LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

LFE Medieninformatik • Raluca Vlad

Evaluation visueller Programmierungstools für Tangible Interaction

Projektarbeit SoSe 2010

Gliederung

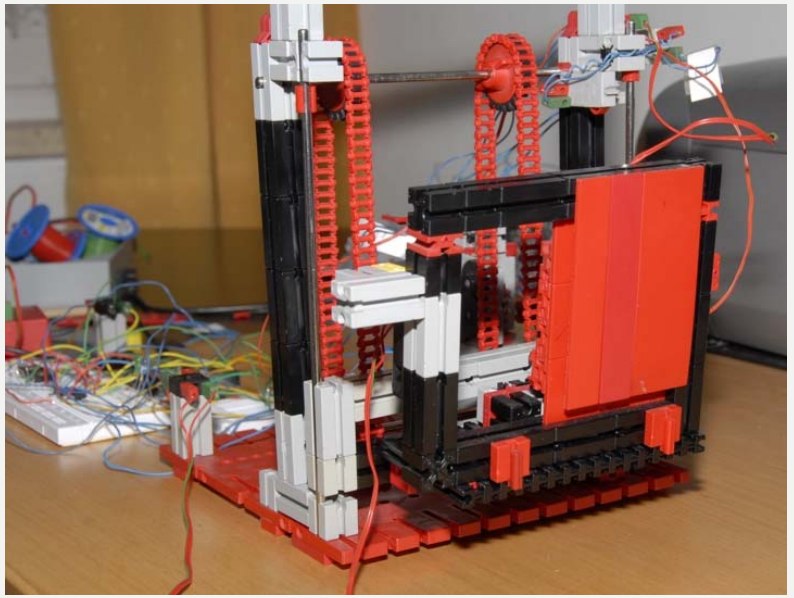
- Aufbau der Projektarbeit
- Evaluierte VPLs
- Vergleichskriterien und Ergebnisse
- Fazit
- Demonstration

Aufbau der Projektarbeit

- Bau eines Aufzugmodells mit Aktoren, Sensoren und Controllern
- Programmierung und Steuerung des Prototypen mit VPLs
- Evaluation anhand von neun Kriterien
- Satz von Vorschlägen zur Implementierung einer VPL

Aufzugsmodell

- Liftkabine mit beweglicher Tür: „Tür auf“ und „Tür zu“
- zwei verschiedene Stockwerke: „Lift oben“ und „Lift unten“
- zwei Wunschknöpfe: „Lift hoch“ und „Lift runter“



Aufzug

Evaluierte VPLs

- **ROBO Pro**
- **Physical Etoys**
- **Scratch**

Evaluationskriterien
Setup des Systems

1. Aufwand für die Inbetriebnahme des Systems

2. Eingesetzte Hardware zum Bau des Prototypen

Evaluationskriterien Programmieroberfläche

3. Eingesetztes Programmierparadigma
4. Eingesetze visuelle Notation
5. Manipulation der Programmierbausteine
6. Möglichkeit der Annotation
7. Erweiterbarkeit durch textuellen Programmcode

Evaluationskriterien Programmierung und Testen

8. Erreichbarkeit an Komplexität

9. Testen der programmierten Hardware

Erstes Kriterium: Aufwand für die Inbetriebnahme des Systems

- Scratch: aufwändige Installation mit Einbinden von Bibliotheken und Drittsoftware
- ROBO PRO: einfache Führung durch die Installation

Zweites Kriterium
Eingesetzte Hardware zum Bau des Prototypen

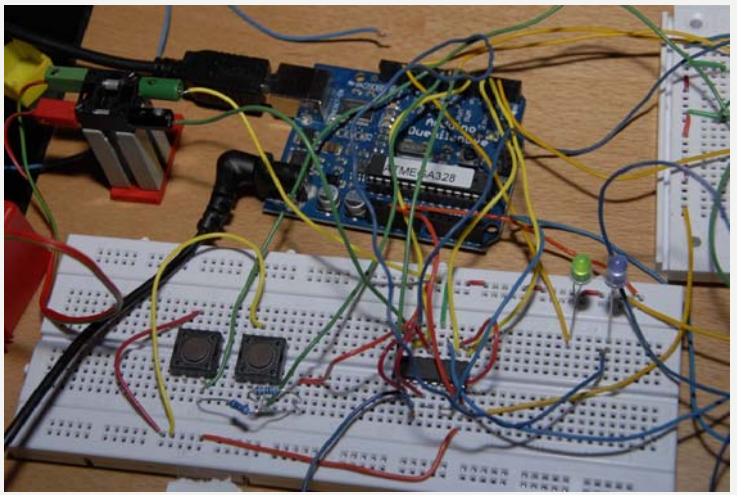
- **Motoren** zur Bewegung der Tür und der Liftkabine
- **Push Buttons** als Endanschlagsensoren für „Tür zu“, „Tür offen“, „Lift unten“, „Lift oben“
- **LEDs** zur Betriebsanzeige

Zweites Kriterium
Eingesetzte Hardware zum Bau des Prototypen

• **Controller** zur Steuerung der Hardwarekomponenten

ROBO TX
- wenig Elektronikwissen

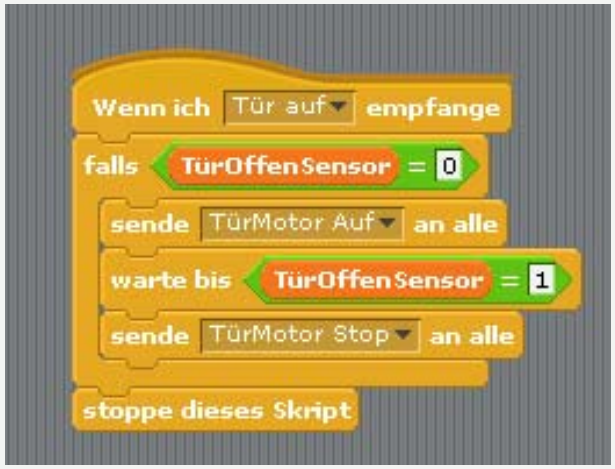
Arduino Duemilanove
- höherer Schaltungsaufwand



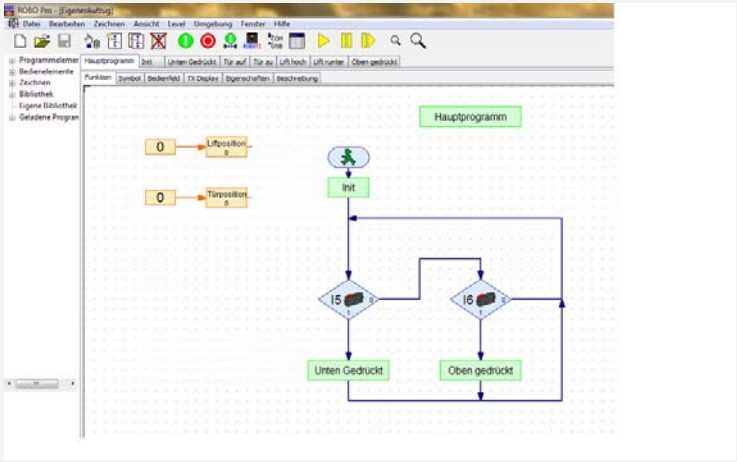
Arduino Beschaltung

Drittes Kriterium
Eingesetztes Programmierparadigma

- ROBO PRO: Paradigmen zur Darstellung von Kontrollfluss und Datenfluss
- Scratch und Etoys: Paradigmen, die auf Regeln basieren



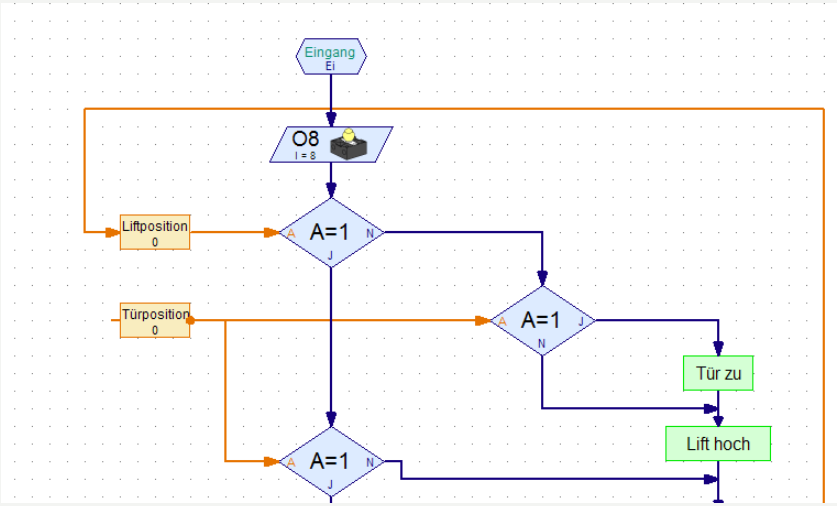
Scratch



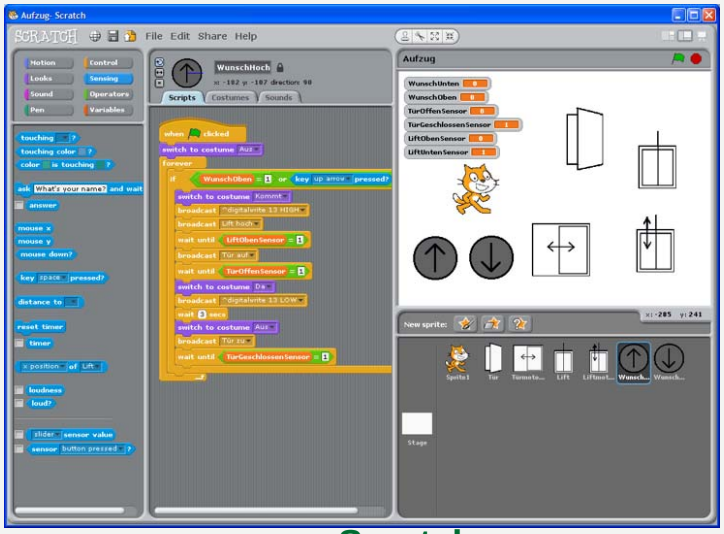
ROBO PRO

Viertes Kriterium
Eingesetze visuelle Notation

- Etoys und ROBO PRO: Symbole
- ROBO PRO: Kontrollflussdiagramm und farbliche Darstellung des Datenflusses
- Etoys und Scratch: Textbausteine, die Befehle repräsentieren



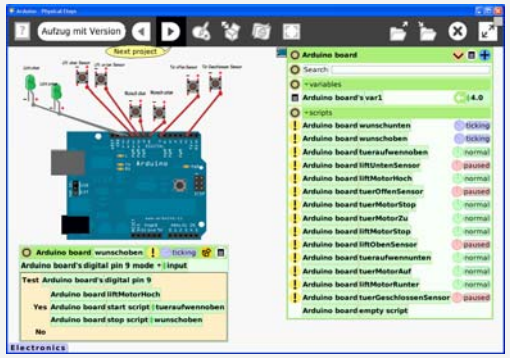
ROBO PRO

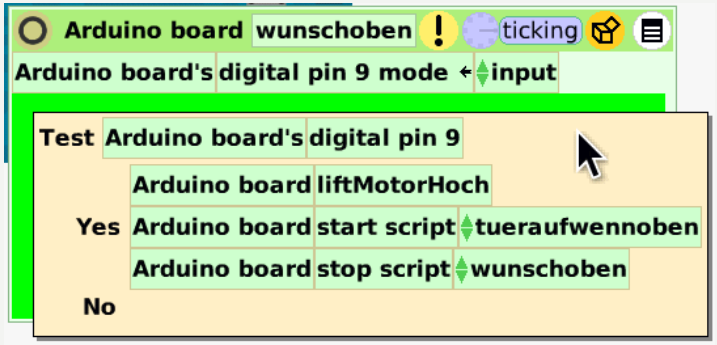


Scratch

Fünftes Kriterium
Manipulation der Programmierbausteine

- Alle: Objekte mit der Maus auf der Oberfläche bewegen
- ROBO PRO: Objekte mit Pfeilen verbinden
- Scratch: Einrasten wie Puzzle-Teile
- Etoys: Kacheln, die man auf farblich hervorgehobene Bereiche einfügt
- Alle: Bausteine lassen sich aus der Toolbox auf die Fläche ziehen

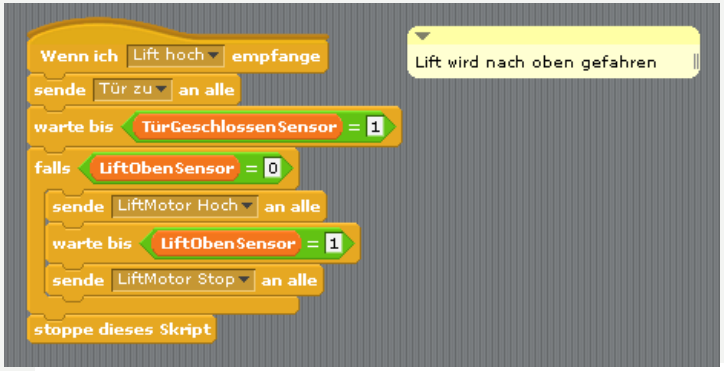




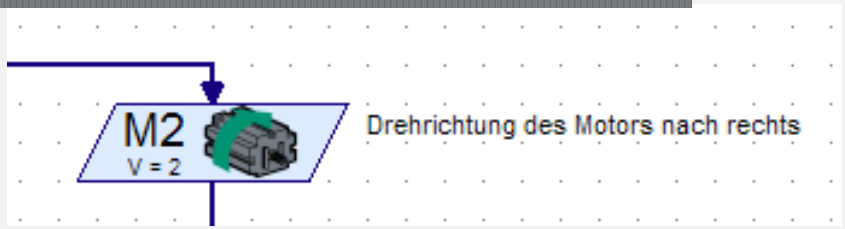
Physical Etoys

Sechstes Kriterium
Möglichkeit der Annotation

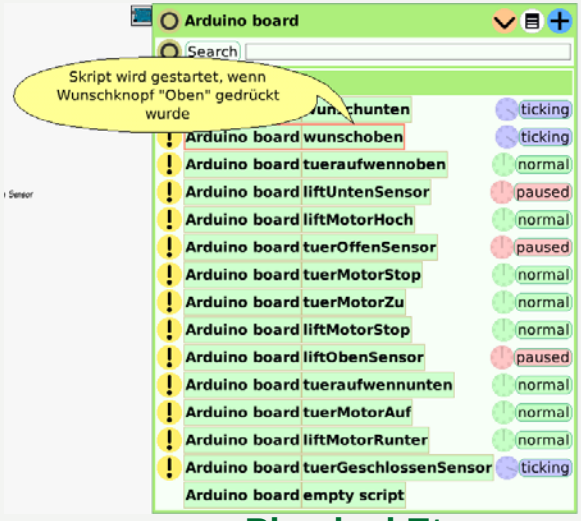
- ROBO PRO und Scratch: freie Textbausteine auf dem Hintergrund positioniert
- Etoys: editierbare „Balloon helps“



Scratch



ROBO PRO



Physical Etoys

Siebtes Kriterium
Erweiterbarkeit durch textuellen Programmcode

- Etoys: Textcodeerweiterung über „Reveal this Script“ Anzeige
- Scratch: grafische Textbausteine für Schleifen, Variablen und Abbruchbedingungen
- ROBO PRO: kein Textcode, dafür mächtigste Symbolsprache

Achtes Kriterium
Erreichbarkeit an Komplexität

- ROBO PRO: Komplexitätsstufen über „Levels“ erreichbar und Aufruf von Unterprogrammen
- Etoys: Schleifen mit Abbruchbedingung nicht möglich
- Scratch: Aufrufe von Skripten über „Broadcast“-Befehl

Neuntes Kriterium
Testen der programmierten Hardware

- ROBO PRO: Code auf den Controller laden und System kann autark laufen, Testen nur über Debug-Modus
- Etoys und Scratch: Code läuft nur auf auf der Programmieroberfläche und ermöglicht direktes Testen

Fazit

Wichtigste Punkte:

- Einfache und einheitliche Installation des Systems
- Fehlertolerante Hardware
- passende visuelle Unterstützung
- Komplexität basierend auf Levels
- Direktes Testen der Hardwarekomponenten

Vielen Dank für Ihre Aufmerksamkeit!

Demonstration

Referenzen

[1] Arduino. http://www.arduino.cc/, 2010. visited 30.04.2010.
[2] Catenary. http://scratchconnections.wik.is/User:Chalkmarrow/Catenary, 2010. visited 30.04.2010.
[3] Etoys. http://www.squeakland.org/, 2010. visited 30.04.2010.
[4] Fischertechnik. http://www.fischertechnik.de, 2010. visited 30.04.2010.
[5] Physicaletoys.http://tecnodacta.com.ar/gira/projects/physical-etoys/,2010. visited 30.04.2010.
[6] Scratch. http://scratch.mit.edu/, 2010. visited 30.04.2010.
[7] Processing. http://processing.org/download/, 2010. visited 30.04.2010.
[8] Processing library. http://www.arduino.cc/playground/Interfacing/Processing, 2010. visited 30.04.2010.