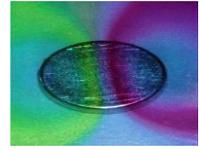




IMST – Innovationen machen Schulen Top

Themenprogramm: Kompetenzen im mathematischen
und naturwissenschaftlichen Unterricht



AUTOMATISIERUNG OHNE KONTROLLVERLUST

Kurzfassung

ID 1796

Georg Reich

**Georg Reich
NMS Stallhofen**

Stallhofen, Juni 2016

DIE IDEE

1.1 Ziele auf SchülerInnenebene

Einstellung: Programmieren soll "salonfähig" werden. Die Fähigkeit zu Programmieren ermöglicht uns Computer und Tablet nicht nur als Anwender zu verwenden sondern erlaubt uns Programme, Apps, etc. auch zu erstellen/gestalten. Ziel ist es, dass sich die Kinder nicht nur als "User" wahrnehmen sondern erkennen, dass sie in weltweiten Community ebenso als Programmierer agieren können.

Handlungen: Schülerinnen und Schüler sollen mit Hilfe von Microcontrollern Schaltungen bauen und diese dann mit geeigneter Software programmieren.

Kompetenzen: Komplexe, aber altersadäquate Codes sollen verstanden werden können.

1.1.1 Überfachliche Kompetenzen

- Schülerinnen und Schüler nehmen neue Herausforderungen als Anreiz und nicht als Bürde wahr.
- Schülerinnen und Schüler denken nicht mehr im Schema eines streng getrennten Fächerkanons.

1.1.2 Fachliche Kompetenzen

- Schülerinnen und Schüler können die grundlegende Funktionsweise eines Microcontrollers anhand einer schematischen Darstellung erklären.
- Schülerinnen und Schüler sind in der Lage eigenständig eine elektronische Schaltung aufzubauen.
- Schülerinnen und Schüler können Sensoren (z.B. Lichtsensor, Temperatursensor, Taster,...) an einen Microcontroller richtig anschließen und die Werte auslesen.
- Schülerinnen und Schüler können Verbraucher (Led, Motor,...) an einen Microcontroller anschließen und ansteuern.
- Schülerinnen und Schüler können mit grafischer Programmiersoftware (z.B. Scratch) Programmstrukturen erstellen, die verschiedene Vorgänge automatisieren (z.B. wenn es dunkel wird – schalte Licht ein)
- Schülerinnen und Schüler können grafisch programmierte Programmstrukturen in textbasierte Programmiersprachen übertragen.

1.2 Ziele auf LehrerInnenebene

Einstellung: Ich höre manchmal wie wehmütig Menschen der "guten alten Zeit" nachtrauern. Nicht nur Lehrerinnen und Lehrer sind manchmal der Meinung, dass die Kinder früher leistungsfähiger,

braver,... gewesen wären. Fakt ist, dass sich im Lauf der Zeit eine Gesellschaft immer ändert - und das ist auch gut so. Das hat zur Folge, dass sich das Aufgabenfeld verändert - besonders jenes von Pädagoginnen und Pädagogen. Kinder von heute haben andere Interessen, andere Stärken und auch andere Anforderungen - unsere Aufgabe ist es sie auf künftige Herausforderungen vorzubereiten. Ich denke, dass ein interessanter, freudvoll gestalteter Unterricht unseren Kindern hilft im Leben zurecht zu kommen.

Handlungen: Der Unterricht soll so gestaltet werden, dass gegenseitiger Respekt selbstverständlich wird. Das Handeln wird stark von der Motivation beeinflusst - diese wird wiederum von den Ergebnissen beeinflusst - so ist es wichtig, dass immer ein gutes Arbeitsklima herrscht.

Kompetenzen: Lernen bedeutet Interaktion - das heißt nicht nur die Kinder lernen, sondern auch wir Lehrer. Bei diesem Projekt handelt es sich um einen doch recht komplexen Lehrinhalt (Programmieren). Ziel ist es geeignete Unterrichtsmethoden zu finden und weiterzuentwickeln, die die Weitergabe dieses Lehrstoffs ermöglichen.

- Kolleginnen und Kollegen sollen den Mehrwert von praktischen und selbstständigen Arbeitsformen beobachten können.
- Kolleginnen und Kollegen sollen motiviert werden neue Unterrichtsmethoden auszuprobieren.
- Kolleginnen und Kollegen sollen Evaluationsmethoden kennen lernen und eine kritische Reflexion des eigenen Unterrichts als einen positiven Anreiz im Sinne einer Weiterentwicklung betrachten.

DAS PROJEKT

1.3 Beschreibung der Umsetzung

1.3.1 Projekttag 1:

Schülerinnen und Schüler hatten folgende Aufgabenstellungen:

1. Eine LED mittels Tasten auf der Computertastatur ein – und wieder auszuschalten.

Die Bedienung der Programmiersoftware Scratch war bekannt. Als erstes bauten sie die entsprechende Schaltung auf – das heißt sie verbanden den Plus-Pol der Diode mit einem schaltbaren Ausgang am und den Minus-Pol mit der Masse des Microcontrollers. Dann war nur noch ein entsprechendes Programm notwendig, dass auf den Tastendruck der Computertastatur reagierte. Hier die Lösung eines Zweier-Teams:

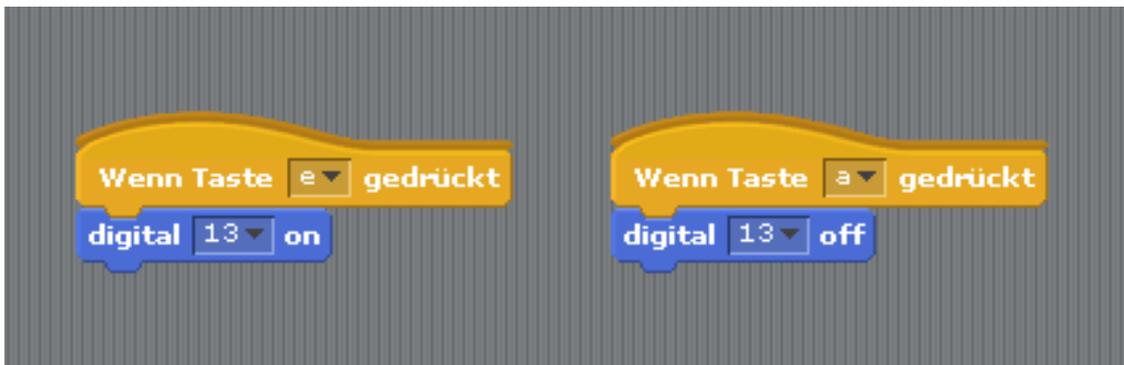


Abbildung 1: Hier wird mit der Taste „e“ eine LED eingeschaltet und mit der Taste „a“ diese wieder ausgeschaltet. Angeschlossen muss sie beim Ausgang 13 des Arduino-Boards sein.



Abbildung 2: Schaltung, wenn die Taste „a“ gedrückt wurde



Abbildung 3: Schaltung, wenn die Taste „e“ gedrückt wurde

Am

Ende des Projekttags hab ich folgende 3 Fragen gestellt:

1. Was hat euch heute besonders gut gefallen?

„alles“; „selbstständiges Arbeiten“; „...dass man viel Neues gesehen hat“; „das Programmieren“

2. Was hat euch heute nicht so gut gefallen?

Antworten (exemplarisch): „Es war nicht so toll, dass es am Anfang nicht so gut funktioniert hat.“; „Die vielen Kabel.“; „...dass Lisa vorbeigekommen ist.“; „nichts“

3. Was habt ihr heute neu dazugelernt?

„...wie man mit Scratch4Arduino eine Ampel zusammenschließt.“; „...wie man programmiert.“; „...wie ich Kabel verknüpfe – soll heißen ...wie man eine Schaltung richtig aufbaut“; „Eine Ampelschaltung programmieren“

1.3.2 Projekttag 2:

Diesmal widmeten wir uns der nächsten Herausforderung – dem Programmieren mit der Arduino-Software. Im Unterschied zu Scratch ist diese Text-basiert.

Der entscheidende Vorteil ist nun, dass die verschiedenen Programme vom Microcontroller autonom ausgeführt werden können. Bei Scratch war immer eine USB-Verbindung vom PC zum Microcontroller notwendig.

Zu Beginn wurden Programmstruktur und Syntax erklärt.



```
Arduino IDE: _2_taster_auslesen | Arduino 1.6.5
Datei Bearbeiten Sketch Werkzeuge Hilfe

_2_taster_auslesen §

int led_1 = 13;
int taster_1 = 7;
int x_1 = 0;

void setup()
{
  pinMode(led_1, OUTPUT);
  pinMode(taster_1, INPUT_PULLUP);
}

void loop()
{
  if (digitalRead(taster_1) == LOW && (x_1 == 0))
  {
    digitalWrite(led_1, HIGH);
    x_1 = 1;
    delay(300);
  }
  if (digitalRead(taster_1) == LOW && (x_1 == 1))
  {
    digitalWrite(led_1, LOW);
    x_1 = 0;
    delay(300);
  }
}
```

Abbildung 4: Arduino-Code zum Ein- und Ausschalten einer Lichtquelle mit nur einem Taster.

Die Abbildung zeigt ein Programm, dass mit ein und dem selben Schalter eine Lampe ein und wieder ausschalten kann. Weitere Aufgabenstellungen waren:

- Bringe eine LED zum Blinken – bzw. erstelle in weiterer Folge ein Programm, das den Morsecode für SOS in Form von Lichtsignalen wiedergibt.
- Programmiere ein Lauflicht mit zumindest 10 LEDs
- Baue eine Ampelschaltung und programmiere sie entsprechend.
- Plane eine eigene Schaltung. Wichtig dabei ist der richtige Aufbau!

Die Schülerinnen und Schüler arbeiteten dann weitgehend selbstständig an den verschiedenen Aufgaben. Alle Infos, die zum programmieren nötig waren standen in Form von Info-Sheets zur Verfügung.

1.3.3 Projekttag 3 – Workshop auf der TU Graz zum Theme „Programmieren mit einem Raspberry Pi“

Am 10. März 2016 besuchte die Wahlpflichtfachgruppe das Institut für Robotik auf der Technischen Universität Graz.

Dort arbeiteten die Schülerinnen und Schüler mit Raspberry Pi-s. Die dort verwendete Programmiersprache nennt sich Python. Wie Arduino ist auch Python eine Text-basierte Programmiersprache. Nach einer kurzen Einführung konnten die einzelnen Teams bereits einfache Aufgabenstellungen selbstständig lösen.

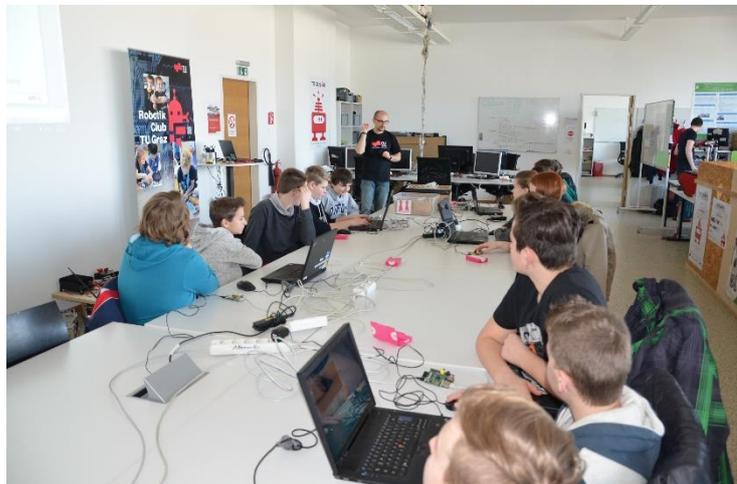


Abbildung 5: Workshop im Robotik-Labor der TU Graz

Im Unterschied zum Arduino-Board, das ein Microcontroller ist, ist der Raspberry Pi ein vollwertiger Computer, der üblicherweise Linux als Betriebssystem verwendet und wie jeder andere PC mit Maus und Tastatur bedient werden kann.

Der Raspberry Pi besitzt wie auch das Arduino-Board eine Allzweckeingabe bzw. -ausgabe (engl. GPIO – general purpose input/output), welche angesteuert werden kann – jedoch beschränkten sich die Aufgabenstellungen in diesem ersten Einführungs-Workshop auf Programmieraufgaben, die rein virtuell (am Bildschirm) abliefen.

Zwei Schülerinnen verfassten folgenden Bericht über diesen Projekttag:

Besuch auf der TU Graz

Am 10.März.2016 fuhren wir (die Schülerinnen und Schüler der Robotik Gruppe) auf die Technische Universität Graz um einen Workshop zum Thema Raspberry Pi zu besuchen.

Zuerst wurde uns das Robotik Labor der TU vorgestellt wo wir einen Blick in den Alltag der Studenten werfen durften.

Bis jetzt haben wir in der Schule hauptsächlich grafisch programmiert. Hier wurde uns gezeigt wie man mit einer anderen Programmiersprache (Python) leicht arbeiten kann.

Wir erstellten mit Hilfe von „Mini-Computer“ (Raspberry Pi) und dem Programm Python_sogenannte Turtles (Schildkröten) die wir danach so programmierten, dass sie z.B. Dreiecke oder Vierecke „malen“ konnten. Ebenfalls lernten wir das Benutzen einer Schleife, sodass die Formen auch öfters wiederholt wurden.

Natürlich war es für uns schwieriger mit dieser Programmiersprache zu arbeiten, da wir es gewohnt waren grafisch zu programmieren. Wir mussten jetzt alle Befehle einzeln aufschreiben.

Trotz allem war der Vormittag auf der TU sehr lustig und lehrreich.

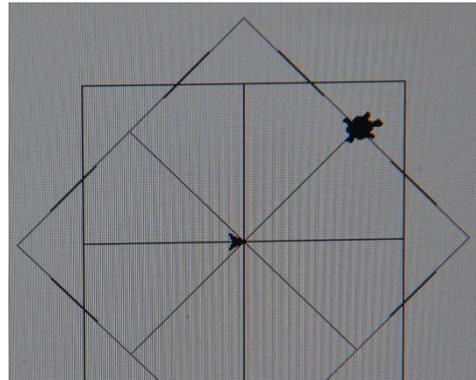


Abbildung 6: Screenshot der Bildschirm- ausgabe einer Python-Programmierung

Simone und Elena 4m

1.3.4 Projekttag 4 – Vortrag eines Mitarbeiters der Firma HMI Master

Am Montag den 09. Mai 2016 führte Wolfgang Amreich von der Firma HMI Master einen Workshop zum Thema Gebäudeautomation durch. Das Angebot der Firma reicht von Heizungssteuerung über Energiemanagement, Datenmonitoring bis hin zu diversen Speziallösungen.

Der Zugang zum Thema wurde am Beispiel CO₂-Gehalt in der Luft hergestellt. Die Kinder arbeiteten engagiert mit. In einem motivierten Lehrer-Schüler-Gespräch wurden die verschiedenen Symptome, die mit einem zu hohen CO₂-Gehalt in einem Raum einhergehen (Konzentrationsschwäche,...), besprochen.



Abbildung 7: Vortrag zum Thema Gebäudeautomation von Herrn Wolfgang Amreich



Ergebnis der Diskussion war, dass ein System, welches einen zu hohen CO₂-Gehalt automatisch erkennt und darauf reagiert, zu einem besseren Wohlbefinden führt.

Abbildung 8: Schülerinnen während des Vortrags

Ein System, das den CO₂-Gehalt regulieren kann, sieht möglicherweise wie folgt aus:

Bestandteile:

- CO₂-Sensor (Eingabegerät)
- Controller (Herzstück des Systems)
- Motorisierter Fensteröffner (Ausgabegerät)

Der Controller wird so programmiert, dass er ständig den CO₂-Gehalt in einem Raum erfasst. Steigt der Anteil auf einen ungesunden Wert an, gibt der Controller ein Signal an den motorisierten Fensteröffner und dem Raum wird so lange Frischluft zugeführt, bis der CO₂-Gehalt wieder im „grünen“ Bereich ist.

Eine alternative Variante wäre ein System, wo statt dem motorisierten Fensteröffner eine sogenannte kontrollierte Wohnraumlüftung verbaut ist.

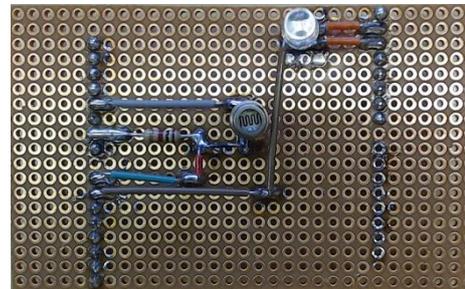
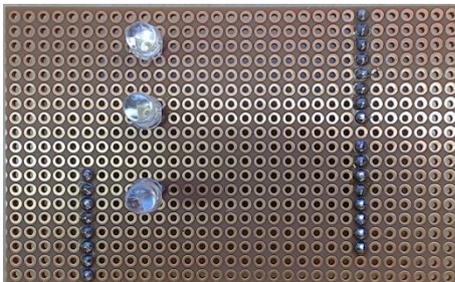
DIE ERGEBNISSE

1.4 Beschreibung einer Leistungsaufgabe

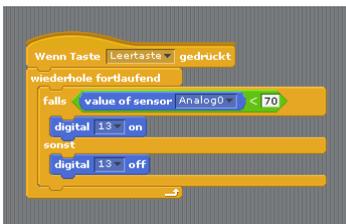
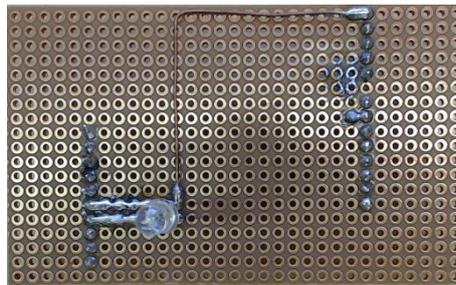
Arbeitsblatt – Leistungsaufgabe:

WAS GEHÖRT ZUSAMMEN?

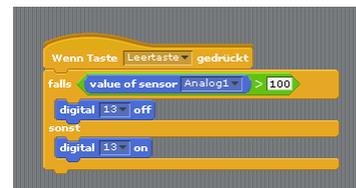
Kennzeichne die zusammengehörenden Abbildungen und Begriffe mit jeweils dem gleichen Buchstaben (A, B und C).



Taster



Dämmerungsschalter



Ampel

In dieser ersten Leistungsaufgabe kann überprüft werden, ob die Schaltungen, Programmierungen wiedererkannt und richtig benannt werden können.

Bei der Überprüfung konnten mehr als 90% der Schülerinnen und Schüler diese spezielle Aufgabenstellung richtig lösen.

Mich interessierte aber auch die Einstellung der Schülerinnen und Schüler zu diesem doch speziellen Lehrinhalt. So stellte ich vor und nach dem 4. Projekttag folgende Fragen:

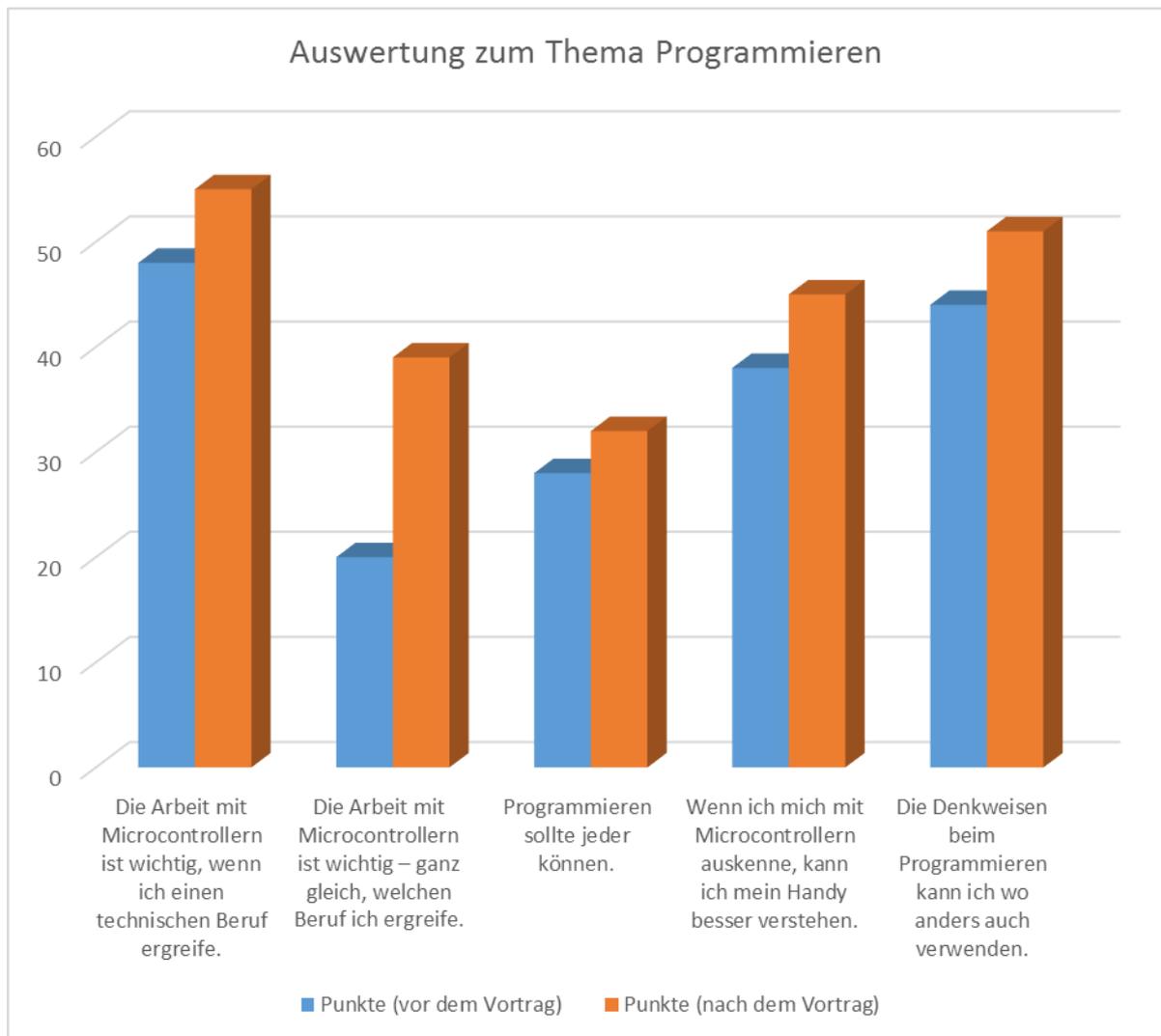
Die Arbeit mit Microcontrollern ist wichtig, wenn ich einen technischen Beruf ergreife.					
Die Arbeit mit Microcontrollern ist wichtig – ganz gleich, welchen Beruf ich ergreife.					
Programmieren sollte jeder können.					
Wenn ich mich mit Microcontrollern auskenne, kann ich mein Handy besser verstehen.					
Die Denkweisen beim Programmieren kann ich wo anders auch verwenden.					

Die Schülerinnen und Schüler konnten bis zu 5 Sterne für ihr Maß an Zustimmung der einzelnen Aussagen geben.

Es wurden 12 Kinder befragt, davon waren 2 weiblich.

1.5 Daten

Frage	Punkte (vor dem Vortrag)	Punkte (nach dem Vortrag)
Die Arbeit mit Microcontrollern ist wichtig, wenn ich einen technischen Beruf ergreife.	48	55
Die Arbeit mit Microcontrollern ist wichtig – ganz gleich, welchen Beruf ich ergreife.	20	39
Programmieren sollte jeder können.	28	32
Wenn ich mich mit Microcontrollern auskenne, kann ich mein Handy besser verstehen.	38	45
Die Denkweisen beim Programmieren kann ich wo anders auch verwenden.	44	51



1.6 Interpretation

In der Tabelle, so wie im Diagramm ist zu erkennen, dass nach einer intensiven Außernandersetzung mit dem Thema Programmieren von Microcontrollern die Bedeutung nach Meinung der Schülerinnen und Schülern noch gestiegen ist. Zum Teil war die Zustimmung schon vor dem Workshop recht hoch, trotzdem hat es nach dem Projekttag eine noch höhere Zustimmung in allen Bereichen gegeben.