

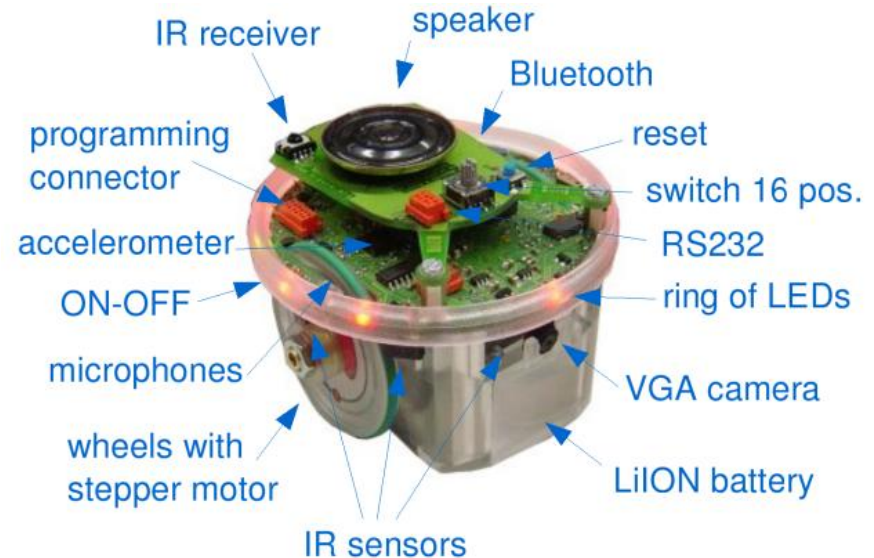
Eingebettete System Projektarbeit

ePuck Steuerung via Bluetooth

*Florian Grützmacher,
Simeon Wiedenmann*

ePuck Roboter

- mobiler Roboter für Lehre
- open hardware
- onboard software open source



Technische Details:

- | | | |
|--------------------------|--------------------|----------------------|
| • 70 mm Durchmesser | • Bluetooth | • 640x480 VGA Kamera |
| • 50 mm Höhe | • Infrarot | • 3D accelerometers |
| • 200g Gewicht | • 8 KB RAM | • 3 Mikrofone |
| • 12 cm/s Speed | • 114 KB Flash | • 1 Lautsprecher |
| • 2h autonome Versorgung | • 2 Schrittmotoren | |

ePuck Projektaufgabe

Aufgabe:

- Aufbau einer Bluetooth RFCOMM Verbindung zum ePuck
- Steuerung des ePucks über serielle Schnittstelle anhand Herstelldokumentation
- Userinterface zur Steuerung des ePucks

Lösung:

- klassisches Client-Server Modell
 - Client: Notebook/ Android-Gerät
 - Server: ePuck
- 2 Steuermöglichkeiten
 - USB Controller
 - Android-Gerät



ePuck Projekt

Lösungsdetails:

// request socket

- `s = socket(AF_BLUETOOTH, SOCK_STREAM, BTPROTO_RFCOMM);`
 - Adressfamilie: `AF_BLUETOOTH`
 - Sockettype: `SOCK_STREAM`
 - Protocol: `BTPROTO_RFCOMM`
- Quellcode in drei Funktionen
 - `main`
 - `eventloop`
 - `Joystick_Info`
- Verteilte Versionsverwaltung mittels GIT
- GitHub – Webbasierter Hosting-Dienst für Software-Entwicklungsprojekte

ePuck Client: main

- bt_destination_address festlegen & Format checken
- Socket erstellen
- connect 2 Server & send messages
 - initial dummies
 - Standartgeschwindigkeit
 - 1 sec LEDs blinken lassen
 - Flashlight anmachen
- polling Schleife für USB Controller Befehle
- Verbindung beenden per JoyStick
 - Rädergeschwindigkeit = 0
 - Licht aus
 - Sound aus
 - LED's aus
- Ausgabe „bye“
- Speicher freigeben
- Bluetoothverbindung beenden: close(s)

ePuck Client: eventloop

Zum Polling der JoyStick Befehle

- Sounds abspielen
- Variablen setzen für:
 - Geschwindigkeit Rad Links
 - Geschwindigkeit Rad Rechts
 - LED Status
 - Flashlight Status
 - Gang
- Räder anhalten
 - ButtonUp – Event
- neue Status senden
 - Geschwindigkeit der Räder
 - LED Zustände
 - Flashlight Zustand

ePuck Client: Joystick_Info

Allgemeine Infos und Verbindung zu Joystick

- Suche nach Joysticks
- gibt Anzahl gefundener Joysticks aus
- ggf. Fehlermeldung
"Kann Joystick %d nicht öffnen"
- Ausgabe /* Informationen zum Joystick */

```
printf ("Joystick %d\n", i);  
printf ("\tName:    %s\n", SDL_JoystickName(i));  
printf ("\tAxen:    %i\n", SDL_JoystickNumAxes(js));  
printf ("\tTrackballs: %i\n", SDL_JoystickNumBalls(js));  
printf ("\tButtons:  %i\n",SDL_JoystickNumButtons(js));
```

ePuck Befehle

- d,X,Y //drive, left wheel with speed X, right wheel with speed Y
 //Maximum for X,Y = 1000

- f,1 //turning on the flashlight
- f,0 //turning of the flashlight

- l,X,1 //turning on LED number X $1 \leq X \leq 8$
- l,X,0 //turning off LED number X

- l,9,1 //turning on all LEDs
- l,9,0 //turning off all LEDs

- t,0 //stop sounds
- t,X //plays sound X, for $1 \leq X \leq 5$

ePuck Steuerung



DEMO