

Fitnessarmband der Zukunft

Anmerkung

Die folgenden Materialien sind eine Überarbeitung des von der Autorin 2017 veröffentlichten Unterrichtsbeispiels (vgl. <http://www.uni-goettingen.de/de/unterrichtsmaterialien/224495.html> letzter Zugriff am 14.02.2020).

Einleitung

Im Rahmen der Unterrichtseinheit „Das Fitnessarmband der Zukunft“ entwickeln Schülerinnen und Schüler eigenständig Ideen für ein „Fitnessarmband der Zukunft“. Dabei werden ausgewählte Funktionalitäten wie beispielsweise ein Schrittzähler eigenständig umgesetzt. Im Vordergrund der gesamten Sequenz steht der kreative Charakter sowie das Erschaffen eines eigenen Produktes. So werden die Schülerinnen und Schüler explizit dazu aufgefordert, sich neben dem Erfassen der gelaufenen Schritte weitere Funktionalitäten zu überlegen und über mögliche Realisierungen nachzudenken. Die konkrete Umsetzung des Schrittzählers fördert Kompetenzen im Bereich des algorithmischen Problemlösens unter Berücksichtigung von Sensoren und Aktoren. Gleichzeitig lädt die Aufgabe zur Entwicklung eines „Fitnessarmbandes der Zukunft“ zur Diskussion vielfältiger anderer informatischer Aspekte ein: Was ist der Unterschied zwischen Daten und Information? Welche Informationen kann man alles aus den gewonnenen Daten über den Nutzer gewinnen? Hier haben Schülerinnen und Schüler ganz unterschiedliche Ideen, die häufig zu leidenschaftlichen Diskussionen in Lerngruppen führen. Beispiele sind etwa: „Das Fitness-Armband sucht aus allen Nutzern einen geeigneten Partner“, „Das Fitnessarmband untersucht, ob jemand lügt“, „Das Fitnessarmband empfiehlt einen guten Arzt.“

In der Praxis hat sich gezeigt, dass Schülerinnen und Schüler bei der Entwicklung eines „eigenen“ Fitnessarmbandes von sich aus häufig auch eine Diskussion unterschiedlicher Fragen im Bereich „Informatik und Gesellschaft“ einfordern: Wie können die erhobenen Daten geschützt werden? Warum erfordern die meisten verkäuflichen Fitnessarmbänder eine Registrierung und Weiterleitung der erhobenen Daten an einen Server des Herstellers, wenn die eigenständige Umsetzung gezeigt hat, dass dies gar nicht nötig ist? Wer interessiert sich eigentlich für die Daten und warum? Kann man eigentlich „schummeln“ und Schrittzahlen erfassen, die gar nicht stimmen?

In diesem Sinn trägt das Unterrichtsbeispiels zur Mündigkeit der Schülerinnen und Schüler bei. Algorithmen wie etwa zur Erfassung der gelaufenen Schritte können verstanden werden auch ohne die konkreten Umsetzungen der verschiedenen Firmen zu kennen. Durch die eigenständige Umsetzung ausgewählter Funktionalitäten können verschiedene Aspekte, beispielsweise zum Datenschutz oder zu den Konsequenzen der Erfassung der Daten, fundierter selbst beurteilt werden. Zielgruppe der Unterrichtssequenz zum „Fitnessarmband der Zukunft“ sind Lerngruppen, die bereits über erste Vorerfahrungen im Umgang mit einer grafischen Programmiersprache verfügen. Durchgeführt und erprobt wurde sie in Lerngruppen der Jahrgänge 9, 10 und 11.

Mögliche Gestaltung einer Sequenz

Im Folgenden wird ein möglicher Unterrichtsgang skizziert. Dabei können je nach Lerngruppe und zur Verfügung stehender Zeit einzelne Blöcke auch variiert, getauscht oder ganz weggelassen werden.

1. Block: Informatik macht fit – Visionen für ein Fitnessarmband (ca. 2 Doppelstunden)

Die Lernenden entwickeln in Kleingruppen eigene Ideen für ein Fitnessarmband der Zukunft. Diese sollen möglichst kreativ sein und sich nicht an bereits vorhandenen Produkten oder möglichen Schwierigkeiten bei der technischen Umsetzung orientieren. Es hat sich gezeigt, dass Schülerinnen und Schüler in der ersten Phase bereits häufig auch auf neue Aspekte kommen, die tatsächlich noch nicht in erwerbenden Fitnessarmbändern umgesetzt wurden. Auch können hier bereits erste Diskussionen im Bereich „Informatik und Gesellschaft“ entstehen.

Möglicher Unterrichtsgang:

- Brainstorming in Gruppenarbeit zum „Fitnessarmband der Zukunft“. Dabei sollte den Lernenden unbedingt klar sein, dass sie noch nicht über technische Realisierungen nachdenken und sich auch nicht überlegen sollen, was es schon gibt. Stattdessen sollen sie wirklich möglichst kreativ sein.
- Vorbereitung des Austauschs, z.B. durch Erstellung eines Plakats, einer Präsentation o.ä.
- Austausch der Ideen, z.B. durch Präsentationen der einzelnen Gruppen im Plenum oder in Form eines Museumsrundgangs

2. Block: Umsetzung eines Schrittzählers (ca. 2 Doppelstunden)

Ein „Fitnessarmband der Zukunft“ sollte die gelaufenen Schritte zählen können. Daher bietet es sich an, diese Funktionalität von allen Lernenden implementieren zu lassen.

Zur Bestimmung der gelaufenen Schritte wird ein entsprechender Sensor benötigt. Im Idealfall steht Hardware wie beispielsweise ein Calliope mini, ein micro:bit o.ä. zur Verfügung. Diese Mikrocontroller enthalten jeweils bereits einen 3-Achsen-Beschleunigungssensor. Alternativ kann ein Schrittzähler auch mit einem Arduino mit angeschlossenem 3-Achsen-Beschleunigungssensor umgesetzt werden. Auch hier ist eine Implementierung in einer grafischen Programmiersprache möglich (zum Beispiel microBlocks vgl. <http://microblocks.fun/> letzter Zugriff am 17.02.2020 oder mBlock vgl. <https://www.mblock.cc/en-us/> letzter Zugriff am 19.02.2020).

Zur Implementierung sind die folgenden Arbeitsschritte nötig:

Arbeitsschritt 1 – Analyse der Werte des vorhandenen 3-Achsen-Beschleunigungssensors: Wie verhalten sich die erfassten Sensorwerte bei einem Schritt? Welche Werte sind bei der Interpretation der Daten als Schritt relevant, welche nicht? Diese Analyse und Interpretation der Werte des Beschleunigungssensors stellt eine Herausforderung bei der Implementierung eines Schrittzählers dar.

Arbeitsschritt 2 – Tatsächliche Implementierung unter Verwendung der Beobachtungen aus Schritt 1: Hat man erst einmal festgelegt, welche Sensorwerte wie als Schritt interpretiert werden können, muss dies noch als Algorithmus implementiert werden.

Im Folgenden werden verschiedene Varianten zur Umsetzung der beiden Arbeitsschritte mithilfe eines Calliope vorgestellt. Die Verwendung eines micro:bit erfolgt analog.

Variante 1: Umsetzung mit microBlocks

microBlocks ist eine Scratch-ähnliche Programmierumgebung, die das Arbeiten mit verschiedensten Microcontroller erlaubt. Das Programm sowie genauere Beschreibungen findet man unter <http://microblocks.fun/> (letzter Zugriff am 17.02.2020).

Möchte man erstmalig einen Mikrocontroller mit microBlocks programmieren, so muss man über Einstellungen → update firmware on board zunächst die zugehörige Firmware auf das Board übertragen. Anschließend signalisiert das grün unterlegte USB-Symbol die vorhandene Verbindung und die zum Board zugehörigen Blöcke erscheinen als weitere Kategorie.

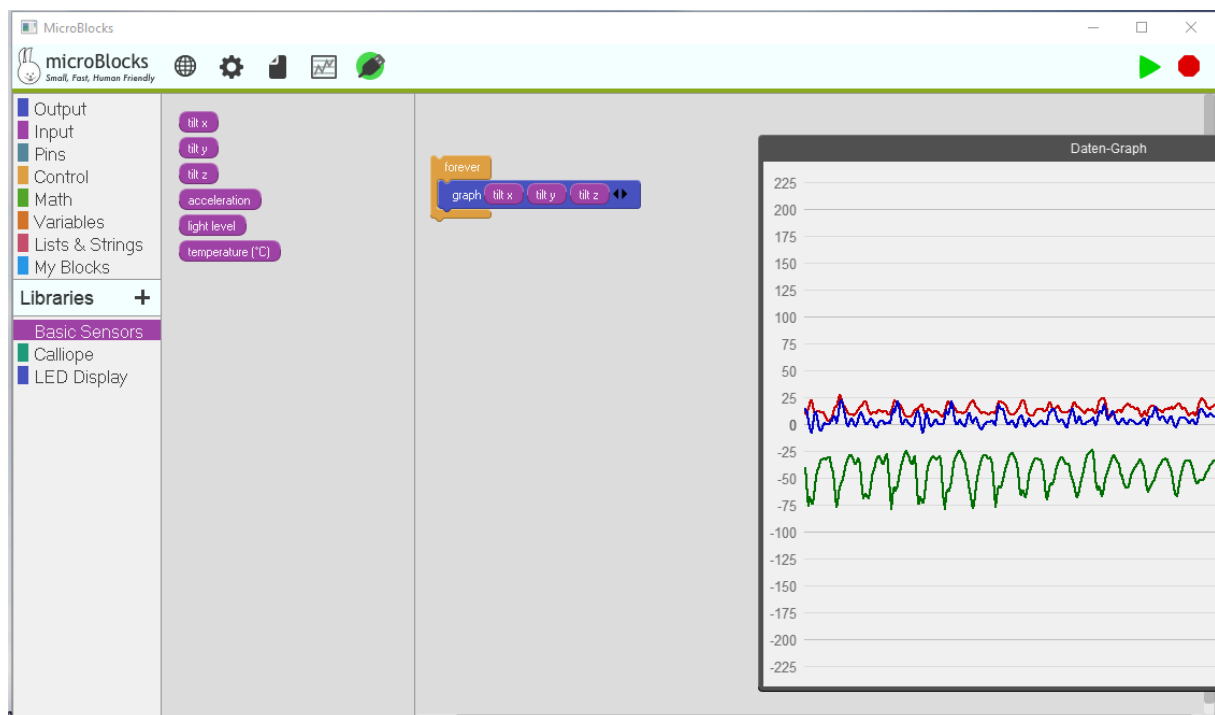
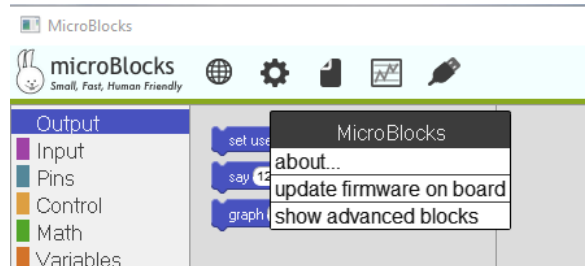


Abbildung 1

In Abbildung 1 ist dargestellt, wie die Sensorwerte des 3-Achsenbeschleunigungssensors grafisch veranschaulicht werden können. Dabei kann die Darstellung variieren, je nachdem, wie der Calliope beim Gehen gehalten wird. Die grafische Darstellung erlaubt dann Rückschlüsse auf die zu verwendenden Daten des 3-Achsenbeschleunigungssensors.

In Abbildung 2 ist eine mögliche Implementierung eines Schrittzählers in microBlocks dargestellt.



Abbildung 2

Variante 2: Umsetzung in Snap!

Zur Zeit gibt es noch keine Möglichkeit, die Sensorwerte eines Mikrocontrollers wie beispielsweise des Calliopes direkt in Snap einzulesen und zu verwerten. Deswegen hat Andreas Flemming für das Lehrerbildungszentrum Informatik an der Georg-August-Universität Göttingen 2017 eine entsprechende Software entwickelt, die die Sensordaten an Snap! weiterleitet (vgl. <http://www.uni-goettingen.de/de/unterrichtsmaterialien/224495.html> letzter Zugriff am 17.02.2020. Die Materialien wurden von Andreas Flemming erstellt und unter einer CC-BY-NC-SA lizenziert. Der Ordner LBZ-Andreas-Flemming-2017 enthält einen Auszug der dort veröffentlichten Materialien. Die Snap!-Datei Grundprogramm.xml wurde um einen Block „Werte auslesen“ erweitert. Weitere Änderungen wurden nicht vorgenommen.)

Die Auswertung in Snap! erhält man wie folgt:

- Aufspielen der Firmware `firmware_calliope.hex` auf den Calliope (Kopieren der Datei in das entsprechende Verzeichnis des Calliopes)
- Starten des Programms `hardware_an_snap_proc` zur Weiterleitung (in Processing; alternativ kann einmalig in Processing eine ausführbare Datei erstellt werden, die dann unabhängig von Processing läuft).
- Implementierung der gewünschten Funktionalitäten in der Datei `Grundprogramm.xml` in Snap!. Dabei muss einmalig der Block „Werte auslesen“ aktiviert werden, um kontinuierlich die aktuellen Sensorwerte in den zugehörigen Variablen zu speichern.

Diese drei Schritte können im Hintergrund laufen und müssen von den Lernenden nicht verstanden werden. Zentral ist, dass im Anschluss an diese drei Schritte die Sensorwerte kontinuierlich in Snap! in den zugehörigen Variablen angezeigt und gespeichert werden. Will man sich den Verlauf der Werte wie in microBlocks grafisch anzeigen, muss dies eigenständig implementiert werden. Alternativ kann das Programm Auswertung.xml genutzt werden.

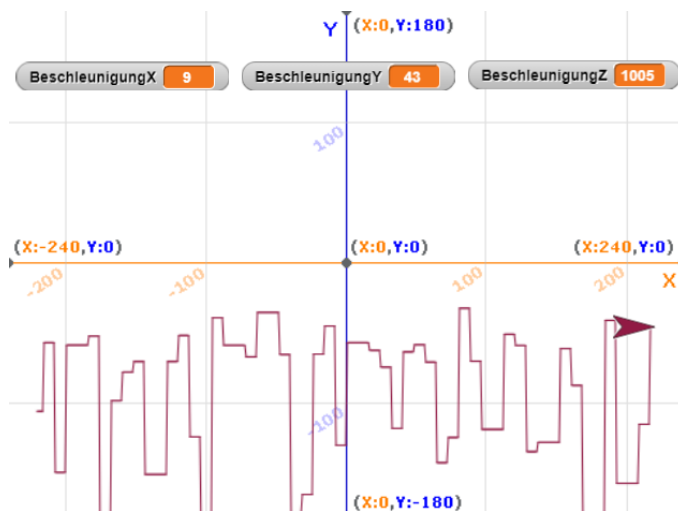


Abbildung 3

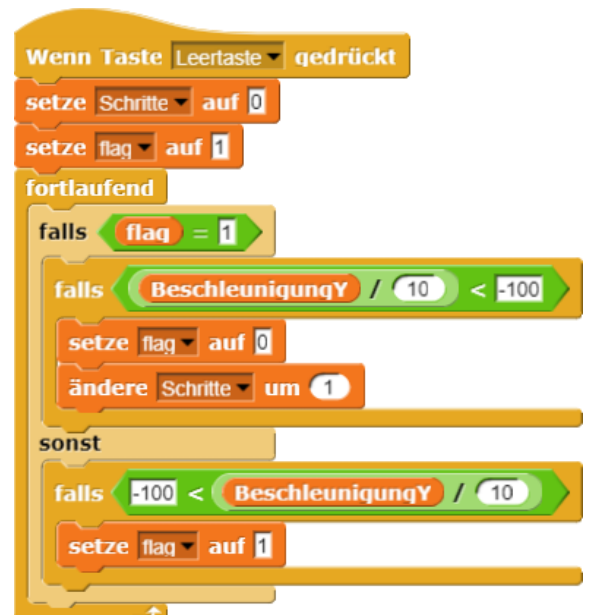


Abbildung 4

In Abbildung 3 wird ein beispielhafter Verlauf der Sensorwerte BeschleunigungY dargestellt.

Abbildung 4 zeigt eine mögliche Implementierung eines Schrittzählers in Snap!

3. Block: individuell unterschiedliche Umsetzung weiterer Funktionalitäten (Dauer hängt stark von Ideen und Lerngruppe ab)

Hier gibt es vielfältige Gestaltungsmöglichkeiten mit stark variierendem Schwierigkeitsgrad. Mögliche Ideen sind beispielsweise:

- Abfrage des Namens und anschließende persönliche Begrüßung
- Berechnung der gelaufenen Kilometer aus Schrittzahl und Schrittweite
- Abfrage weiterer Daten und Berechnung zugehöriger Fitnessdaten wie beispielsweise BMI, Einteilung in „Fitnesslevel“ o.ä.
- Schlafanalyse
- „Stimmungsanalyse“ über Interpretation der Sensorwerte des Mikrofons
- uvm.

4. Block: Analyse von „Konkurrenzprodukten“ (ca. 1 Doppelstunde)

Häufig sind Lernende überrascht, dass sie tatsächlich einen eigenen Schrittzähler implementieren können. Sind Fitnessarmbänder vorhanden, bietet sich ein Vergleich der eigenen Produkte mit den industriell hergestellten Armbändern an. Dabei wird deutlich, dass die selbst programmierten Schrittzähler genauso gut Schritte zählen wie die gekauften. Auch werden Unterschiede beispielsweise zur Verarbeitung der Daten sichtbar. Bei manchen Fitnessarmbändern ist eine vorherige Registrierung notwendig, je nach Anbieter müssen dabei verschiedene persönliche und möglicherweise sensible Daten (zum Beispiel Gewicht, Alter, Körpergröße) angegeben werden. Dabei stellt sich den Lernenden die Frage, wofür diese Daten benötigt werden. Manche Fitnessarmbänder zeigen die gelaufenen Schritte direkt an, andere erfordern eine Synchronisierung mit einem Smartphone, d.h. der Ort der Datenverarbeitung unterscheidet sich je nach Hersteller. Auch die Qualität der kommerziellen Schrittzähler ist sehr unterschiedlich. All diese Aspekte erhöhen den „Produktstolz“ der Lernenden enorm.

Dieses Werk ist lizenziert unter einer [Creative Commons Namensnennung - Nicht kommerziell - Keine Bearbeitungen 4.0 International Lizenz](#). Sie erlaubt Download und Weiterverteilung des vollständigen Werkes unter Nennung meines Namens, jedoch keinerlei Bearbeitung oder kommerzielle Nutzung.

Für die korrekte Ausführbarkeit der Quelltexte in diesem Text wird keine Garantie übernommen. Auch für Folgeschäden, die sich aus der Anwendung der Quelltexte oder durch eventuell fehlerhafte Angaben ergeben, wird keine Haftung oder juristische Verantwortung übernommen.