Informatikunterricht in der Grundschule? Erprobung und Auswertung eines Unterrichtsmoduls mit Calliope mini

Kevin Baum, Nadine Kirsch, Kerstin Reese,
Pascal Schmidt, Lukas Wachter, Verena Wolf

UNIVERSITÄT DES SAARLANDES

SIC Saarland Informatics Campus

Ausgangssituation

In Deutschland, z.B.:

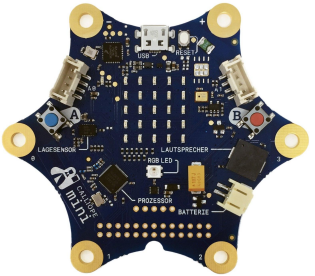
- Informatiksysteme in der Lebenswelt von Kindern
- Vorstellung Calliope mini auf dem IT-Gipfel 2016 im Saarland

Im Saarland, z.B.:

- Informatische Inhalte fehlen in den Lehrplänen der Grundschule
- Seit 2017 soll Calliope mini ab Klasse 3 eingesetzt werden
- Lehrerfortbildungen seit Frühjahr 2017

Aber: Kaum Antworten darauf, ob Grundschulkinder Informatik lernen können und ob das mit Calliope mini geht.

Motivation und Vorgehen



Wir haben uns gefragt:

- Können Grundschulkinder Informatiksysteme als von Menschen gemachte Systeme erkennen und altersgerecht erklären?
- Können Grundschulkinder ein Informatiksystem als reaktives System verstehen und selber programmieren?

So wollten wir vorgehen:

- Vorwissen ermitteln
- Wirksamkeit der entwickelten Unterrichtsmaterialien und des Calliope mini bestimmen

Forschungsfrage A - Programmierung

Gelingt es Kindern der 4. Klasse eigenständig Programme zur Lösung vorgegebener Eingabe-Ausgabe-Zusammenhänge zu entwerfen?

A1 Nutzen die Kinder zielgerichtet algorithmische Grundbausteine (Eingaben, Ausgaben, Sequenzen) sowie Sensoren und Aktoren?

A2 Lassen sich typische Missverständnisse oder Verständnisschwierigkeiten identifizieren?

Forschungsfrage B - Vorwissen

Erkennen Kinder ein ihnen unbekanntes Informatiksystem als solches und wie beschreiben sie ein solches System?

B1 Welche Bestandteile und Funktionsweisen eines beobachteten Informatiksystems nennen die Kinder?

B2 Beschreiben die Kinder einen zutreffenden kausalen Zusammenhang zwischen Eingabe und Ausgabe?

B3 Falls die Kinder einen solchen Zusammenhang hergestellt haben: Welche Ursache schreiben die Kinder dem genannten Zusammenhang zu?

Zeitlicher Ablauf

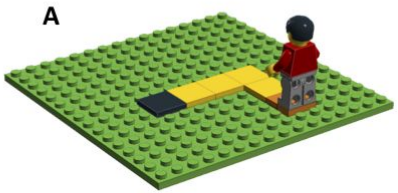
Winter 2017/18
Entwurf der Unterrichtseinheit

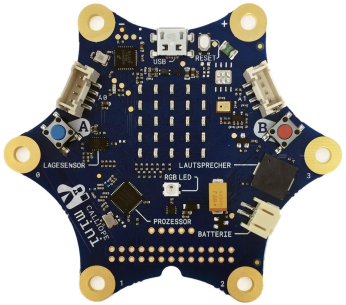
Frühjahr 2018
Erprobung der Unterrichtseinheit als AG
-> Anpassungen

September - Dezember 2018
Erhebungsphase

Klasse	Schule	#SchülerInnen
1	1	20
2	1	17
3	1	21
4	2	18
5	3	17
6	4	15
7	4	18
8	5	11
		137

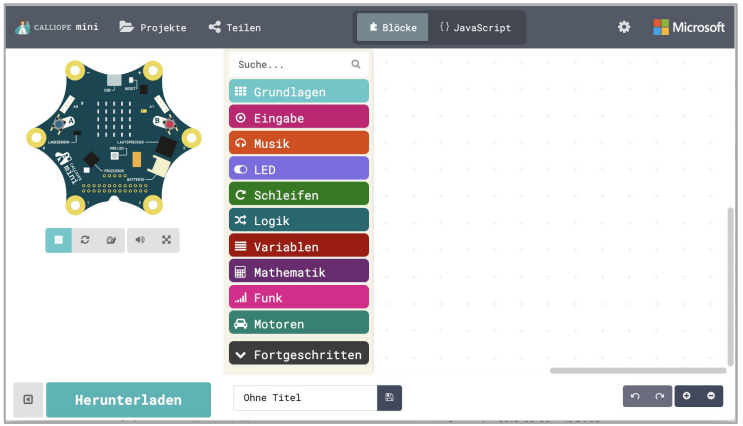
Übersicht Unterrichtseinheit





Drei Doppelstunden:

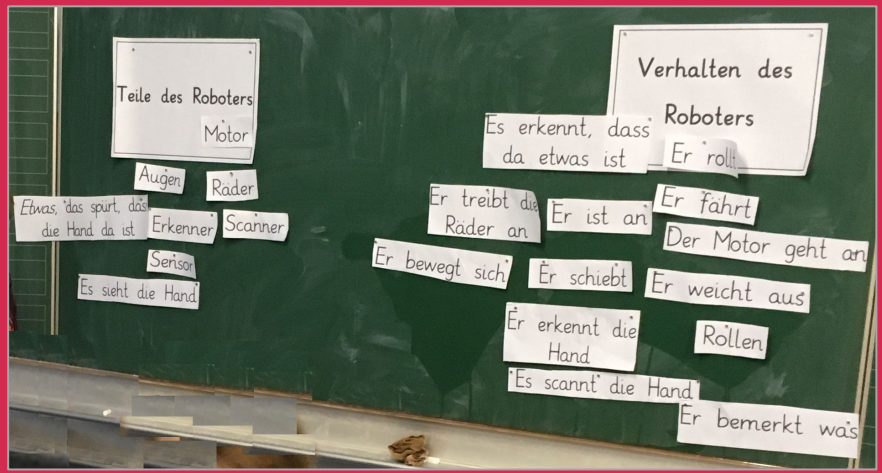
1	1	Interviews
	2	
2	3	Calliope ohne und mit Strom erkunden
	4	
3	5	Test: Programmieren
	6	Freies Programmieren



1. Doppelstunde 1/2

Einzel-Interviews

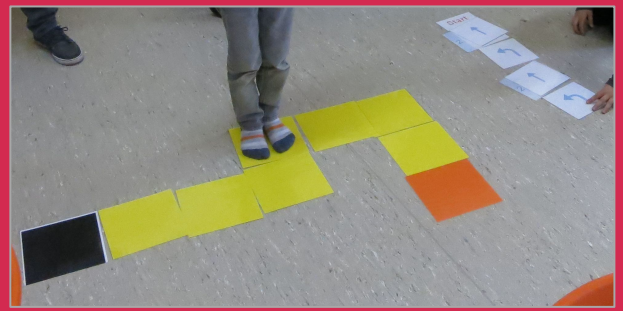
Wortfeldspiel

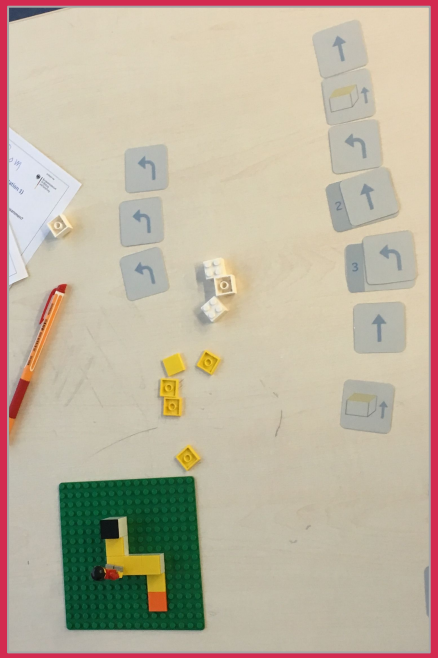


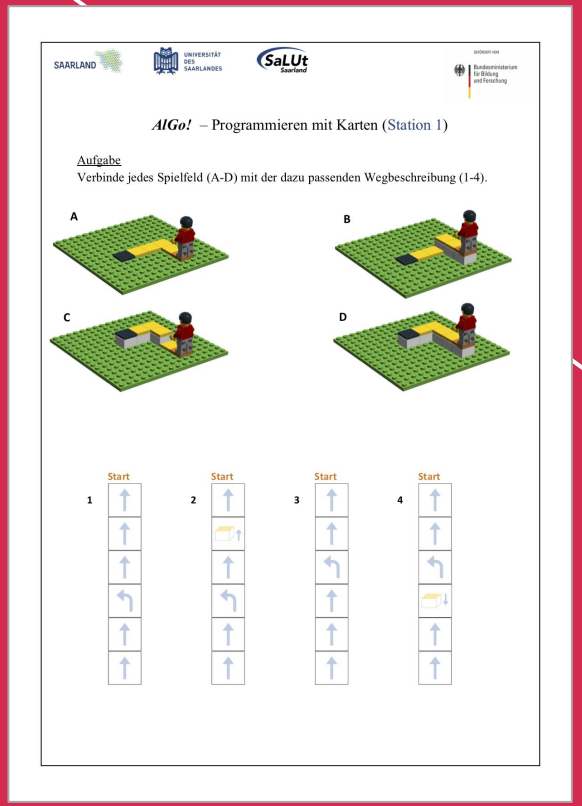
1. Doppelstunde 2/2

AlGo!-Spiel:

- Als Bodenspiel
- Mit Lego in 2er-Teams







SAARLAND · UNIVERSITÄT DES SAARLANDES · SaLUt Saarland · Bundesministerium für Bildung und Forschung

AlGo! – Programmieren mit Karten (Station 1)

Aufgabe

Verbinde jedes Spielfeld (A-D) mit der dazu passenden Wegbeschreibung (1-4).

A

B

C

D

1

2

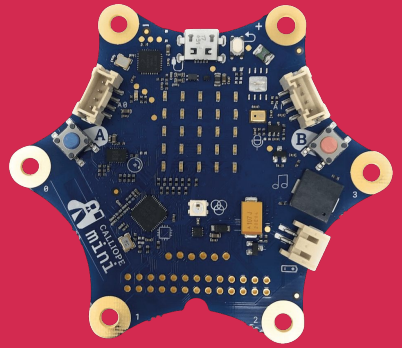
3

4

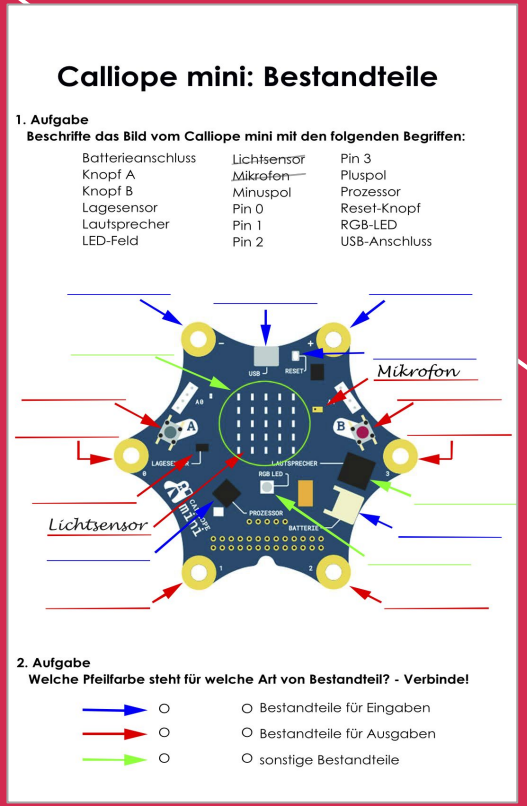
2. Doppelstunde 1/2

- Stromlosen Calliope mini erkunden
- Zuordnung Ein-/Ausgabe
- Erkundung mit Strom.

Eingabe	wenn \ dann	Ausgabe: Smiley	Ausgabe: Ampel
	Taste A	I	II
	Bewegung	III	IV



Feststellung:
Unterschiedliches Verhalten ergibt sich aus unterschiedlicher Programmierung.



Calliope mini: Bestandteile

1. Aufgabe
Beschriffe das Bild vom Calliope mini mit den folgenden Begriffen:

Batterieanschluss	~~Lichtsensor~~	Pin 3
Knopf A	~~Mikrofon~~	Pluspol
Knopf B	Minuspol	Prozessor
Lagesensor	Pin 0	Reset-Knopf
Lautsprecher	Pin 1	RGB-LED
LED-Feld	Pin 2	USB-Anschluss

Mikrofon

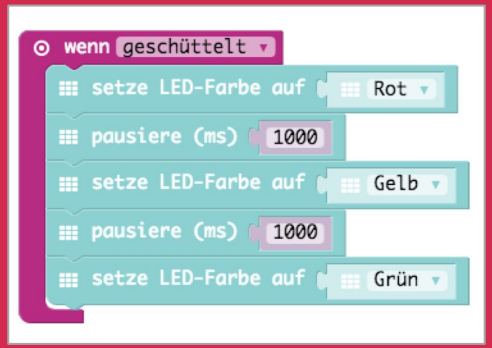
Lichtsensor

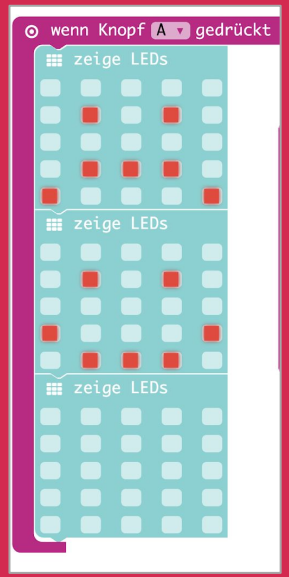
2. Aufgabe
Welche Pfeilfarbe steht für welche Art von Bestandteil? - Verbinde!

- ○ Bestandteile für Eingaben
- ○ Bestandteile für Ausgaben
- ○ sonstige Bestandteile

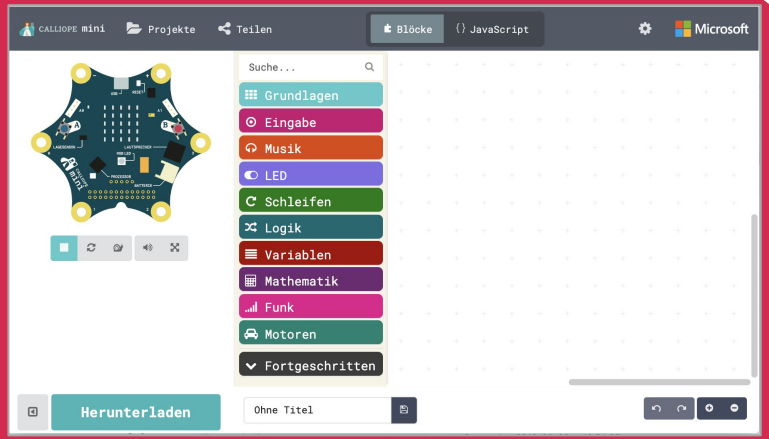
2. Doppelstunde 2/2

- Einführung in den Makecode-Editor am Beamer
- Erstellen von 2 Programmierungen aus der Erkundung





Eingabe \ Ausgabe: wenn \ dann	Smiley	Ampel
Taste A	I	II
Bewegung	III	IV

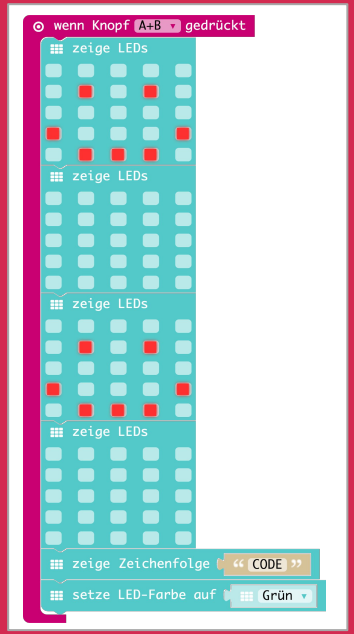


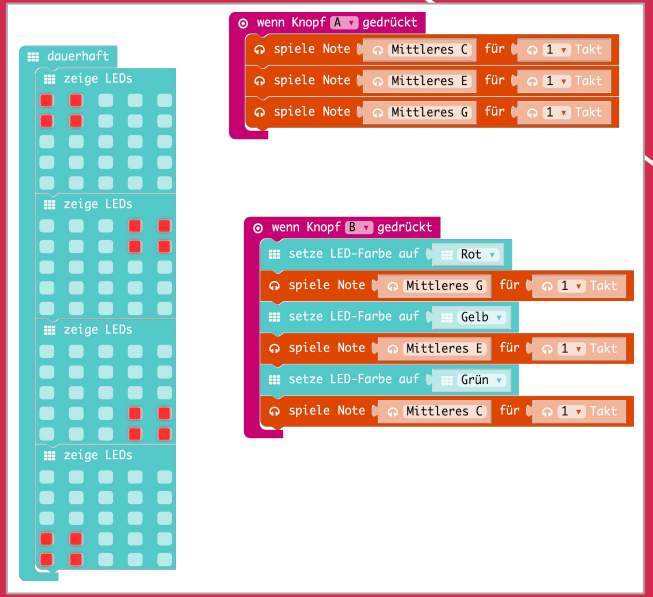
3. Doppelstunde - Test

Nur sequentielle Programme:

1. Reproduktion:
   Bekanntes Programm
2. Transfer:
   Knopf B statt A,
   Text anzeigen, statt Smiley
3. Leichtes Rätsel
4. Schweres Rätsel

Danach freies Programmieren
und Präsentation der eigenen Ideen





Erhebungsinstrumente 1/2
Interview

Während der Interviews:
Notizen (Formular)

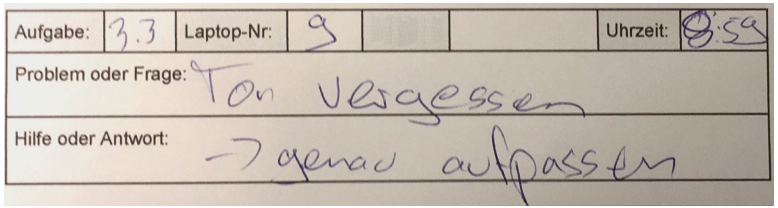
	selbstständig	Auf Nachfragen
Hat eine Bewegung erkannt	☒	☐
Hat Kausalität zwischen Bewegung des Roboters und der Näherung der Hand erkannt	☐	☒
Kann Phänomen reproduzieren	☐	☒

Begründung, warum sich der Roboter bewegen kann:	• Hier leuchtet es • Es hat Rollen • Es ist magnetisch
Vorstellung von „Programmierung“	

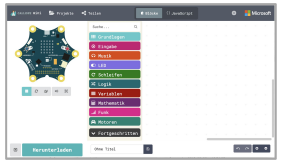
Erhebungsinstrumente 2/2
Screencast

Während der Arbeit am Laptop

- Screencast mit OBS
- Hilfestellungen wurden notiert



Ton- und Videoaufnahmen der Kinder waren nicht erlaubt.



Auswertungsmethoden

Interviews

- Qualitative Auswertung angelehnt an Mayring
  (mit Kategoriensystem)

Programmieren

- Quantitative Auswertung
  Bearbeitungsdauer bis zur Lösung
  Aufgabe vollständig, bis auf kleine Mängel, gar nicht gelöst

Kategoriensystem in Anlehnung an Mayring

- 3 Oberkategorien: Verhalten, Begriffe, Weiterführender/Spezieller
- Beispiel "OK2 Begriffe"

ID	Kategoriebezeichnung	Definition	Ankerbeispiel
OK 2	Begriffe		
UK 2.1	benutzt mystische Begriffe	Aussagen, in denen Begriffe auftreten, die vermenschlichen oder eine höhere Gewalt hinzuziehen	„Es hat Angst, dass es kaputt geht und weiß, dass es weglaufen soll", Interview-Nr. 61
UK 2.2	benutzt nicht-mystische, aber nicht-technische Begriffe	Aussagen, in denen Begriffe aus der Biologie oder Analogien mit dem menschlichen Körperbau, z.B. Augen, beinhaltet sind	„Die Augen reagieren auf die Hand", Interview-Nr. 64
UK 2.3	benutzt technische Begriffe	Aussagen, in denen Begriffe, die für ein technisches System angemessen sind, z.B. Räder, Sensoren, beinhaltet sind	„Da ist ein Sensor drin, der Bewegung wahrnimmt", Interview-Nr. 82

Tabelle: Interview-Auswertung (Ausschnitt)

OK 2	Begriffe		
UK 2.1	benutzt mystische Begriffe	Aussagen, in denen Begriffe auftreten, die vermenschlichen oder eine höhere Gewalt hinzuziehen	„Es hat Angst, dass es kaputt geht und weiß, dass es weglaufen soll“, Interview-Nr. 61

Nr.	UK	1.1 R	1.2 M	2.1	2.2	2.3	3.1	3.2	3.3	3.4
61	* "Es reagiert, wenn man versucht dranzugehen" * Kann reproduzieren (hält Hand zunächst an die falsche Seite) * "Liegt an den Rädern, die laufen nur, wenn man die Hand an den Sensor hält" * "Es hat Angst, dass es kaputt geht und weiß, dass es weglaufen soll"	x	x	x	x	x	x	-	-	-

Ergebnisse der Interviewauswertung

Beschreibung des Systems

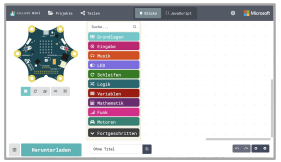
- Oft Analogien aus Biologie oder Technik allgemein
- Cubelets werden als handelnd beschrieben:
  Sie denken oder erschrecken sich
- Fazit: Passender Wortschatz fehlt

Antwort auf die “Warum?”-Frage

- “so programmiert”, “so eingebaut”
- Nur kurz in Schlagworten
- Bei Nachfrage hatte kein Kind eine konkrete Vorstellung, was Programmieren ist.

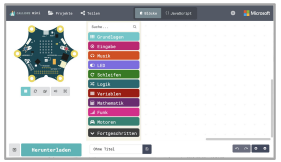
Klasse	Schule	#Interviews
1	1	17
2	1	17
3	1	21
4	2	18
5	3	17
6	4	15
7	4	17
8	5	11
		133

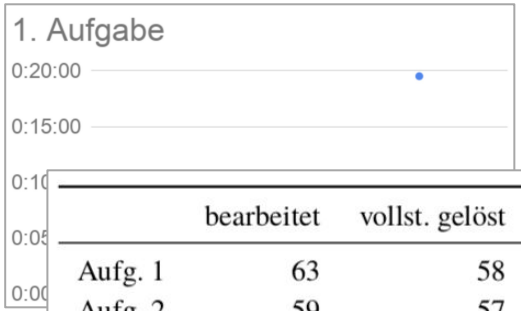
Tabelle: Screencast-Auswertung (Ausschnitt)

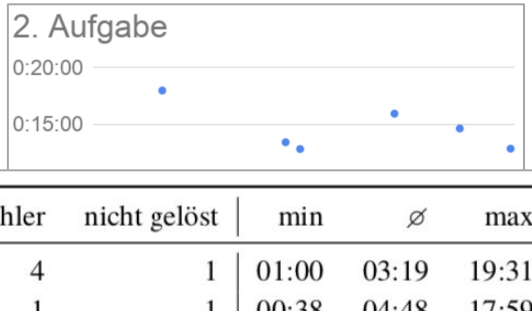


		Reproduktion				Transfer				Rätsel I				Rätsel II			
Kl.	Video	Start	Stop		**Dauer**	Start	Stop		**Dauer**	Start	Stop		**Dauer**	Start	Stop		**Dauer**
6	3	0:01:30	0:03:21	2	**0:01:51**	00:04:19	0:09:26	2	**0:05:07**	0:11:16	0:15:56	2	**0:04:40**	0:18:33	00:53:00	0	**0:34:27**
6	5	00:02:30	00:03:56	2	**0:01:26**	00:05:11	00:05:49	2	**0:00:38**	00:08:03	00:10:53	1	**0:02:50**	00:14:21	00:50:50	2	**0:36:29**
6	6	00:00:58	00:03:56	2	**0:02:58**	00:04:02	00:06:13	2	**0:02:11**	00:08:17	00:26:36	2	**0:18:19**	00:29:20	00:55:36	0	**0:26:16**
6	9	0:00:59	0:09:52	1	**0:08:53**	0:10:22	0:11:57	2	**0:01:35**	0:22:13	0:29:36	2	**0:07:23**	0:36:08	0:56:00	0	**0:19:52**

Ergebnisse der Screencast-Auswertung



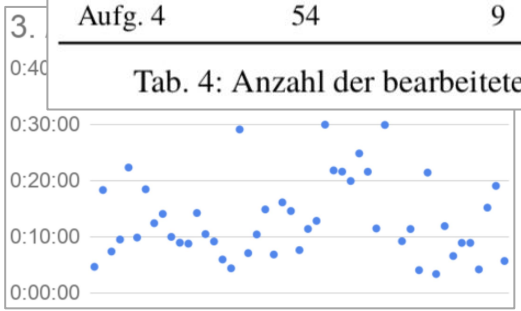




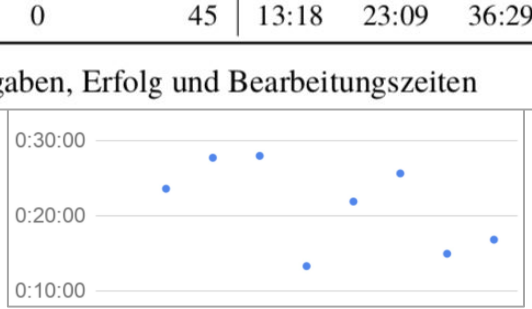
	bearbeitet	vollst. gelöst	kl. Fehler	nicht gelöst	min	⌀	max
Aufg. 1	63	58	4	1	01:00	03:19	19:31
Aufg. 2	59	57	1	1	00:38	04:48	17:59
Aufg. 3	62	47	10	5	03:23	13:38	37:12
Aufg. 4	54	9	0	45	13:18	23:09	36:29

Tab. 4: Anzahl der bearbeiteten Aufgaben, Erfolg und Bearbeitungszeiten

Re



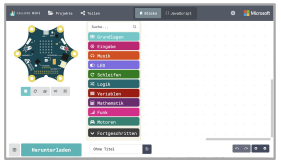
Leichtes Rätsel



Schweres Rätsel

Klasse	Schule	#Screencasts
1	1	8
2	1	11
3	1	9
4	2	8
5	3	8
6	4	8
7	4	9
8	5	4
		65

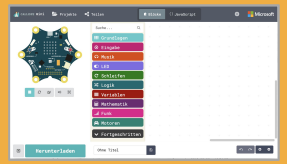
Beobachtungen während des Programmierens



- Kleine Flüchtigkeitsfehler
- Wenige technische Schwierigkeiten
- Manchmal zurückhaltendes Erkunden
- Selten falsch verstandene Semantik

- Alles in allem erwies sich der Makecode-Editor jedoch als sehr kindgerecht.

- Zur Reduktion der Komplexität wäre ein Ausblenden nicht benötigter Kategorien und Bausteine sinnvoll.



Antwort auf Forschungsfrage A

Programmieren

A1 Programmier-Bausteine und Sensoren/Aktoren wurden zielgerichtet genutzt.

A2 Probleme gab es beim genauen Beobachten und teilweise bei der Benutzung des Editors.

Sehr gute Ergebnisse bei den gestellten Aufgaben und auch beim freien Programmieren. Aufgaben konnten gelöst, eigene Ideen umgesetzt werden.

Antwort auf Forschungsfrage B

Vorwissen

B1 Benutzt wurden oft technische oder biologische Begriffe.
Es fehlte der passende Wortschatz.

B2 Der funktionale Zusammenhang zwischen “Hand vor den Cubelets” und “Cubelets fahren weg” konnte erstellt werden.

B3 Eine präzise Erklärung für das Verhalten gab es kaum.

Fazit

- Schülern der 4. Klasse fällt es schwer ein Informatiksystem adäquat zu beschreiben und seine Funktion zu erklären.
- Jedoch können sie einfache Beispiele für das EVA-Prinzip verstehen und in konkreten Informatiksystemen implementieren.
- Calliope mini und der Makecode-Editor sind dafür geeignete Systeme.
- Bemerkenswert war das lange, konzentrierte Arbeiten an den Aufgaben sowie Kreativität und Begeisterung beim freien Programmieren.

Was hat sich im letzten Jahr getan?

In Deutschland, z.B.:

- Viel neues Material zu Calliope mini
- Weitere Projekte und Veröffentlichungen nicht nur zu Calliope mini
- “Kompetenzen zur informatischen Bildung im Primarbereich” der GI
- Informatik findet vermehrt den Weg in Lehrpläne

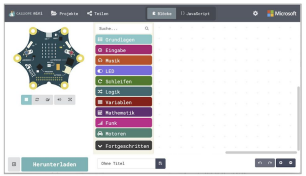
Im Saarland, z.B.:

- Tag des Informatikunterrichts (immer Aschermittwoch)
- Basiscurriculum “Medienbildung und informatische Bildung”
- Seminar “Informatische Bildung im Primarbereich”
- Aufbau Schülerlabor InfoLab Saar

Unterrichtseinheit

Drei Doppelstunden:

1	1	Interviews
	2	
2	3	Calliope ohne und mit Strom erkunden
	4	
3	5	Test: Programmieren
	6	Freies Programmieren



Ergebnisse der Interviewauswertung

Beschreibung des Systems

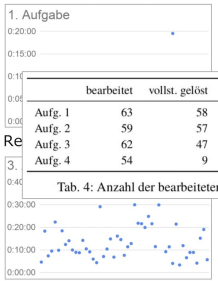
- Oft Analogien aus Biologie oder Technik allgemein
- Cubelets werden als handelnd beschrieben:
  Sie denken oder erschrecken sich
- Fazit: Passender Wortschatz fehlt

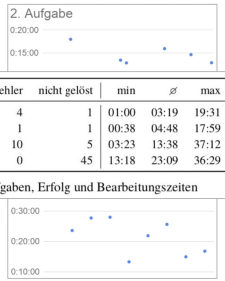
Antwort auf die "Warum?"-Frage

- "so programmiert", "so eingebaut"
- Nur kurz in Schlagworten
- Bei Nachfrage hatte kein Kind eine konkrete Vorstellung, was Programmieren ist.

Klasse	Schule	#Interviews
1	1	17
2	1	17
3	1	21
4	2	18
5	3	17
6	4	15
7	4	17
8	5	11
		133

Ergebnisse der Screencast-Auswertung





	bearbeitet	vollst. gelöst	kl. Fehler	nicht gelöst	min	⌀	max
Aufg. 1	63	58	4	1	01:00	03:19	19:31
Aufg. 2	59	57	1	1	00:38	04:48	17:59
Aufg. 3	62	47	10	5	03:23	13:38	37:12
Aufg. 4	54	9	0	45	13:18	23:09	36:29

Tab. 4: Anzahl der bearbeiteten Aufgaben, Erfolg und Bearbeitungszeiten

Leichtes Rätsel

Schweres Rätsel

Klasse	Schule	#Screencasts
1	1	8
2	1	11
3	1	9
4	2	8
5	3	8
6	4	8
7	4	9
8	5	4
		65

Fazit

- Schülern der 4. Klasse fällt es schwer ein Informatiksystem adäquat zu beschreiben und seine Funktion zu erklären.
- Jedoch können sie einfache Beispiele für das EVA-Prinzip verstehen und in konkreten Informatiksystemen implementieren.
- Calliope mini und der Makecode-Editor sind dafür geeignete Systeme.
- Bemerkenswert war das lange, konzentrierte Arbeiten an den Aufgaben sowie Kreativität und Begeisterung beim freien Programmieren.