

Dokumentation zu Pacman Reloaded

Jens Wöhrle <jens.woehrle@tum.de> 22.06.2017

Inhalt

Einleitung.....	1
Einrichten und Loslegen	1
Spielsteuerung.....	2
Datenstruktur	3
Konfigurationsparameter:	3
Verkabelung:	3

Einleitung

Die Grundlegende Idee ist es, dass Projekt von Philipp Riedmann, Michael Schreier und Tabea Schmidt als Vorlage zu verwenden und zu optimieren. Die ursprüngliche Idee ist es, das Spielerlebnis durch eine interaktive Schnittstelle zu verbessern. Als Spielkontroller zur Steuerung wird ein Kuscheltier verwendet. Durch einen Gyrosensor wird der Pac-Man gesteuert. Das Auge des Monstter AG Monsters enthält einen Neofruit-Ring, der die Position der Geister, die im Spiel unsichtbar werden anzeigt. Ist ein Geist sehr nah, so fängt das Kuscheltier an zu vibrieren. Außen herum gibt es einen größeren Neufruit-Ring, der die umliegenden Wände des Pacmans anzeigt. Da im Spiel auch der Pacman unsichtbar werden kann, so ist man dann vollkommen auf die Orientierung auf Basis von diesem ring angewiesen.

Diese Dokumentation gibt einen Überblick über das gesamte Projekt, sodass man sich schnell wieder zurechtfinden kann, um zu spielen oder daran weiterzuarbeiten.

Einrichten und Loslegen

Folgende Anleitung beschreibt das Schrittweise vorgehen, um anzufangen:

Go to correct directory	pacman\NodeServer\
Install dependencies	<i>npm i</i>
Plug-In Controller	
Run Node with	<i>node index.js <SerialPort> e.g. „node index.js COM5“</i>
Play on	<i>http://localhost:3001</i>

Spielsteuerung

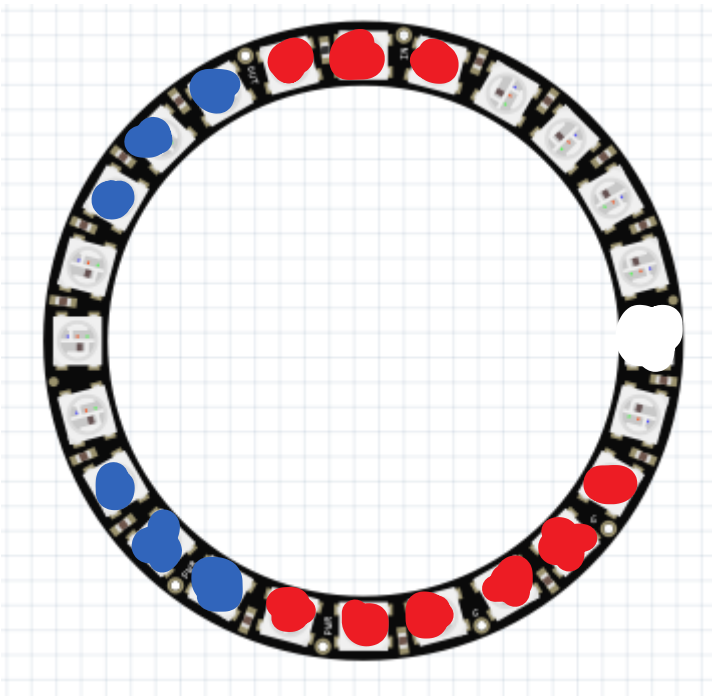
Neigen steuert de Pacman. Die Sichtbarkeit kann mit folgenden tasten geändert werden:

Sichtbarkeit Geist	Taste S
Sichtbarkeit Pacman	Taste A

Der kleine Kreis zeigt an, wo sich relativ vom Pacman die Geister befinden. Die Nähe der Geister ist durch die Vibration erkennbar. In dieser Illustration wäre der gelbe Geist über, der blaue rechts von und der grüne Geist links unten von dem Pacman:



Der große Kreis zeigt die umliegenden Wände, sowie die Bewegungsrichtung des Pacman an:



Das weiße Led, sowie die blauen LED's zeigen die Bewegungsrichtung des Pacman an. In diesem Fall geht also der Pacman nach rechts. Die roten LED's zeigen an, wo sich Wände befinden. In diesem expliziten Fall würde das bedeuten, dass der Pacman weder nach unten noch nach oben gehen kann. Weil oben rechts jedoch nichts leuchtet, bedeutet dies, dass der Pacman demnächst (in etwa 1 Sekunde) nach oben gehen kann. Weil es unten rechts rot leuchtet, kann der Pacman nicht in Kürze

nach unten gehen. Da es rechts nicht rot leuchtet, läuft der Pacman aktuell weiter nach rechts. Links leuchtet es nicht, also könnte sich der Pacman auch um 180 Grad drehen und nach links gehen.

Es sei zu beachten, dass wir den Ring lediglich in 8 Gruppen (Oben, Unten, ..., Links-Oben, Links-Unten, ...) aufgeteilt haben.

Datenstruktur

In folgender Tabelle ist die Datenstruktur genau spezifiziert.

.\game_original	Altes Javascript Standardspiel
.\Sketches	Beispielcode der einzelnen Sensoren
.\pacman	Controller code, sowie Node Code
.\Evaluation	Statistische Auswertung

Konfigurationsparameter:

Das Spielerlebnis kann mit verschiedenen Parametern angepasst werden:

Anzahl Spieler:	.\NodeServer\public\js\ghost.js::35
Kleiner Ring Ausrichtung:	.\Arduino\controller\controller.ino::107
Großer Ring Ausrichtung:	.\Arduino\controller\controller.ino::114
Kalibrieren der Nullausrichtung MPU:	.\Arduino\controller\controller.ino::122-124
Empfindlichkeit in Grad MPU:	.\Arduino\controller\controller.ino::120-121

Der Wert der Empfindlichkeit entspricht dem Winkel in Grad um den der Controller geneigt werden muss, damit die neue Laufrichtung des Pacman erkannt wird.

In Arduino\controller\controller.ino::2 sowie in Zeile 5 kann ein Debugging Modus eingestellt werden, sodass das genaue Controllerverhalten über den Seriellen Ausgabenmonitor verfolgt werden kann. Das Makro „Arduino-Calibration“ dient dazu, die Ausrichtung der LEDs einzustellen.

Verkabelung:

Folgendes Bild zeigt die Verkabelung des Controllers:

