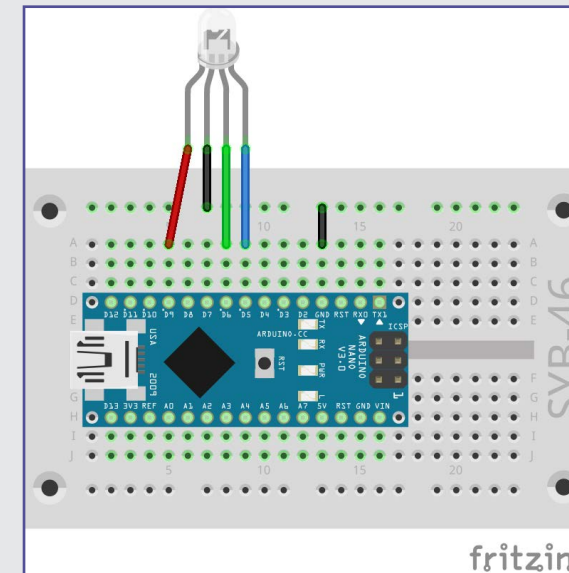
- 1 Nano-Board
- 1 Steckbrett
- 1 RGB-LED mit Vorwiderständen
- 2 Verbindungskabelpaare
- 1 Drahtbrücke



Anschlusspins einer RGB-LED.



Schaltplan für eine RGB-LED mit drei Vorwiderständen.



Die Verbindungskabelpaare können zur späteren Installation in dem Häuschen genutzt werden, erleichtern aber jetzt auch das Einsetzen der LED.

Das längste „Beinchen“ befindet sich am GND-Pin des Nano. Das zweitlängste ist für Grün (Pin 6) zuständig. Direkt neben dem Masseanschluss wird Rot (Pin 9) und direkt neben Grün wird Blau (Pin 5) geschaltet.

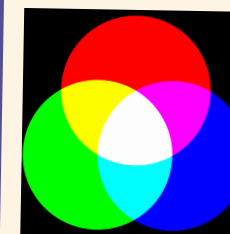
ACHTUNG! Besonders Grün leuchtet sehr intensiv! Nicht direkt in die LED schauen!

AUFGABE

Die RGB-LED soll nacheinander in den Grundfarben und Mischfarben leuchten.

Additive Farbmischung

RGB-LEDs verwenden die sogenannte additive Farbmischung. Dabei werden die drei Lichtfarben Rot, Grün und Blau addiert und ergeben so reines Weiß. Im Gegensatz dazu nutzt ein Farbdrucker die subtraktive Farbmischung. Jede Farbe wirkt auf einem weißen Blatt wie ein Filter, der einen Teil des weiß reflektierten Lichts wegnimmt (= subtrahiert). Drückt man alle drei Druckerfarben übereinander, ergibt das Schwarz, da kein Licht mehr reflektiert wird.



DAS PROGRAMM

In einer Endlosschleife werden nacheinander verschiedene digitale Pins ein- und ausgeschaltet. In diesem Fall handelt es sich um die drei Farbkomponenten der RGB-LED.



SO FUNKTIONIERT DAS PROGRAMM

Im Programm 15-rgbled.mblock leuchten durch abwechselndes Ein- und Ausschalten immer mal eine, mal zwei Farbkomponenten. Dadurch wechselt die RGB-LED zwischen sechs verschiedenen Farben hin und her. Die Variablen **r**, **g**, **b** bezeichnen die drei Anschlusspins der RGB-LED. So ist im Programm leicht zu erkennen, welche Farbkomponenten ein- oder ausgeschaltet werden. Außerdem hat diese Methode den Vorteil, dass sie

leicht für andere Pins umgebaut werden kann. Die Pinnummern sind nur an einer Stelle im Programm eingetragen – bei der Zuweisung der Variablen.

Die Geschwindigkeit des Farbwechsels wird auch hier über die Variable **zeit** gesteuert, die über einen Schieberegler in Zehntelsekunden eingestellt werden kann und dann bei jedem Farbwechsel abgefragt wird.

KONSTRUIEREN MIT 3D EASY – DER TANNENBAUM, TEIL 3

Heute erstellen Sie die Spitze und den Stern Ihres Tannenbaums und stellen somit Ihr drittes Projekt fertig. Lesen Sie dazu die Anleitung zu Tag 15.

Tag 16

RGB-FARBMISCHUNG MIT PWM

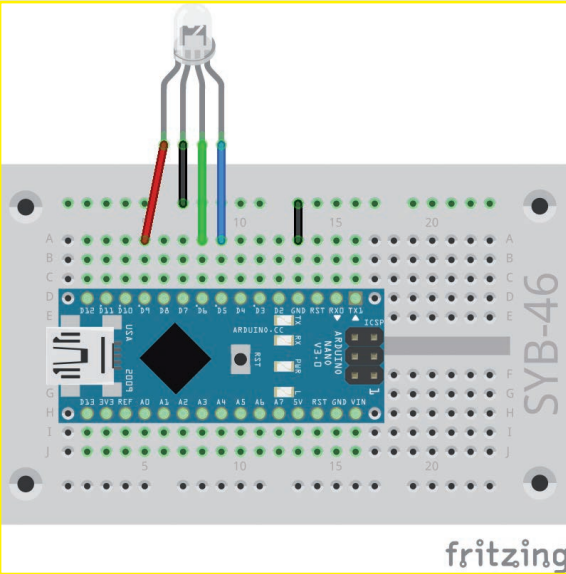
Im Kalender: Verbindungskabelpaar

AUFGABE

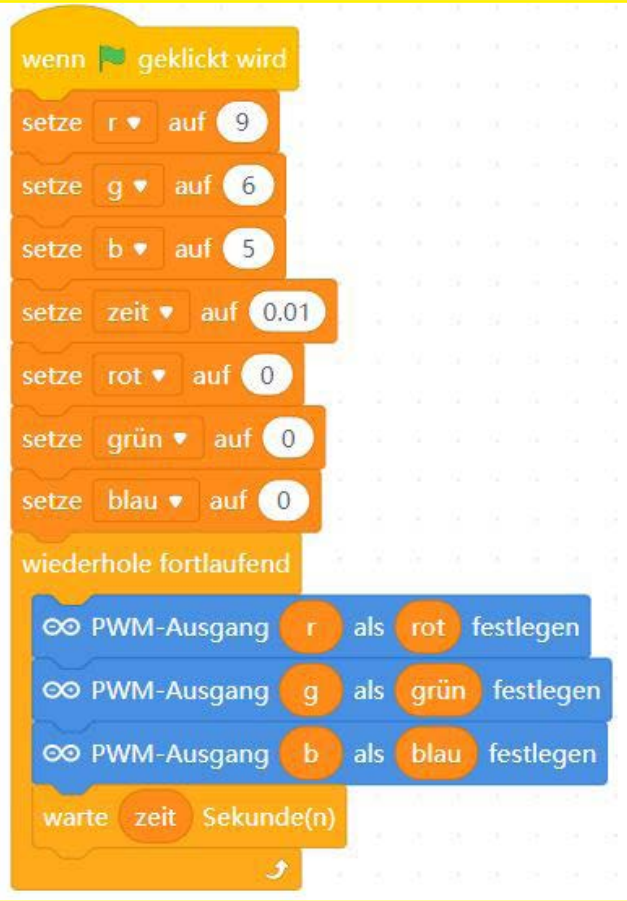
Über PWM-Signale lassen sich auf RGB-LEDs noch deutlich mehr Farben anzeigen.

Über drei Schieberegler sollen die drei Farbkomponenten unabhängig voneinander eingestellt und auch Mischfarben erzeugt werden können.

- Bauteile**
- 1 Nano-Board
 - 1 Steckbrett
 - 1 RGB-LED mit Vorwiderständen
 - 2 Verbindungskabelpaare
 - 1 Drahtbrücke



Der Schaltungsaufbau ist der gleiche wie am vorherigen Tag.



DAS PROGRAMM

Die Variablen *r*, *g*, *b* bezeichnen im Programm 16-rgbledpwm.mblock wieder die drei Anschlusspins der RGB-LED. Die Variablen *rot*, *grün* und *blau* zeigen auf der Bühne Schieberegler, mit denen sich die drei Farbkomponenten einstellen lassen.

SO FUNKTIONIERT DAS PROGRAMM

Am Anfang werden die drei Farben der RGB-LEDs auf 0 gesetzt, also ausgeschaltet. Erst wenn die Schieberegler betätigt werden, erscheinen verschiedene Farben.

Die Endlosschleife setzt die PWM-Pins der drei Farbkomponenten auf die über die Schieberegler eingestellten Werte. Danach wartet sie eine Hundertstelsekunde und liest die Werte der Schieberegler wieder neu ein. So kann die Farbe der RGB-LED vom Benutzer in Echtzeit verändert werden.

KONSTRUIEREN MIT 3D EASY – DER WEIHNACHTSSTERN, TEIL 1

Heute beginnen Sie ein neues Projekt – einen Weihnachtsstern. Sie können ihn als Beigabe zu den Geschenken verwenden oder auf dem Dach des Häuschens platzieren.

Lesen Sie dazu die Anleitung zu Tag 16.

Tag 17

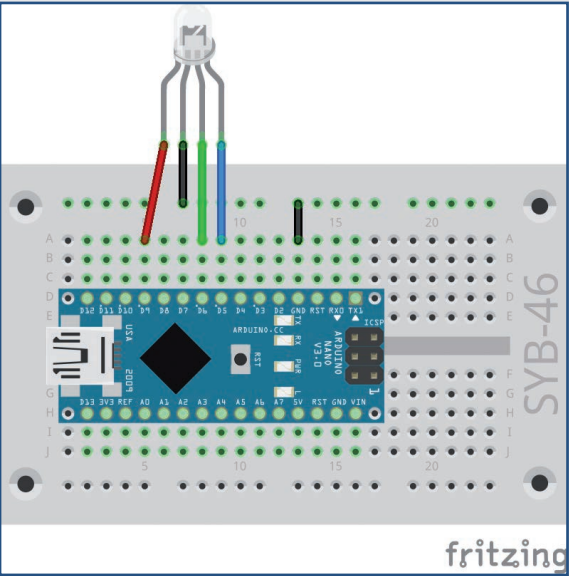
RGB-LED LEUCHTET IN REGENBOGENFARBEN

● Im Kalender: Verbindungskabelpaar

AUFGABE

Das Programm soll eine RGB-LED zyklisch in allen Farben des Farbspektrums leuchten lassen.

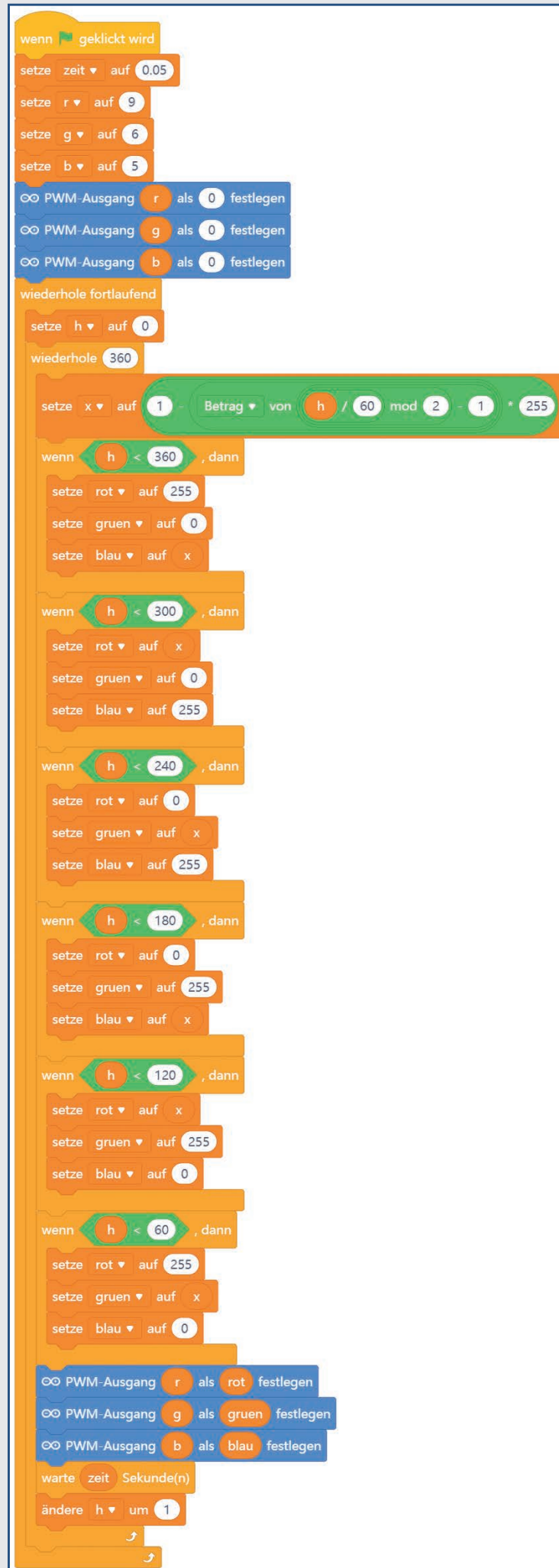
- Bauteile
- 1 Nano-Board
 - 1 Steckbrett
 - 1 RGB-LED mit Vorwiderständen
 - 2 Verbindungskabelpaare
 - 1 Drahtbrücke



Der Schaltungsaufbau ist der gleiche wie am vorherigen Tag.

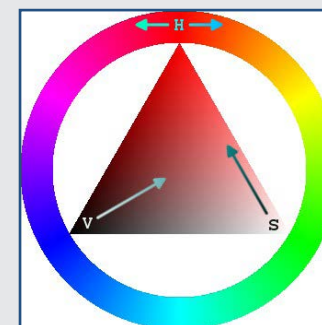
DAS PROGRAMM

Die Hauptschleife des Programms 17-rgbledregen.mblock errechnet nacheinander die RGB-Werte für alle Farbwerte des Farbspektrums. Bei jedem Wert leuchtet die RGB-LED 0,05 Sekunden in der entsprechenden Farbe.



HSV- und RGB-Farbsystem

Das RGB-Farbsystem, das bisher in allen Programmen verwendet wurde, beschreibt Farben als die drei Komponenten Rot, Grün und Blau, die miteinander gemischt werden. Für Menschen ist es relativ schwierig, sich eine Mischfarbe vorzustellen. Im Gegensatz dazu beschreibt das HSV-Farbsystem die Farben über die Werte $H = \text{Hue}$ (Farbton, Farbwinkel), $S = \text{Saturation}$ (Sättigung) und $V = \text{Value}$ (Helligkeitswert). Durch eine einfache Veränderung des H-Werts können alle Farben des Farbspektrums in voller Intensität beschrieben werden, wenn man die beiden anderen Werte auf Maximum einstellt. Eine Darstellung des Farbraums durch das HSV-Modell für Farbwähler in Computern ist verbreitet. Das Grafikmodul in mBlock 5, mit dem man Kostüme und Hintergründe zeichnen kann, verwendet eine HSV-Palette, um eine bestimmte Farbe mithilfe ihrer HSV-Parameter auszuwählen.



SO FUNKTIONIERT DAS PROGRAMM

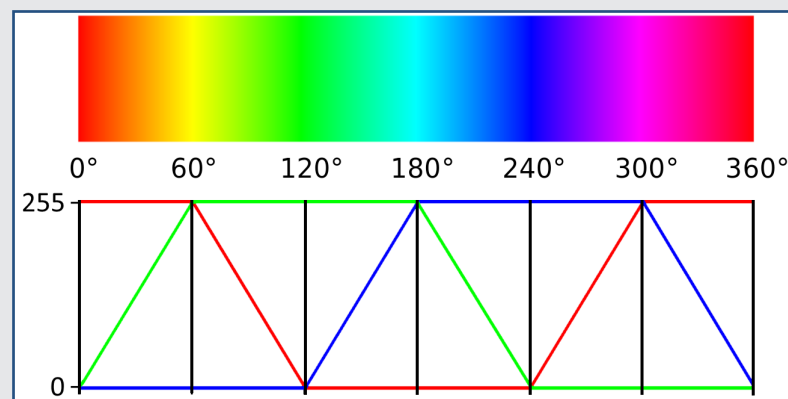
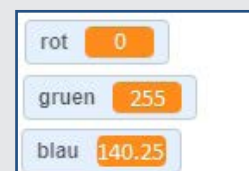
Die drei Variablen **r**, **g** und **b** enthalten die als PWM-Ausgänge definierten Anschlusspins für die RGB-LED. Zusätzlich werden die drei Variablen **rot**, **gruen** und **blau** angelegt, in die während der Berechnung die PWM-Werte für die drei Farbkomponenten der RGB-LED eingetragen werden.

Die Hauptschleife zählt den H-Wert den Gradzahlen entsprechend auf einem Farbkreis zwischen 0 und 360 hoch (in der Abbildung unten als Balken dargestellt) und errechnet daraus die drei Farbkomponenten R, G und B. Die Werte S (Saturation, Sättigung) und V (Value, Helligkeitswert) werden automatisch als Maximalwert gesetzt. Wie die Grafik zeigt, gelten für die sechs 60-Grad-Bereiche jeweils eigene lineare Verlaufskurven. In jedem dieser Bereiche wird ein Farbwert variabel errechnet und in der Variablen **x** gespeichert. Die

anderen beiden werden auf Minimum 0 oder Maximum 255 gesetzt.

Nachdem die RGB-Werte errechnet und in den Variablen **rot**, **gruen**, **blau** gespeichert wurden, werden die PWM-Pins auf die jeweiligen Farbwerte gesetzt. Nach einer kurzen Wartezeit wird der H-Wert um 1 erhöht. Nach einem vollen Durchlauf über 360 Grad beginnt der Zyklus von Neuem bei 0.

Die Farbmischung lässt sich auch sehr gut in den Variablenfeldern auf der Bühne verfolgen.



So wird der H-Wert einer HSV-Farbe in RGB-Werte umgerechnet.

KONSTRUIEREN MIT 3D EASY – DER WEIHNACHTSSTERN, TEIL 2

Heute arbeiten Sie an Ihrem Stern weiter und erstellen seine Zacken. Lesen Sie dazu die Anleitung zu Tag 17.

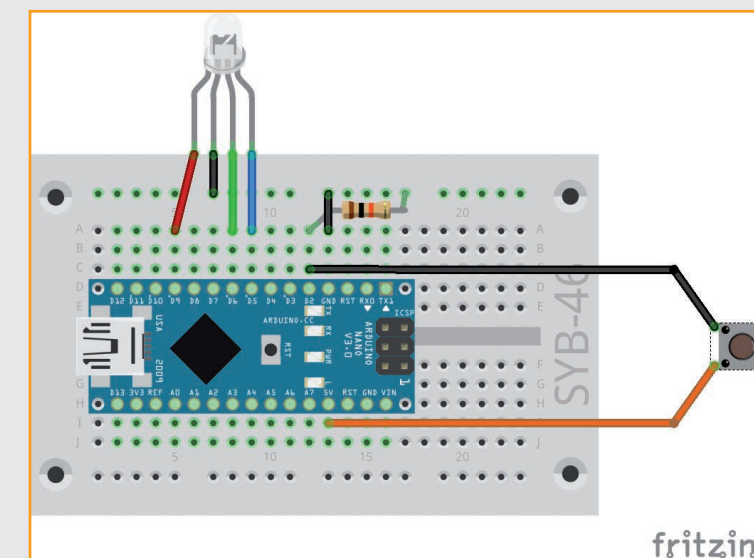
Tag 18

RGB-LED MIT TASTER

Im Kalender: Verbindungskabelpaar

AUFGABE

Der Farbwechsel auf der RGB-LED soll durch Drücken der Taste eingeschaltet werden. Beim nächsten Drücken schaltet sich die RGB-LED wieder aus.



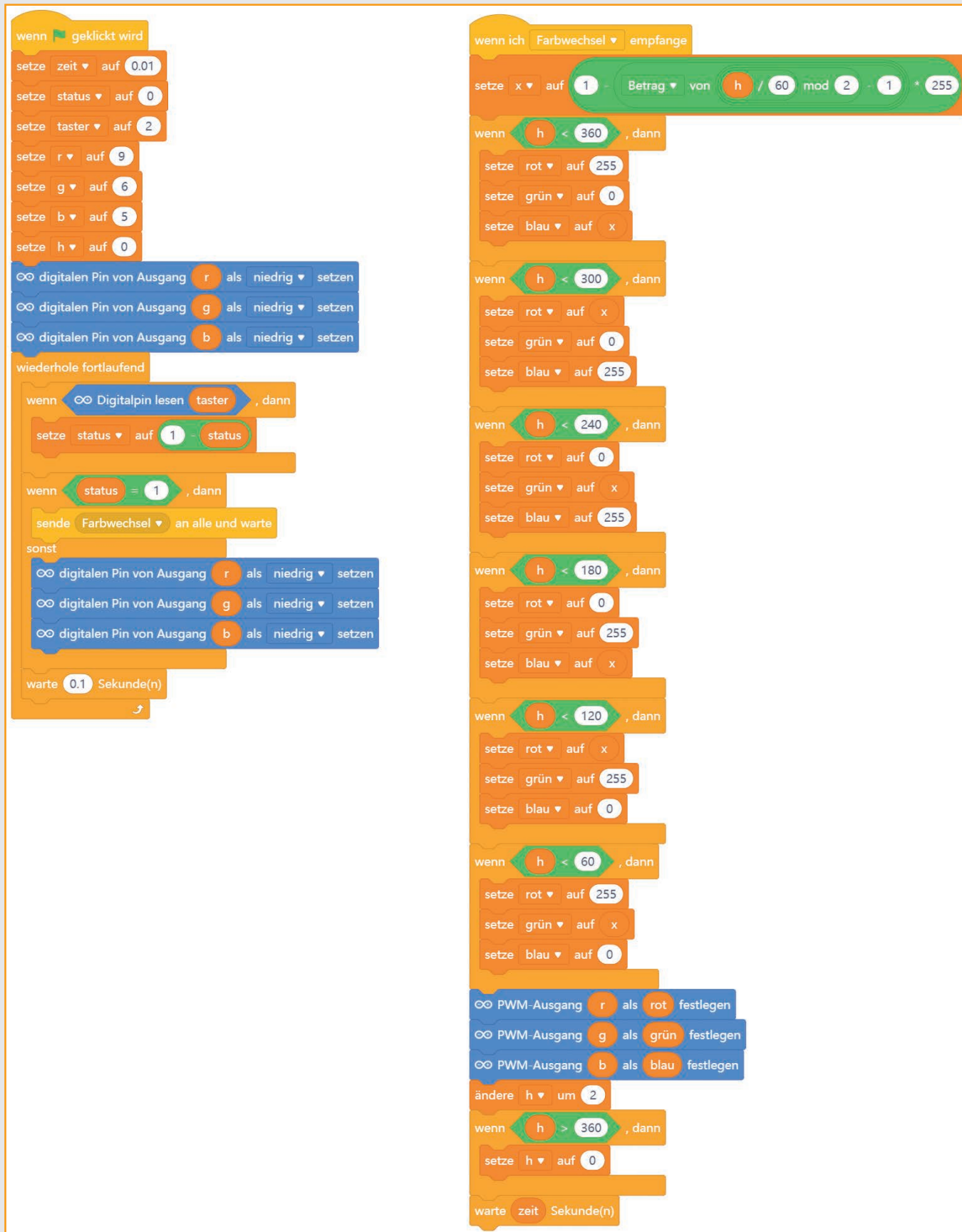
Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 1 RGB-LED mit Vorwiderständen
- 1 Taster
- 1 10-kOhm-Widerstand (braun – schwarz – orange)
- 3 Verbindungskabelpaare
- 1 Drahtbrücke

DAS PROGRAMM

Der Farbwechsel im Programm 18-rgbledtaster.mblock läuft nach dem gleichen Muster wie im Programm des 17. Tags. Das Programm besteht aus zwei voneinander unabhängigen Programmblöcken.

Der Block mit dem Farbwechsel wird aus dem Hauptprogramm heraus über eine Nachricht **Farbwechsel** mit dem Block **sende ... an alle und warte** aufgerufen. Auf diese Weise lassen sich in mBlock Unterprogramme bauen, die komplett abgearbeitet werden, bevor das Hauptprogramm weiterläuft.



SO FUNKTIONIERT DAS PROGRAMM

Die neue Variable **taster** legt Pin D2 als digitalen Eingang für den Taster fest. Die Variable **status** beschreibt, ob der Farbwechsel gerade läuft oder die RGB-LED ausgeschaltet ist.

Der Taster wird ähnlich wie im Programm des 12. Tags in einer Endlosschleife abgefragt. Wenn der Taster gedrückt ist, wird die Variable **status** auf den jeweils anderen Wert – 0 oder 1 – gesetzt. Der Block **setze status auf 1 – status** macht aus einer 0 eine 1 und aus einer 1 eine 0.

Anschließend prüft ein **wenn ... dann ... sonst**-Block den Wert der Variablen **status**.

Ist **status** = 1, sendet ein Block **sende ... an alle und warte** von der Blockpalette **Ereignisse** die Nachricht **Farbwechsel**.

Der zweite Programmblock beginnt mit einem Block **wenn ich Farbwechsel empfangen**. Danach wird der bereits bekannte Farbwechsel einmal gestartet, wiederholt sich aber nicht automatisch.

Anschließend läuft das Hauptprogramm weiter und fragt den Taster erneut ab. Solange er nicht gedrückt wird, bleibt die Variable **status** auf 1, und der nächste Farbwechsel folgt.

Wird der Taster wieder gedrückt, wird die Variable **status** diesmal auf 0 gesetzt. Daraufhin wird der Teil **sonst ...** aus der Abfrage abgearbeitet. Hier werden alle drei Farben der RGB-LED ausgeschaltet. Beim nächsten Drücken der Taste wird der Farbwechsel wieder gestartet.

KONSTRUIEREN MIT 3D EASY – DER WEIHNACHTSSTERN, ABSCHLUSS

An den Tagen zuvor haben Sie bereits die Basis und die Zacken Ihres Sterns erstellt, heute stellen Sie Ihren Stern fertig. Lesen Sie dazu die Anleitung zu Tag 18.



NANO IM ARDUINO-MODUS

● Im Kalender: Verbindungskabelpaar

NANO MIT BATTERIE BETREIBEN

Das Nano-Board kann auch ohne PC genutzt werden und ein gespeichertes Programm abarbeiten. Dazu benötigt es eine externe Stromversorgung. Das kann ein USB-Handyladegerät, eine Powerbank oder auch eine Batterie sein. Wir verwenden den Batteriekasten vom 2. Tag aus dem Adventskalender.

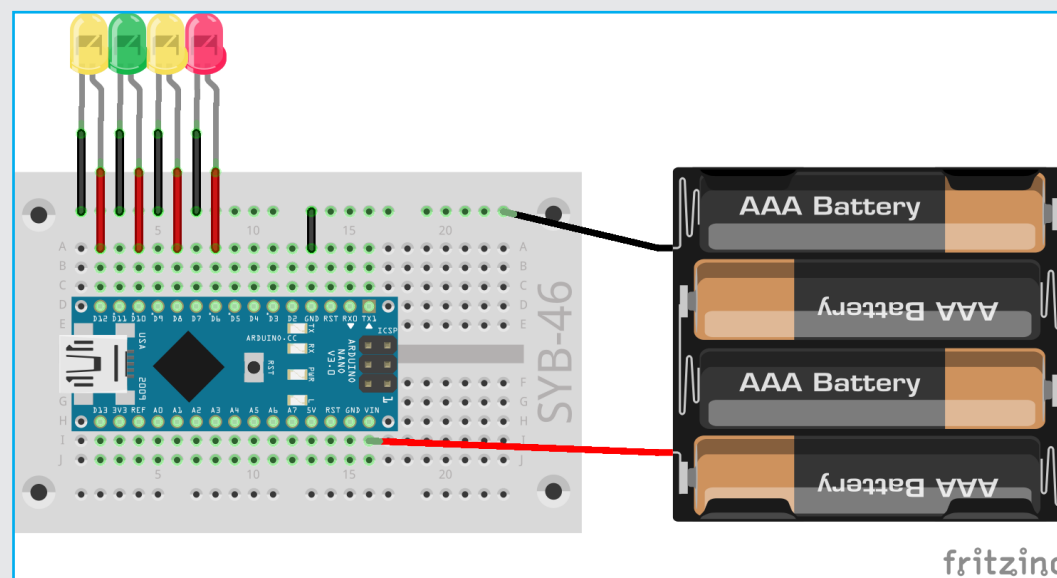
Schließen Sie den Batteriekasten erst einmal nicht an, da der Nano so lange noch über den PC mit Strom versorgt wird, bis das neue Programm übertragen ist. Der Batteriekasten wird nach dem Trennen des USB-Kabels mit den Pins VIN und GND des Nano-Boards verbunden.

AUFGABE

Vier LEDs (z. B. auf dem Weihnachtsbaum) sollen gleichzeitig blinken.

Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 2 LEDs gelb mit Vorwiderständen
- 1 LED rot mit Vorwiderstand
- 1 LED grün mit Vorwiderstand
- 1 Taster
- 4 Verbindungskabelpaare
- 1 Drahtbrücke
- 1 Batteriefach mit 4 AAA-Batterien



DAS PROGRAMM

Das Programm 19a-blink.mblock läuft im sogenannten Arduino-Modus ohne Verbindung mit dem PC. Schalten Sie dazu unten im Bereich **Geräte** von **Live** auf **Hochladen** um. Jetzt können Sie das fertige Programm mit einem Klick auf **Hochladen** auf den Nano übertragen. Es läuft dort eigenständig ohne USB-Verbindung zum PC.



Nach dem erfolgreichen Hochladen trennen Sie den Nano vom Computer und schließen das Batteriefach wie weiter oben beschrieben an.

Wenn Sie es später laufen lassen, werden Sie merken, dass der Arduino-Modus deutlich schneller ist als der interaktive Modus von mBlock 5. Es sind keine Verzögerungen zwischen den LEDs zu erkennen.



SO FUNKTIONIERT DAS PROGRAMM

Jedes Programm im Arduino-Modus beginnt mit dem Block **wenn Arduino startet**, das heißt, wenn Sie den Taster auf dem Nano betätigen oder die Stromversorgung neu anschließen. Blöcke, die Objekte auf der Bühne, Tasten der PC-Tastatur oder Mauseingaben ver-

wenden, können wie Schieberegler im Arduino-Modus nicht genutzt werden. Die weiteren Blöcke in diesem Programm sind bereits von den früheren Tagen bekannt.

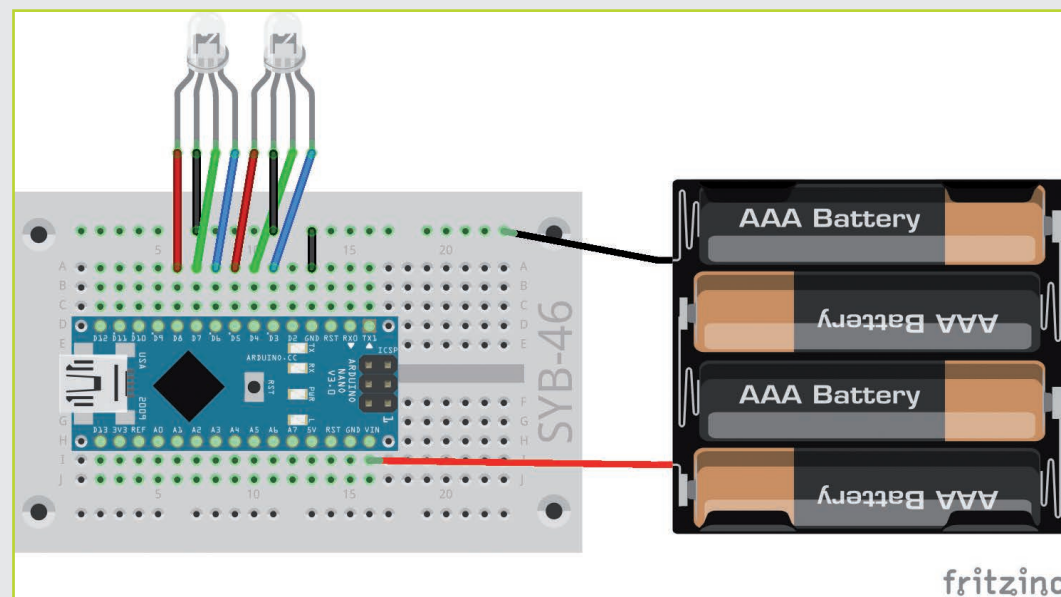
RGB-LEDs IM ZUFALLSMODUS

● Im Kalender: RGB-LED mit Vorwiderstand

AUFGABE

Zwei RGB-LEDs sollen in zufälligen Farben blinken.

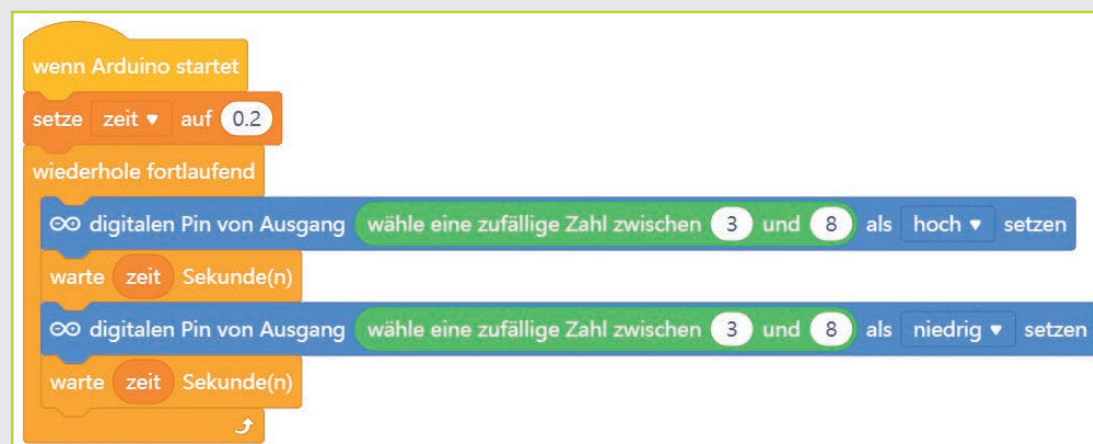
Bauteile
 1 Nano-Board
 1 Steckbrett
 2 RGB-LEDs mit Vorwiderständen
 1 Taster
 4 Verbindungskabelpaare
 1 Drahtbrücke
 1 Batteriefach mit 4 AAA-Batterien



DAS PROGRAMM

Das Programm 20a-2rgbled.mblock zeigt in einer Endlosschleife zufällige Farben auf den RGB-LEDs. Es läuft im Arduino-Modus. Das Programm kann ohne

Verbindung zum PC mit dem Batteriekasten betrieben werden.



SO FUNKTIONIERT DAS PROGRAMM

Wenn Sie auf den Taster des Nano drücken, wird in jedem Schleifendurchlauf zufällig einer der sechs Pins D3 bis D8, an denen die beiden RGB-LEDs angeschlossen sind, eingeschaltet, und nach einer eingestellten Zeit wird ein zufällig gewählter Pin ausgeschaltet.

Bei dieser Methode kann es passieren, dass sich die angezeigten Farben in einem Schleifendurchlauf nicht ändern oder auch einmal gar keine Farbe leuchtet. Das Blinkmuster erscheint unregelmäßig und zufällig.

KONSTRUIEREN MIT 3D EASY – DER SCHNEEMANN, TEIL 1

Heute starten Sie mit dem letzten 3D-Druck-Projekt, dem Schneemann. Sie beginnen mit seinem Körper. Lesen Sie dazu die Anleitung zu Tag 20.



SUMMER UND LEDS LIVE-MODUS

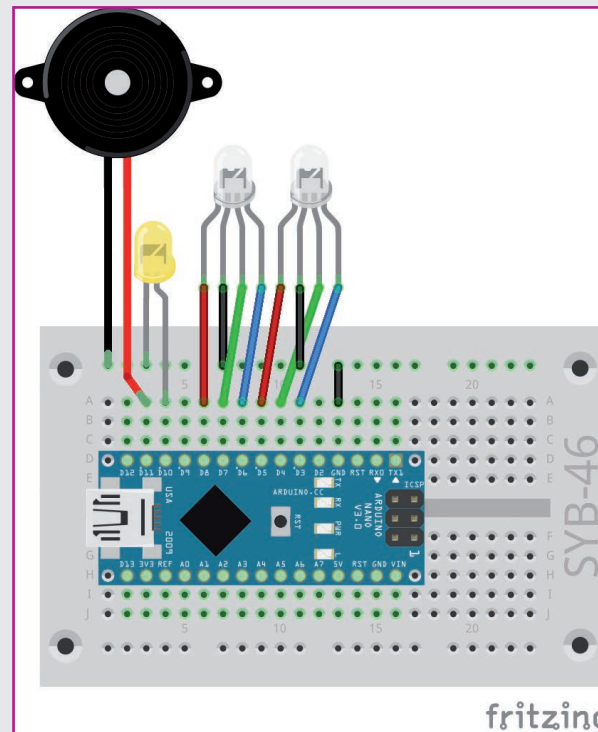
● Im Kalender: Taster

AUFGABE

Wenn Sie auf das grüne Fähnchen klicken, geht im Haus das Licht an, der Summer ertönt, und nach kurzer Zeit blinken die RGB-LEDs bunt. Einige Sekunden später gehen alle LEDs aus.

Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 2 RGB-LEDs mit Vorwiderständen
- 1 LED gelb mit Vorwiderstand
- 4 Verbindungskabelpaare
- 1 Drahtbrücke
- 1 Piezo-Summer



```

wenn geklickt wird
  setze led auf 10
  setze n auf 4
  wiederhole 6
    digitalen Pin von Ausgang n als niedrig setzen
    ändere n um 1
  digitalen Pin von Ausgang led als hoch setzen
  setze summer auf 11
  
```

```

sende ton1 an alle
warte pause / 10 Sekunde(n)
sende aus an alle
warte pause / 10 Sekunde(n)
sende ton1 an alle
warte pause / 10 Sekunde(n)
sende aus an alle
warte pause / 10 Sekunde(n)
sende ton2 an alle
warte pause / 10 Sekunde(n)
sende aus an alle
warte pause / 10 Sekunde(n)
sende ton1 an alle
warte pause / 10 Sekunde(n)
sende aus an alle
warte pause / 10 Sekunde(n)
  
```

```

setze blinkzeit auf 1
setze w auf 16
wiederhole w
  digitalen Pin von Ausgang wähle eine zufällige Zahl zwischen 4 und 6 als hoch setzen
  digitalen Pin von Ausgang wähle eine zufällige Zahl zwischen 7 und 9 als hoch setzen
  warte blinkzeit / 10 Sekunde(n)
  digitalen Pin von Ausgang wähle eine zufällige Zahl zwischen 4 und 6 als niedrig setzen
  digitalen Pin von Ausgang wähle eine zufällige Zahl zwischen 7 und 9 als niedrig setzen
  warte blinkzeit / 10 Sekunde(n)
setze n auf 4
wiederhole 6
  digitalen Pin von Ausgang n als niedrig setzen
  ändere n um 1
digitalen Pin von Ausgang led als niedrig setzen
  
```

DAS PROGRAMM

Das Programm 21-haus01.mblock besteht im Wesentlichen aus bekannten Blöcken, die beim Klick auf das grüne Fähnchen nacheinander abgearbeitet werden. Das Programm läuft wieder im Live-Modus von mBlock 5.

Firmware für mBlock 5 neu installieren

Wenn Sie zwischendurch ein Programm im Arduino-Modus von mBlock 5 oder ein anderes Arduino-Programm verwendet haben, müssen Sie, bevor Sie mBlock 5 wieder interaktiv benutzen können, die Firmware erneut installieren. Nachdem der Nano in mBlock 5 verbunden wurde, erscheint unten ein orangefarbener Button **Aktualisieren**. Klicken Sie darauf und dann auf **Firmware aktualisieren**. Klicken Sie im nächsten Fenster auf **Aktualisierungen**, wird die Firmware heruntergeladen und installiert. Anschließend verbinden Sie den Nano neu.

```

wenn ich ton1 empfangen
  PWM-Ausgang summer als 6000 festlegen

wenn ich ton2 empfangen
  PWM-Ausgang summer als 500 festlegen

wenn ich aus empfangen
  PWM-Ausgang summer als 0 festlegen
  
```

SO FUNKTIONIERT DAS PROGRAMM

Das Programm verwendet elf Variablen:

- **blinkzeit** – Legt in Zehntelsekunden fest, wie lange die RGB-LEDs beim Blinken ein- oder ausgeschaltet sein sollen. Über einen Schieberegler von 1 bis 10 einstellbar.
- **led** – Pinnummer der LED im Haus.
- **n** – Zähler für eine Schleife (beginnt bei dem ersten Pin der ersten RGB-LED).
- **pause** – Pause zwischen zwei Tönen und Länge eines Tons der Tonfolge in Zehntelsekunden. Über einen Schieberegler von 1 bis 10 einstellbar.
- **summer** – Pinnummer des Summers.
- **ton1** bis **ton4** – Tonwerte für die vier Töne, werden im Programm mit 100 multipliziert, um hörbare PWM-Werte zu erhalten. Über Schieberegler von 1 bis 10 einstellbar.
- **w** – Anzahl der Wiederholungen beim Blinken. Über einen Schieberegler von 1 bis 10 einstellbar.
- **zeit** – Wartezeit zwischen Summertönen und dem LED-Blinken.



Die Regler auf der Bühne werden nur bei den Variablen angezeigt, bei denen das blaue Häkchen gesetzt ist. Sie können die Einstellungen jederzeit ändern.

1. Nach Klick auf das grüne Fähnchen werden die RGB-LEDs ausgeschaltet (eine Schleife zählt beginnend beim Pin der ersten RGB-LED bis zum letzten RGB-LED-Pin durch, also sechsmal), und die einzelne LED im Haus wird eingeschaltet.
2. Die vier Töne werden über PWM-Signale erzeugt, wobei die vier Tonwerte über Schieberegler interaktiv eingestellt werden können.
3. Nach einer einstellbaren Wartezeit beginnt eine Schleife, die die RGB-LEDs blinken lässt. Die Anzahl der Wiederholungen ist über die Variable **w** einstellbar. In jedem Durchlauf der Schleife wird auf beiden RGB-LEDs je eine zufällig gewählte Farbe eingeschaltet. Nach der eingestellten Blinkzeit wird wieder je eine zufällig gewählte Farbe auf den beiden RGB-LEDs ausgeschaltet. Auf diese Weise können verschiedene Mischfarben entstehen. Es ist aber auch möglich, dass sich die Farbe einer LED in einem Schleifendurchlauf nicht verändert oder eine LED ganz ausgeht.
4. Nach der festgelegten Zahl von Wiederholungen schaltet eine weitere Schleife alle sechs Farben der beiden RGB-LEDs aus. Danach wird auch das Licht im Haus ausgeschaltet.

KONSTRUIEREN MIT 3D EASY – DER SCHNEEMANN, TEIL 2

Gestern haben Sie bereits den Körper des Schneemanns erstellt. Heute geht es mit seinem Kopf weiter. Lesen Sie dazu die Anleitung zu Tag 21.



WEIHNACHTSLIED AUF DEM PC-LAUTSPRECHER

- Im Kalender: Widerstand 10 kOhm (braun – schwarz – orange)

AUFGABE

Die heutige Aufgabe ist der von gestern sehr ähnlich. Allerdings sollen das Licht im Haus und das Blinken der RGB-LEDs nicht durch einen Klick auf das grüne Fähnchen ausgelöst werden, sondern durch Drücken des Klingeltasters an der Haustür. Ein zweiter Taster lässt ein Weihnachtslied ertönen. Dazu verwenden wir diesmal statt des Piezo-Summers die Lautsprecher am PC, die eine deutlich bessere Tonqualität bieten.

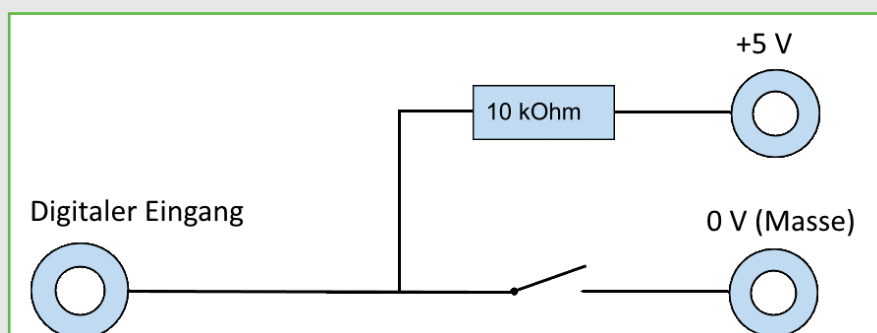
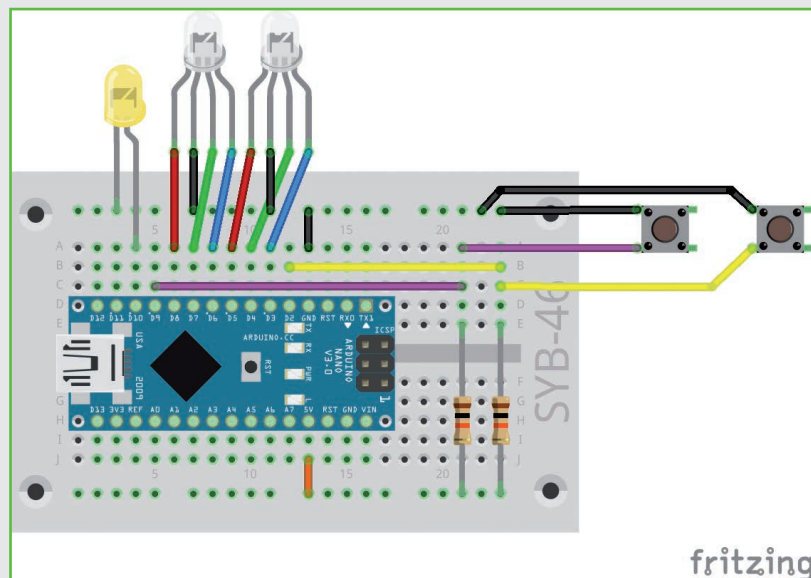
Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 2 RGB-LEDs mit Vorwiderständen
- 1 LED mit Vorwiderstand
- 2 Taster
- 2 10-kOhm-Widerstände (braun – schwarz – orange)
- 6 Verbindungskabelpaare
- 4 Drahtbrücken

Taster mit Pull-up-Widerstand

Vielleicht ist Ihnen bei den letzten Experimenten mit dem Taster aufgefallen, dass manchmal etwas im Programm passiert, ohne dass der Taster gedrückt wurde. Der Taster scheint sich selbst auszulösen.

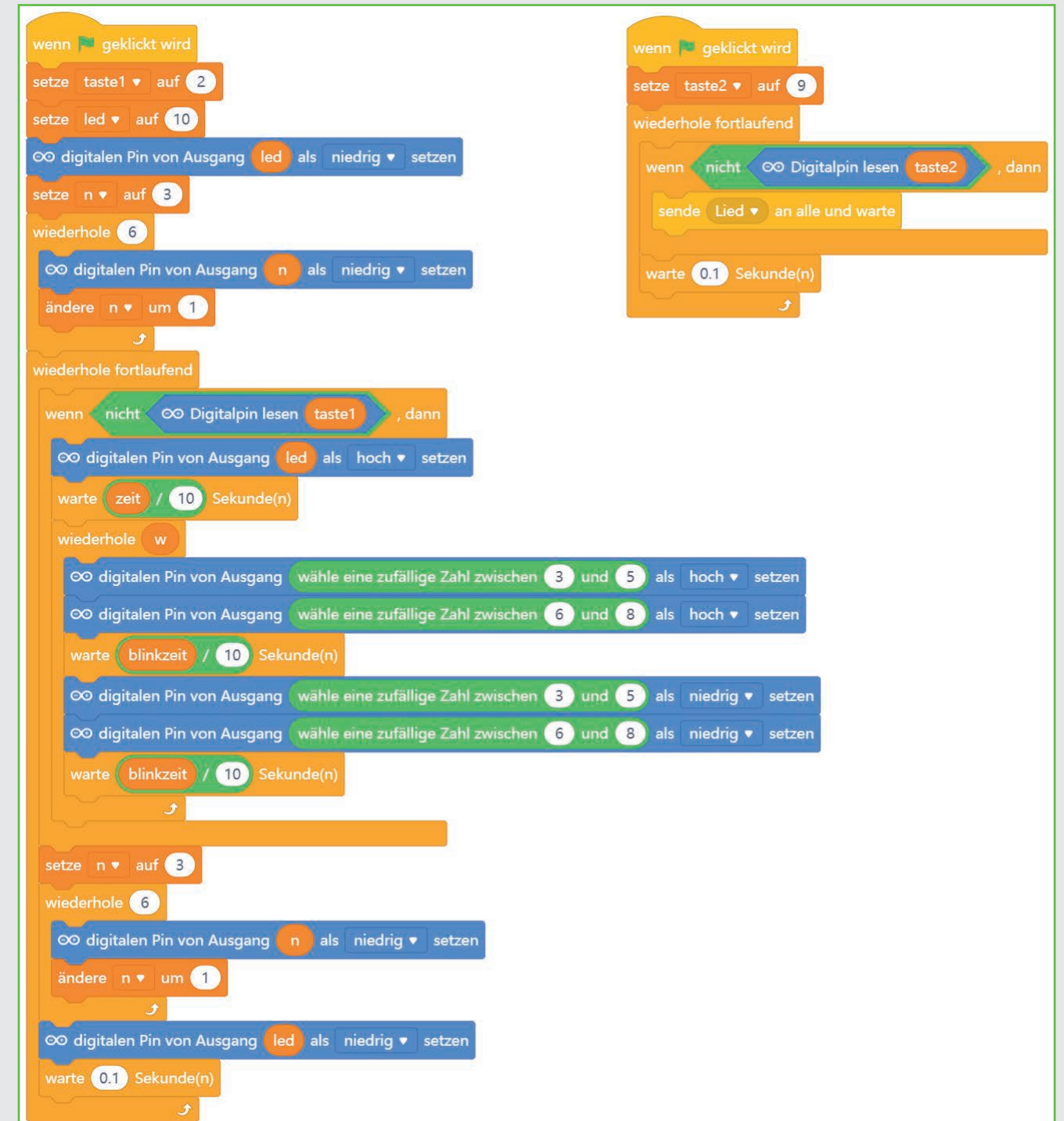
Im Experiment heute nutzen wir statt des bisher verwendeten Pull-down-Widerstands einen Pull-up-Widerstand. Er zieht den Status des Pins an einem nicht gedrückten Taster nach oben auf *High*. Am digitalen Eingangspin kommt also immer ein *High*-Signal an. Der Taster muss beim Drücken mit der Masseleitung verbunden werden und liefert dann ein *Low*-Signal. Das Programm muss also umgekehrt fragen, ob der Pin *Low* und nicht *High* ist. Das bezeichnet man als inverse Logik.



Taster mit Pull-up-Widerstand. Wird der Taster gedrückt, wird der digitale Eingang mit dem Minuspol (Masse) verbunden.

DAS PROGRAMM

Der wichtigste Unterschied des Programms 22-haus02.mblock gegenüber bisherigen Programmen mit einem Taster sind die Endlosschleifen, die den Zustand der Taster mit inverser Logik abfragen.



SO FUNKTIONIERT DAS PROGRAMM

Beim Klick auf das grüne Fähnchen starten zwei unabhängige Programmblöcke. Der erste bringt die gesamte Schaltung in eine Grundstellung: Licht im Haus aus und RGB-LEDs dunkel. In beiden Programmblöcken beginnt je eine Endlosschleife, die jede Zehntelsekunde einen der Taster abfragt. Wegen des Pull-up-Widerstands wird hier inverse Logik gebraucht. Die **wenn ... dann**-Abfrage muss prüfen, ob an dem Pin, der durch die Variable **taste1** oder **taste2** festgelegt ist, ein **Low**-Signal liegt. Der Block **Digitalpin lesen ...** liefert **wahr**, wenn der Pin **High** ist. Dieser Block ist in einen Block **nicht ...** eingebaut, der den Logikwert genau umkehrt. Die **wenn ... dann**-Abfrage erhält also ein **wahr**, wenn der Taster gedrückt ist und damit der Pin ein **Low**-Signal hat.

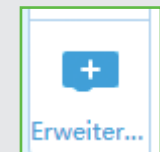
Wenn Taste 1 gedrückt wurde, beginnt die bekannte Szene: Das Licht im Haus geht an, und die RGB-LEDs auf dem Dach blinken bunt. Nach einer bestimmten Zeit gehen die RGB-LEDs und das Licht im Haus wieder aus. Die Zeiten und die Anzahl der Wiederholungen beim

Blinken der RGB-LEDs lassen sich wieder über Schieberegler einstellen. Am Ende läuft die Endlosschleife weiter und wartet erneut darauf, dass die Taste 1 gedrückt wird.

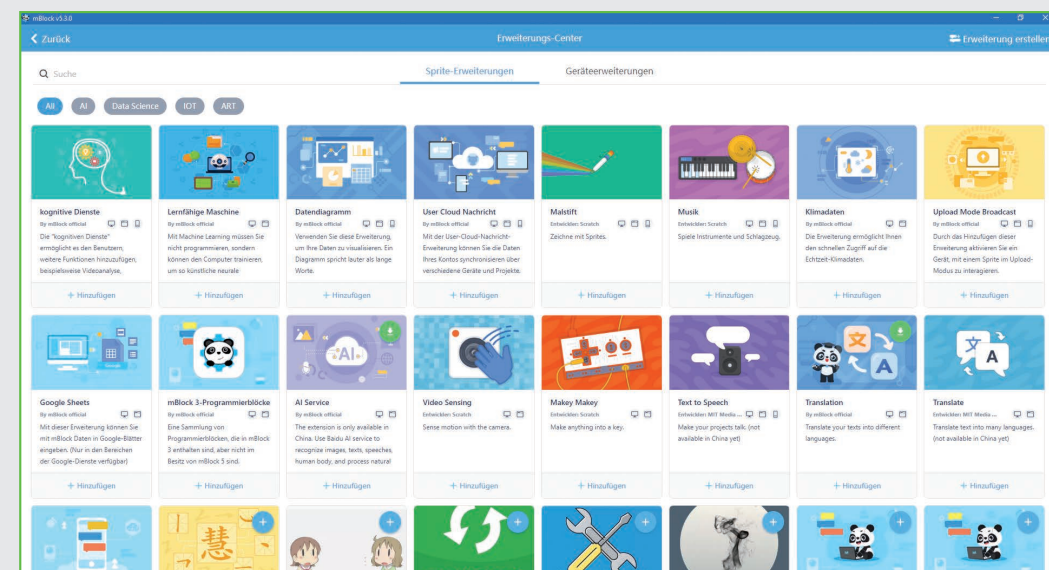
Der andere Programmblock wartet darauf, dass der zweite Taster gedrückt wird, und spielt dann ein Lied ab.

Musik in mBlock 5

mBlock 5 kann wie auch Scratch in seinem Funktionsumfang erweitert werden. Dazu wird eine Vielzahl an Erweiterungen unterschiedlicher Entwickler angeboten. Eine solche Erweiterung spielt beliebige Töne auf dem PC-Lautsprecher ab.



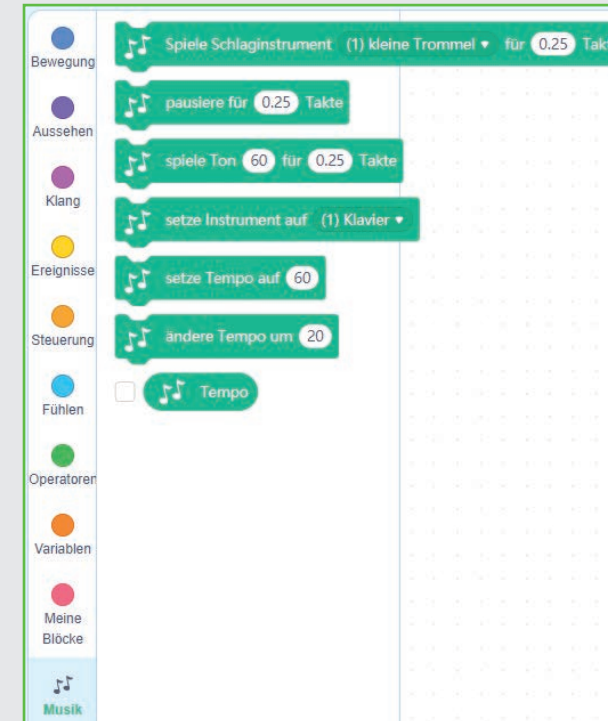
Klicken Sie unterhalb der Blockpalette auf das Symbol **Erweiterungen** und wählen Sie die Erweiterung **Musik** auf der Seite **Sprite-Erweiterungen** aus.



KONSTRUIEREN MIT 3D-EASY – DER SCHNEEMANN TEIL 3

Heute bekommt der Schneemann seinen Zylinder. Lesen Sie dazu die Anleitung zu Tag 22. Damit haben Sie das letzte 3D-Projekt abgeschlossen und haben bis Heiligabend noch genügend Zeit, die 3D-Drucke anzufertigen.

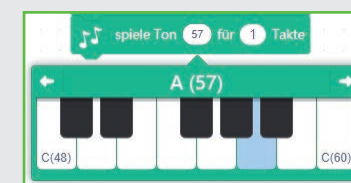
In der Blockpalette erscheint ganz unten ein neuer Bereich **Musik** mit Blöcken zur Tonerzeugung.



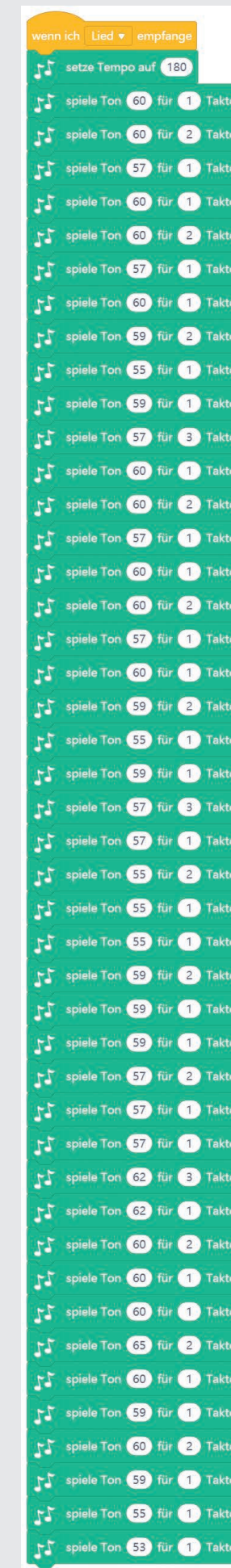
Da die Musikerweiterung nur mit Figuren funktioniert, sendet die Abfrage von Taste 2 eine Nachricht **Lied**, die dann von der Figur im Programm ausgewertet wird.

Dieses Programm spielt ein Weihnachtslied ab. Dazu wird als Erstes mit einem Block **setze Tempo auf ...** aus der Erweiterung **Musik** das Tempo definiert. Diese Erweiterung gibt die Tonlängen nicht in Millisekunden, sondern in sogenannten Takten an.

Anschließend spielen **spiele Ton ... für ... Takte**-Blöcke nacheinander die einzelnen Töne des Lieds ab. Klicken Sie in diesem Block auf das Zahlenfeld hinter **Ton**, kann über eine Klaviertastatur der gewünschte Ton ausgewählt oder als Zahlenwert eingetragen werden. Im zweiten Feld dieser Blöcke wird die Tonlänge in Takten angegeben.



Diese Töne werden über den Lautsprecher am PC ausgegeben, nicht über einen Piezo-Summer. Deshalb funktionieren diese Blöcke auch im Arduino-Modus von mBlock 5 nicht.



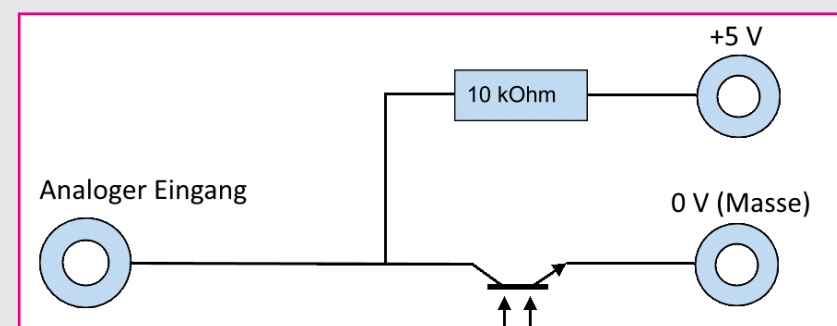
DER DÄMMERUNGSSCHALTER

Im Kalender: Fototransistor

FOTOTRANSISTOR

Ein Fototransistor ist ein lichtempfindliches Bauelement, das auf den ersten Blick wie eine transparente LED aussieht. Je nach Stärke des Lichteinfalls lassen sich mit der abgebildeten Schaltung an einem analogen Eingang des Nano-Boards unterschiedliche Werte erzielen. Je mehr Licht auf den Fototransistor fällt, desto geringer wird der Wert am analogen Eingang. Im Gegensatz

zu LEDs wird bei Fototransistoren der lange Anschluss mit Masse verbunden, nicht der kurze. Die analogen Eingänge A0 bis A7 des Nano werten einen analogen Spannungswert aus und liefern digitale Werte zwischen 0 und 1023. Dabei stehen 0 für 0 V und 1023 für +5 V Spannung am jeweiligen Pin. Im Gegensatz dazu kann ein digitaler Eingang nur die Werte 0 oder 1 liefern.



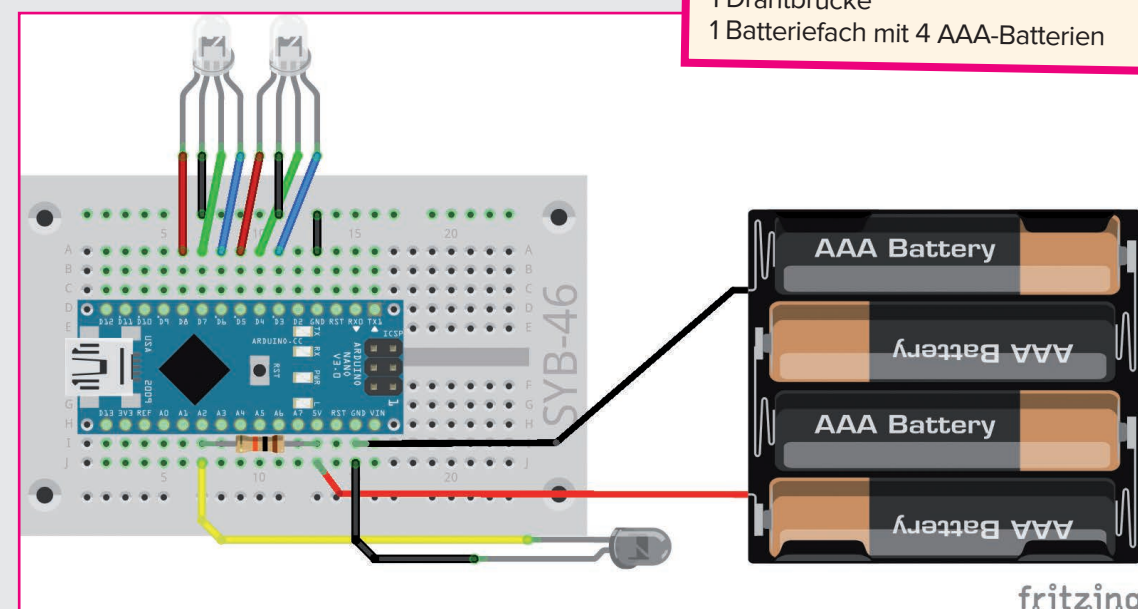
Schaltschema für einen Fototransistor.

AUFGABE

Die beiden RGB-LEDs sollen bunt blinken, wenn es dunkel genug ist.

Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 2 RGB-LEDs mit Vorwiderständen
- 1 Fototransistor
- 1 10-kOhm-Widerstand (braun – schwarz – orange)
- 5 Verbindungskabelpaare
- 1 Drahtbrücke
- 1 Batteriefach mit 4 AAA-Batterien



WEIHNACHTLICHE LICHTEFFEKTE, WENN ES DUNKEL WIRD

Das Programm 23a-blink.mblock läuft eigenständig über den Arduino-Modus von mBlock 5. Schließen Sie nach der Übertragung auf den Nano noch den Batteriekasten an und dunkeln Sie den Fototransistor ab. Dann beginnen die LEDs zu blinken.



SO FUNKTIONIERT DAS PROGRAMM

Der Fototransistor liefert je nach Helligkeit einen analogen Wert zwischen 0 und 1023, der umso höher ist, je weniger Licht auf den Fototransistor fällt. Der Block **analogen Pin (A) ... lesen** liest den analogen Wert an Pin A2. Ist dieser Wert größer als 800, ist es dem Programm dunkel genug, um die LEDs blinken zu lassen. Natürlich können Sie diesen Grenzwert entsprechend der Umgebung, in der Sie die Schaltung verwenden, ändern.

Diesmal nutzen wir zur Abwechslung ein etwas anderes Blinkmuster. Einer der sechs Pins der beiden RGB-LEDs

wird zufällig ausgewählt und blinkt kurz auf. Im nächsten Durchlauf wird wieder ein Pin zufällig gewählt.

Ganz am Anfang steht der Block **wenn Arduino startet**. Das Programm startet nach der Übertragung auf den Nano automatisch und auch jedes Mal, wenn das Nano-Board mit Strom versorgt wird oder man auf den eingebauten Taster drückt. Das grüne Fähnchen aus mBlock 5 kann im Arduino-Modus nicht verwendet werden.

Stellen Sie heute alle 3D-Drucke fertig. Morgen werden Sie die elektronischen Bauteile montieren und das Weihnachtshäuschen fertig stellen.

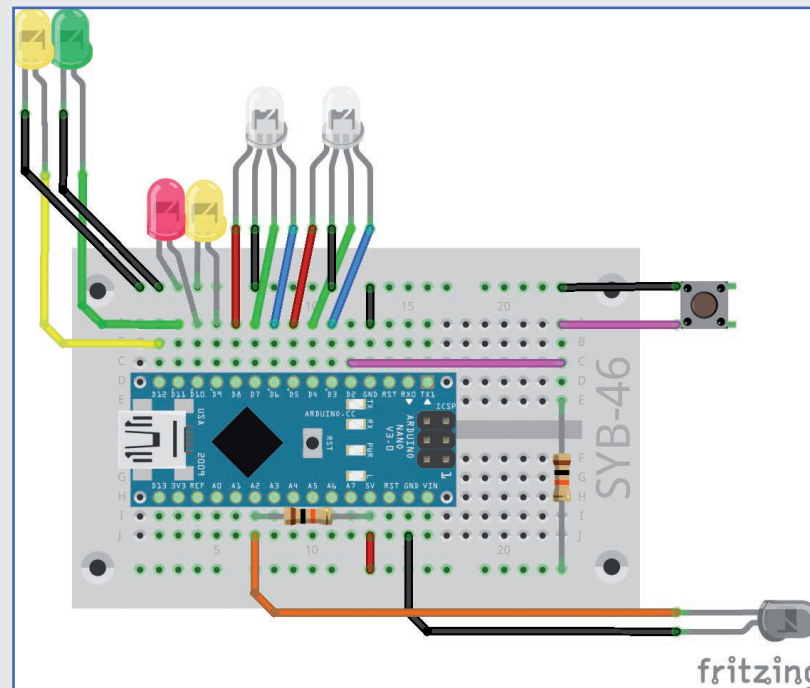
DAS HÄUSCHEN FERTIGSTELLEN

- Im Kalender: Downloadcode für eine kleine Überraschung

AUFGABE

Im Haus leuchtet ein Licht, auf dem Dach und auf dem Weihnachtsbaum blinken LEDs beim Drücken des Klingeltasters und andere auch bei Dunkelheit. Zusätzlich ertönt aus dem PC-Lautsprecher ein Weihnachtslied. Deshalb läuft das Programm im interaktiven Modus von mBlock 5.

Über den Downloadcode im Adventskalender heute gibt es noch weitere Weihnachtslieder im MP3-Format zum Herunterladen.



Bauteile

- 1 Nano-Board
- 1 Steckbrett
- 2 RGB-LEDs mit Vorwiderständen
- 2 LEDs gelb mit Vorwiderständen
- 1 LED grün mit Vorwiderstand
- 1 LED rot mit Vorwiderstand
- 1 Fototransistor
- 2 10-kOhm-Widerstände (braun – schwarz – orange)
- 1 Taster
- 8 Verbindungskabelpaare
- 3 Drahtbrücken

Heute montieren Sie die Elektronikbauteile am Weihnachtshäuschen. Lesen Sie dazu die Anleitung für Projekt 6. Diese Anleitung ist lediglich ein Vorschlag für eine relativ einfache Montage. Ihrem Gestaltungsdrang sind keine Grenzen gesetzt, Sie können die Bauteile auch anders anordnen und neu programmieren. Viel Spaß beim Planen und Ausprobieren!

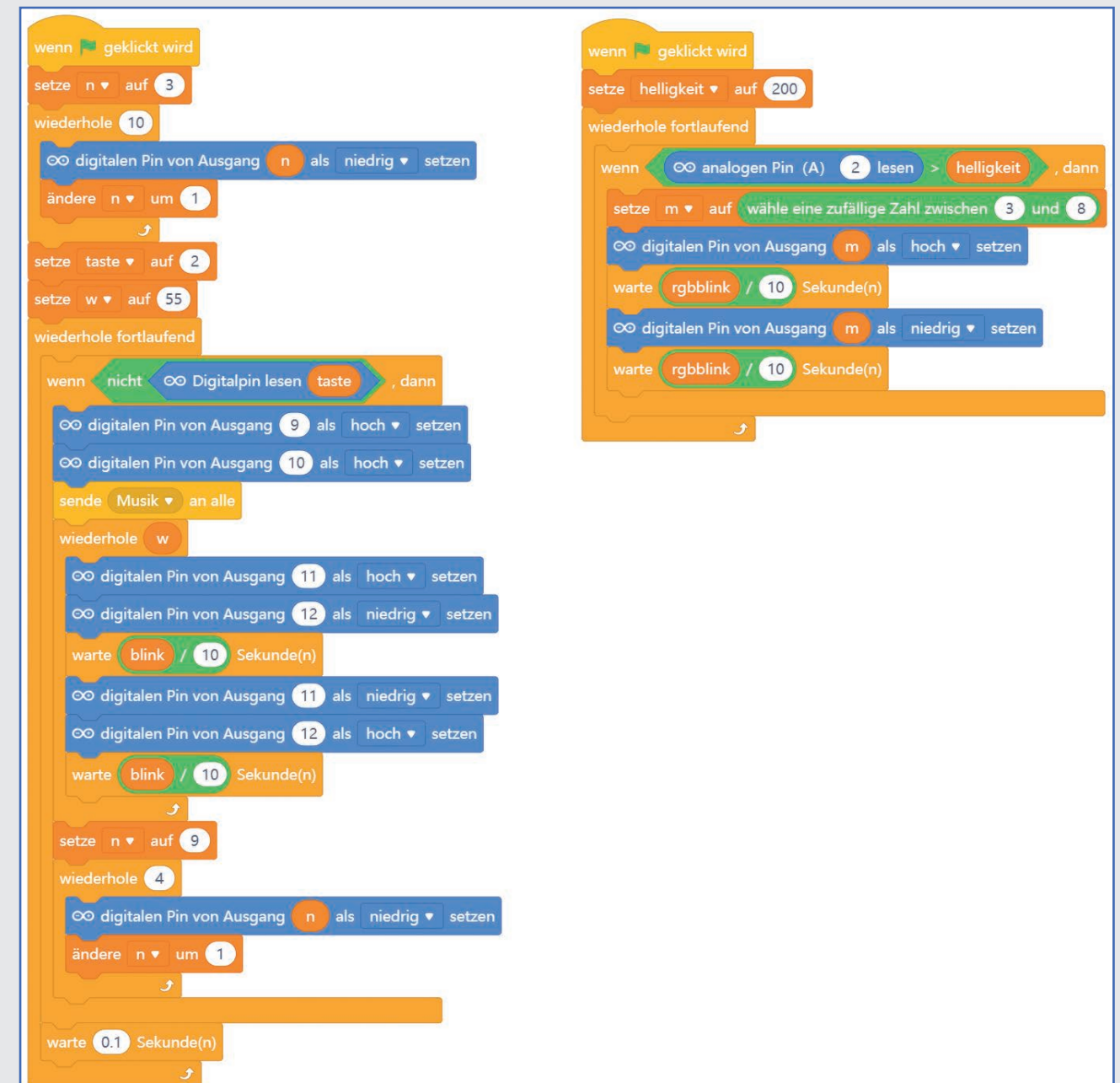
Firmware für mBlock 5 neu installieren

Wenn Sie zwischendurch ein Programm im Arduino-Modus von mBlock 5 oder ein anderes Arduino-Programm verwendet haben, müssen Sie, bevor Sie mBlock 5 wieder interaktiv benutzen können, die Firmware erneut installieren. Nachdem der Nano in mBlock 5 verbunden wurde, erscheint unten ein orangefarbener Button **Aktualisieren**. Klicken Sie darauf und dann auf **Firmware aktualisieren**. Klicken Sie im nächsten Fenster auf **Aktualisierungen**, wird die Firmware heruntergeladen und installiert. Anschließend verbinden Sie den Nano neu.

DAS PROGRAMM

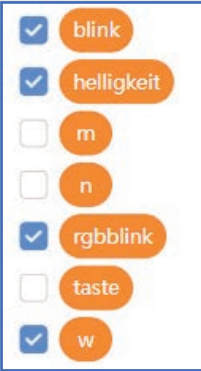
Die Schaltung bringt mehrere aus den letzten Tagen bekannte Elemente zusammen.

Das Programm 24-hausmusik.mblock schaltet beim Drücken des Tasters zwei Lichter im Haus und zwei LEDs auf den Weihnachtsbäumen an. Bei Dunkelheit blinken auch noch die RGB-LEDs auf dem Dach.



SO FUNKTIONIERT DAS PROGRAMM

Das Programm verwendet sieben Variablen:



Helligkeit – Schwellenwert, ab dem es dunkel genug ist, um den Blinkeffekt der RGB-LEDs zu zeigen.

blink – Legt in Zehntelsekunden fest, wie lange die LEDs beim Blinken ein- oder ausgeschaltet sein sollen.

m – Zähler für eine Schleife.

n – Zähler für eine Schleife.

rgbblink – Legt in Zehntelsekunden fest, wie lange die RGB-LEDs beim Blinken ein- oder ausgeschaltet sein sollen.

taste – Pinnummer des Tasters.

w – Anzahl der Wiederholungen beim Blinken.

Die Variablen **w** und **Helligkeit** werden beim Start des Programms auf sinnvolle Werte gesetzt. **blink** und **rgbblink** können, während das Programm läuft, mit Schiebereglern eingestellt werden.

PROGRAMMBLÖCKE FÜR LICHT UND BLINKLICHTER

Die Programmblöcke für das Licht im Haus und das Blinklicht auf dem Weihnachtsbaum sind in ihrer Funktionsweise aus früheren Programmen bekannt.

Beim Klick auf das grüne Fähnchen schaltet der linke Programmblock alle LEDs und RGB-LEDs aus. Danach wartet das Programm darauf, dass der Taster gedrückt wird. Sobald er gedrückt wird, werden die beiden LEDs im Haus an den Pins 9 und 10 eingeschaltet.

Anschließend wird ein Weihnachtslied abgespielt. Da der dazu verwendete Block aus der Blockpalette **Klang**

nur für Figuren zur Verfügung steht, sendet das Programm für Geräte an dieser Stelle eine Nachricht **Musik**.

Während das Lied läuft, blinken die beiden LEDs an den Pins 11 und 12 abwechselnd. Die Geschwindigkeit wird über den Schieberegler **blink** eingestellt. Die Variable **w** für die Anzahl der Wiederholungen ist so eingestellt, dass die LEDs bei einer Blinkgeschwindigkeit von zwei Zehntelsekunden ungefähr mit dem Ende des Lieds aufhören zu blinken. Bei Bedarf können Sie diesen Wert auch im Programm anpassen.

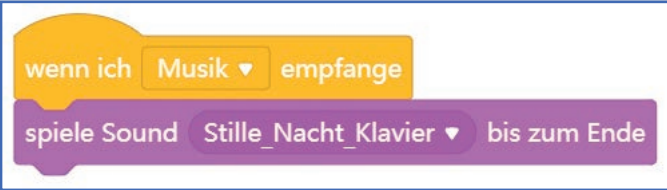
PROGRAMMBLÖCKE FÜR RGB-BLINKLICHTER

Das Programm für den Fototransistor und das Blinklicht entspricht weitgehend dem Programm vom 23. Tag. Die Variable **helligkeit** legt den Schwellenwert fest, ab dem es dunkel genug ist, um den Blinkeffekt zu zeigen. Solange es dunkel genug ist, blinken die beiden RGB-LEDs zufällig.

PROGRAMMBLÖCKE FÜR DAS WEIHNACHTSLIED

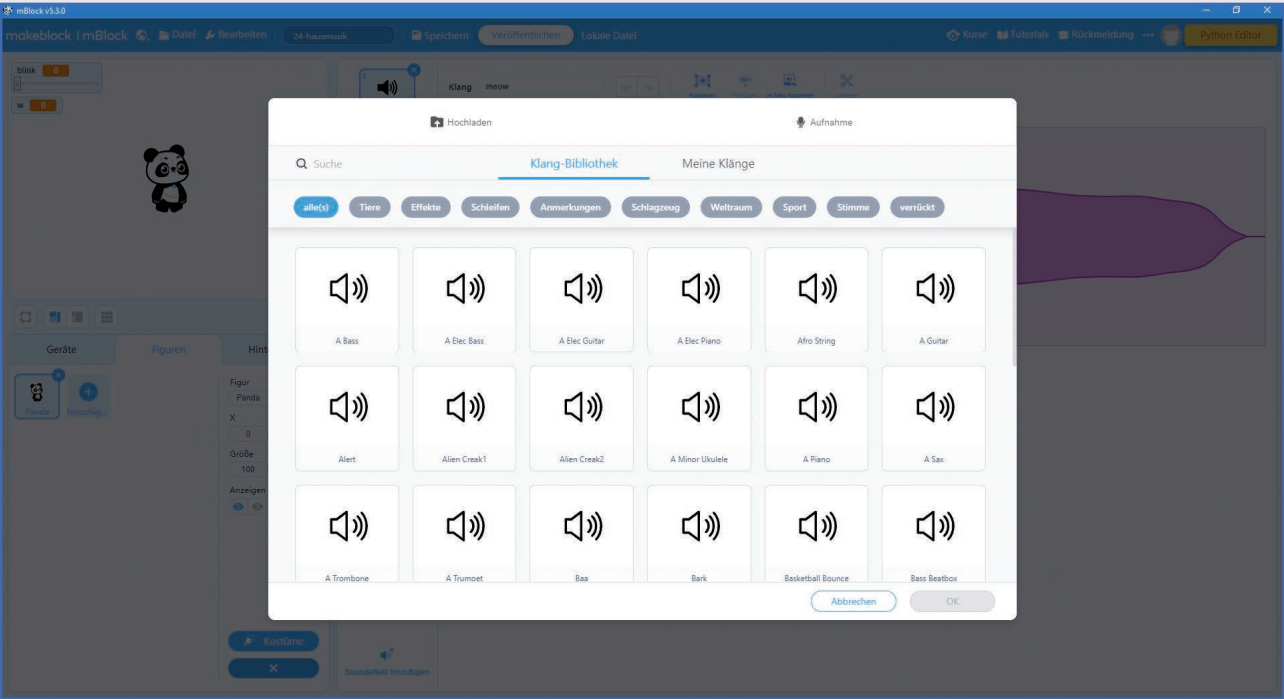
Im Bereich **Figuren** wird ein Weihnachtslied auf den PC-Lautsprechern abgespielt, wenn die Nachricht **Musik** empfangen wird.

Programm enthalten. Löschen Sie ihn, wie Sie auch ein Kostüm löschen würden, mit einem Klick auf das blaue x-Symbol bei dem Namen des Klangs.

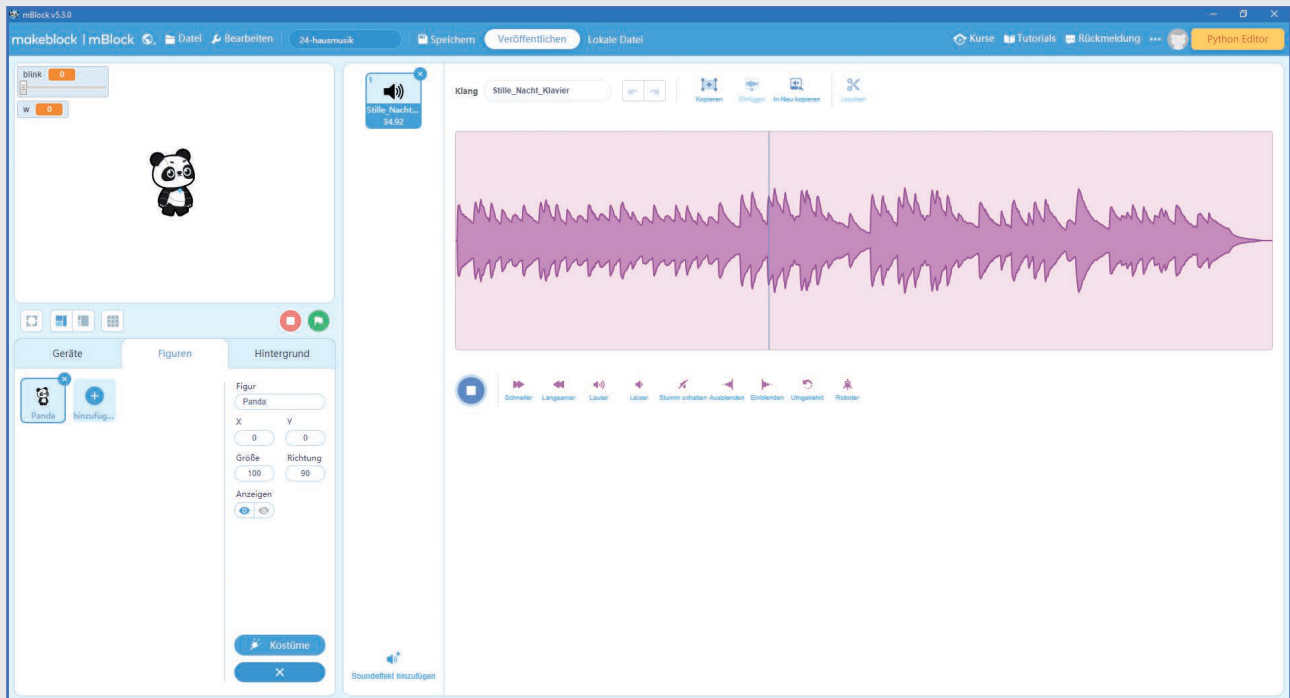


Importieren Sie jetzt über das Symbol **Soundeffekt hinzufügen** ein Weihnachtslied aus dem Download, den Sie über den Downloadcode hinter dem heutigen Türchen des Adventskalenders erhalten. Klicken Sie dazu im nächsten Fenster auf **Hochladen** und wählen Sie das Lied aus.

mBlock 5 kann auf dem PC-Lautsprecher MP3-Dateien abspielen, die in das Programm importiert wurden. Klicken Sie dazu im Bereich **Figuren** unten auf **Klänge**. In dieser Ansicht sind die Klänge im Programm aufgelistet. Standardmäßig ist ein einfacher Klang in jedem neuen



Der importierte Klang erscheint als Wellenformgrafik.
Mit einem Klick auf das blaue Abspielen-Symbol können Sie sich den Klang anhören.



Verlassen Sie die Ansicht **Klänge** mit einem Klick auf das blaue x-Symbol ganz unten.

Sobald die Nachricht **Musik** empfangen wird, spielt ein Block **spiele Sound bis zum Ende** das ausgewählte Weihnachtslied ab. Dieser Block spielt einen Klang in voller Länge ab, bevor der folgende Block im Programm abgearbeitet wird. Auf diese Weise wird das Lied nicht abgebrochen, falls zwischenzeitlich die Schleife im Bereich **Figuren** zu Ende ist und der Taster wieder gedrückt wurde.

We Wish You a Merry Christmas

Wer diese Melodie in der Vorweihnachtszeit fast täglich in Einkaufszentren und auf Weihnachtsmärkten hört, vermutet dahinter ein modernes amerikanisches Weihnachtslied. Tatsächlich handelt es sich aber um ein Lied aus der englischen Weihnachtstradition, das bis ins Jahr 1500 zurückverfolgt und somit heute urheberrechtsfrei genutzt werden kann.

Frohe Weihnachten!