Der GPIO und Scratch

Von Scratch aus lassen sich Elemente wie Leuchtdioden, Summer und Schalter am GPIO steuern. Je nach Version von Scratch muss aber unterschiedlich vorgegangen werden. Auch der mögliche Funktionsumfang unterscheidet sich etwas. Daher wird hier zuerst das Vorgehen in Scratch 1.4 erklärt und anschließend in Scratch 2.

1 Scratch 1.4 und der GPIO

Für die Scratch-Version 1.4 gibt es eine spezielle Anpassung für den Raspberry Pi. Bei dieser gibt es über den Menüpunkt »Bearbeiten« die Möglichkeit den GPIO-Server zu starten. Erst dann zeigen alle nachfolgenden Anweisungen erst ihre Wirkung.

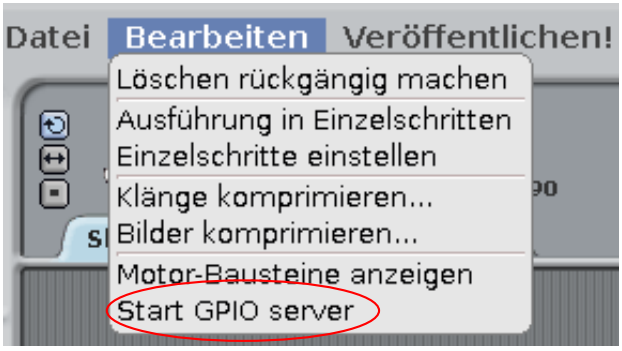


Abbildung 1: Starten des GPIO-Servers in Scratch

1.1 Ausgänge

Wenn man an den GPIO ein Element angeschlossen hat, dass man von Scratch aus schalten will, so muss zuerst angegeben werden, an welchem Pin etwas angeschlossen ist. Dazu nutzt man die »Senden«-Funktion von Scratch und sendet `config<PIN>out` an alle, wie in Abbildung 2a zu sehen. Hierbei wird PIN durch die Nummer des Pins ersetzt. In Scratch werden dabei die Pins mit ihrer GPIO-Nummer angesprochen.

Das Setzen der Pins muss nur ein Mal geschehen. Anschließend kann man über die »Senden«-Funktion mit `gpio<PIN>on` den Pin anschalten und mit `gpio<PIN>off` ausschalten. Ein kleines Beispiel dafür ist in Abbildung 2b dargestellt.

1.2 Eingänge

Mit dem Senden Genauso von `config<PIN>in` wird, wie bei den Ausgängen, Scratch die Eingänge mitgeteilt. Daraufhin wird der Pin als Sensor aufgeführt. Dieser Sensor liefert



(a) Als Ausgang setzen



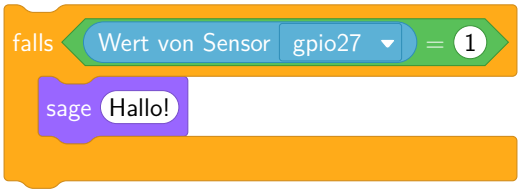
(b) Für 3 Sekunden einschalten

Abbildung 2: Pin 22 als Ausgang

dann die Werte 0 oder 1, je nachdem ob der an den Pin angeschlossene Schalter offen oder geschlossen ist.



(a) Als Eingang setzen



(b) Auf Sensor reagieren

Abbildung 3: Nutzen eines GPIO-Eingang an Pin 27

1.3 Pulsweitenmodulation

Die Ausgänge am GPIO lassen sich nur ein- und ausschalten, was zur Folge hat, dass keine Zwischenwerte möglich sind. Dieses kann man etwas dadurch umgehen, dass die Ausgänge immer nur kurz einschaltet. So etwas wird durch die sogenannte Pulsweitenmodulation erreicht, die man bei einem Eingang durch das Senden von `config<PIN>outputpwm` einstellen kann.

Dann lässt sich dem Pin ein Wert zwischen 0 und 1024 zuordnen, der angibt, wie häufig der Pin eingeschaltet sein soll. So sendet man `gpio<PIN>pwm<WERT>` um einen entsprechenden Wer zu setzen. Das Ausschalten funktioniert analog zu den Ausgabenpins mit `gpio<PIN>off`.

2 Scratch 2 und der GPIO

Die Version 2 von Scratch ist in vielen Elementen überarbeitet worden und bietet nun bei den Kategorien der Blöcke »Weitere Blöcke« an. Damit gibt es die Möglichkeit eigene Blöcke zu erstellen oder Erweiterungen hinzuzufügen, die neue Blöcke mitbringen. Beim Raspberry Pi lässt sich dort die Erweiterung »GPIO« laden, die zur weiteren Arbeit nötig ist.



(a) Konfigurieren



(b) Für 3 Sekunden 50% und anschließend 70%

Abbildung 4: Pin 24 mit Pulsweitenmodulation

2.1 Ausgänge

Gegenüber der der Version 1.4 muss man hier nicht vorher angeben, dass ein Pin als Ausgang genutzt werden soll. Es genügt ihn direkt auf `output high` für ein und `output low` für aus zu stellen.

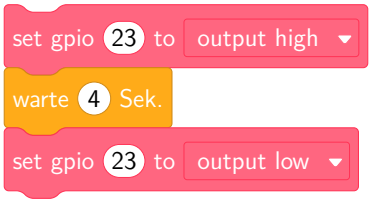
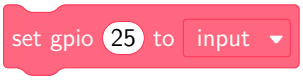


Abbildung 5: Pin 23 für 4 Sekunden einschalten

2.2 Eingänge

Um einen Pin als Eingang zu nutzten, muss dieses auch in Version 2 zuvor mitgeteilt werden, indem man den Pin auf `input` setzt. Anschließend kann man über einen Abfrageblock überprüfen, ob am Eingang ein Signal anliegt.



(a) Als Eingang setzen



(b) Auf Sensor reagieren

Abbildung 6: Nutzen eines GPIO-Eingang an Pin 25

2.3 Weitere Möglichkeiten mit Scratch 2

Die Vorinstallation von Scratch 2 hat auf dem Raspberry Pi auch die Steuerung für den Sense-Hat Aufsatz mit dabei. Die Pulsweitenmodulation wie bei Scratch 1.4 ist aber nicht ohne weiteres möglich. Dafür gibt es aber passende Erweiterung im Internet zu finden, die man erst auf den Raspberry Pi installieren muss. Darüber können auch weitere Möglichkeiten in Scratch eingebunden werden.