

6 Zufälle mit dem senseBox-Orakel

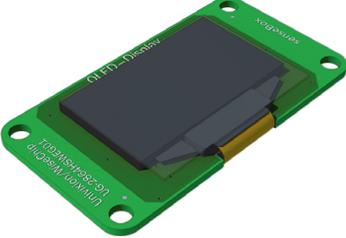
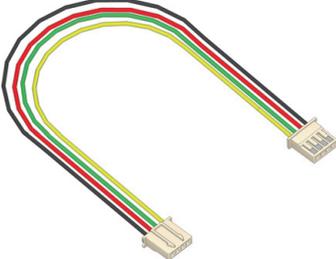


Abb. 6-1 Die senseBox als Orakel

6.1 Projektbeschreibung

Zufall wird häufiger beim Programmieren verwendet, als man glaubt. Gerade im spielerischen Bereich ist es oft nötig, Zufall zu simulieren. Es wäre doch relativ langweilig, wenn ein Programm immer die gleiche Ausgabe besitzt und zum Beispiel Charaktere in einem Computerspiel immer die gleichen Wege laufen. In diesem Kapitel lernst du die Grundlagen des Zufalls beim Programmieren kennen und simulierst einen Würfel. Des Weiteren wirst du ein Orakel erstellen, das in der Lage sein wird, alle deine Lebensfragen zu beantworten. Du kannst es zum Beispiel vor deinem nächsten Examen befragen, ob du die Prüfung bestehen wirst oder nicht ;-).

6.2 Benötigte Bauteile

OLED-Display		1×
JST-JST-Kabel		1×

6.3 Aufbau der Schaltung

Für die folgenden zwei Projekte benötigst du keinen großen Aufbau. Da bereits ein Button auf der senseBox MCU eingebaut ist, brauchst du nur noch eine Ausgabe, um Informationen anzuzeigen. Dazu schließt du das OLED-Display mithilfe eines JST-JST-Kabels an einen der I2C/Wire Ports.

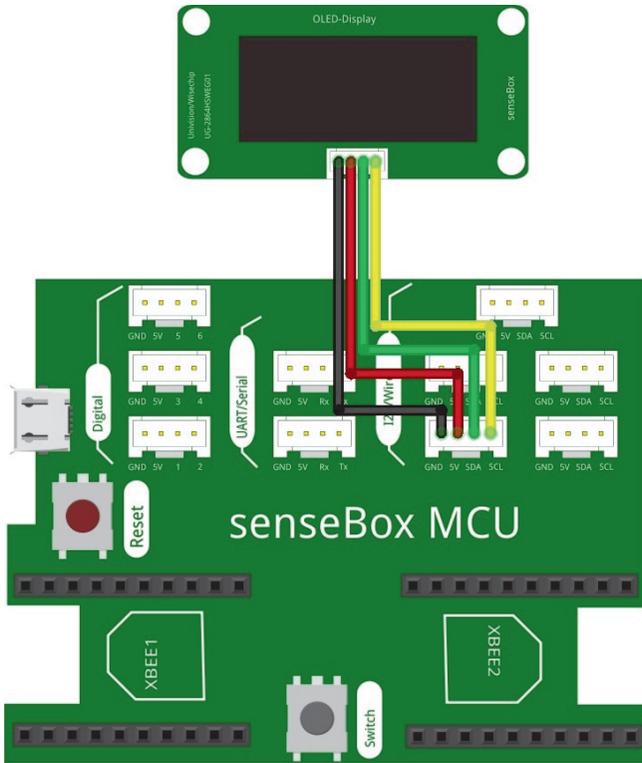


Abb. 6-2 Aufbau für das senseBox-Orakel

6.4 Würfelspiele

Um eine Zufallszahl mit Blockly zu generieren, kannst du den Block ganzzahliger Zufallswert aus der Kategorie Mathematik verwenden. Bei diesem Block kannst du eine obere und untere Grenze wählen. Bei jedem Aufruf wird dann eine zufällige ganze Zahl aus diesem Wertebereich generiert. Natürlich kannst du auch Dezimalzahlen zufällig generieren lassen, jedoch werden diese meist nur für kompliziertere mathematische Formeln benutzt. In diesem Kapitel ist es deshalb völlig ausreichend, eine ganze Zahl zufällig zu generieren.



Abb. 6-3 Dieser Block generiert einen ganzzahligen Zufallswert zwischen 1 und 100.

Um nun einen normalen Würfel zu simulieren, solltest du einen Wertebereich von eins bis sechs auswählen. Natürlich gibt es auch Spezialwürfel, die mehr als sechs Seiten besitzen. Diese kannst du simulieren, indem du den Wertebereich entsprechend erhöhst. Damit du nun auch sehen kannst, welche Zufallszahl ausgewählt wurde, musst du diesen Wert auf dem Display anzeigen lassen. Dazu verwendest du das OLED-Display. Die Blöcke dafür solltest du bereits alle kennengelernt haben, ansonsten kannst du noch einmal in Kapitel 3 nachlesen, wie genau das OLED-Display funktioniert.

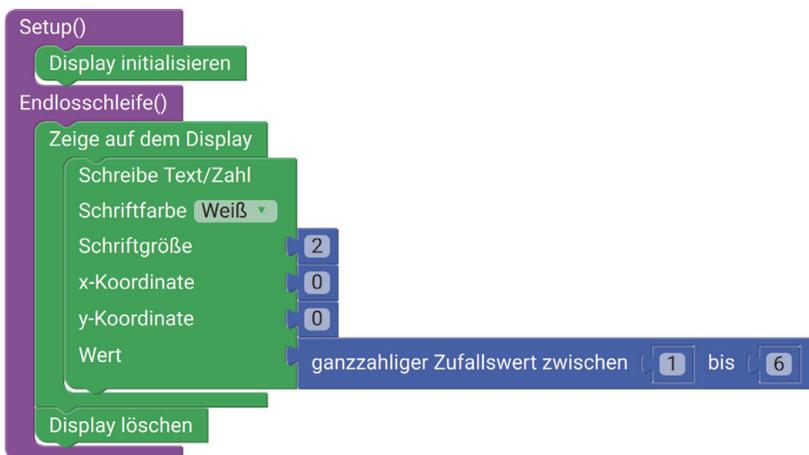


Abb. 6-4 Zufallswerte auf dem Display anzeigen

Nun kannst du bereits den ersten Test durchführen. Du wirst bemerken, dass zwar nun Zufallszahlen generiert werden, diese sich jedoch sehr schnell ändern. Das ist natürlich nicht das Verhalten eines richtigen Würfels. Es soll so lange die gleiche Zahl angezeigt werden, bis eine Aktion ausgeführt wird, wodurch eine neue zufällige Zahl angezeigt wird. Für diese Aktion eignet sich der graue Button mit der Beschriftung *Switch*, der bereits fest auf der senseBox MCU verbaut ist. Um dieses Vorhaben umzusetzen, testest du bei jedem Durchlauf der Schleife zuerst, ob der Button gedrückt ist. Wähle den Block wenn mache aus der Kategorie Logik und setze ihn an den Anfang der Endlosschleife(). In dem Rumpf kannst du die bereits vorhandenen Blöcke einfügen. Um zu überprüfen, ob der interne Knopf gedrückt ist, benötigst du den Block Drucktaster aus der Kategorie Sensoren. Diesen kannst du direkt als Bedingung einsetzen,

ohne einen Vergleich zu verwenden. Jetzt wird bei jedem Knopfdruck eine neue Zufallszahl ausgewählt und angezeigt.

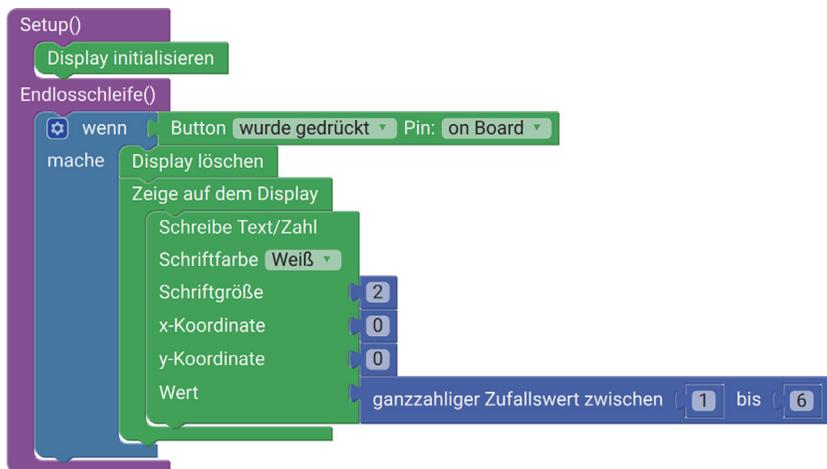


Abb. 6-5 Ein digitaler Würfel, der mit dem Button ausgelöst wird

6.5 Ein Orakel

Natürlich kannst du nun mithilfe der Zufallsfunktion auch ein Orakel erstellen. Das Ziel ist, die senseBox in die Lage zu versetzen, alle deine Ja- oder Nein-Fragen zu beantworten. Das Grundgerüst aus dem vorherigen Projekt kannst du beibehalten. Falls nun der Button gedrückt wird, muss jedoch zuerst die Zufallszahl in einer Variablen gespeichert werden. Suche dazu den Block **Schreibe Element** aus der Kategorie **Variablen** und weise der generierten Variablen eine Zufallszahl zu. Benenne die Variable am besten `zufallszahl`, damit sofort ersichtlich wird, worum es sich handelt. Den Wertebereich des Zufallsgenerators solltest du auf die Anzahl der späteren Antworten anpassen. Dies bedeutet: Möchtest du vier unterschiedliche Antworten, benötigst du vier unterschiedliche Zahlen, die generiert werden können. Nun musst du jede Zahl noch einer Antwort zuordnen. Dies kann man mithilfe einer Fallunterscheidung realisieren. Dazu gibt es in der Kategorie **Logik** den Block **Variablen Fall**.

Dieser Block besteht aus drei Abschnitten. In den Blockabschnitt **Führe aus** kannst du bereits die Anzeige für das Display kopieren. Im Abschnitt **Variablen** definierst du die Variable, für die die Fallunterscheidung durch-

geführt werden soll. Bei Fall (Variable) setzt du den Wert ein, mit dem die Variable verglichen werden soll. Für die erste Fallunterscheidung solltest du somit nun ein Blockgebilde wie in Abbildung 6–6 besitzen.

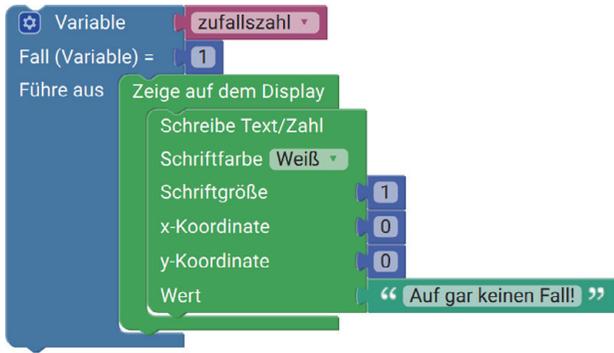


Abb. 6–6 Eine Fallunterscheidung für die Variable `zufallszahl`

Über das Kontextmenü kannst du beliebig viele weitere Fälle hinzufügen. Dies bringt natürlich mehr Abwechslung in das Orakel. Bei einem Klick auf das Zahnrad fällt dir vielleicht auf, dass es auch einen Abschnitt `default` gibt. Dieser wird ausgeführt, falls der aktuelle Wert der Variablen nicht durch die vorherigen Fälle abgedeckt ist. Dies ist nützlich, um herauszufinden, ob der Wertebereich der Zufallszahl erhöht wurde, aber vergessen wurde, zusätzliche Fälle einzufügen. Denn dann kann eine Zufallszahl generiert werden, die nicht von der Fallunterscheidung abgedeckt wird, und es wird der `default`-Abschnitt aufgerufen. Am besten ist es, dort eine ausführliche Fehlermeldung auszugeben, damit du weißt, wieso `default` aufgerufen wurde. In Abbildung 6–7 befindet sich ein Vorschlag für ein mögliches `senseBox`-Orakel. Achte darauf, dass das Display nach jedem Durchgang der Endlosschleife gelöscht wird, bevor etwas neues gezeigt wird. Stelle nun eine Frage und lass dir mit einem Knopfdruck eine Antwort generieren.

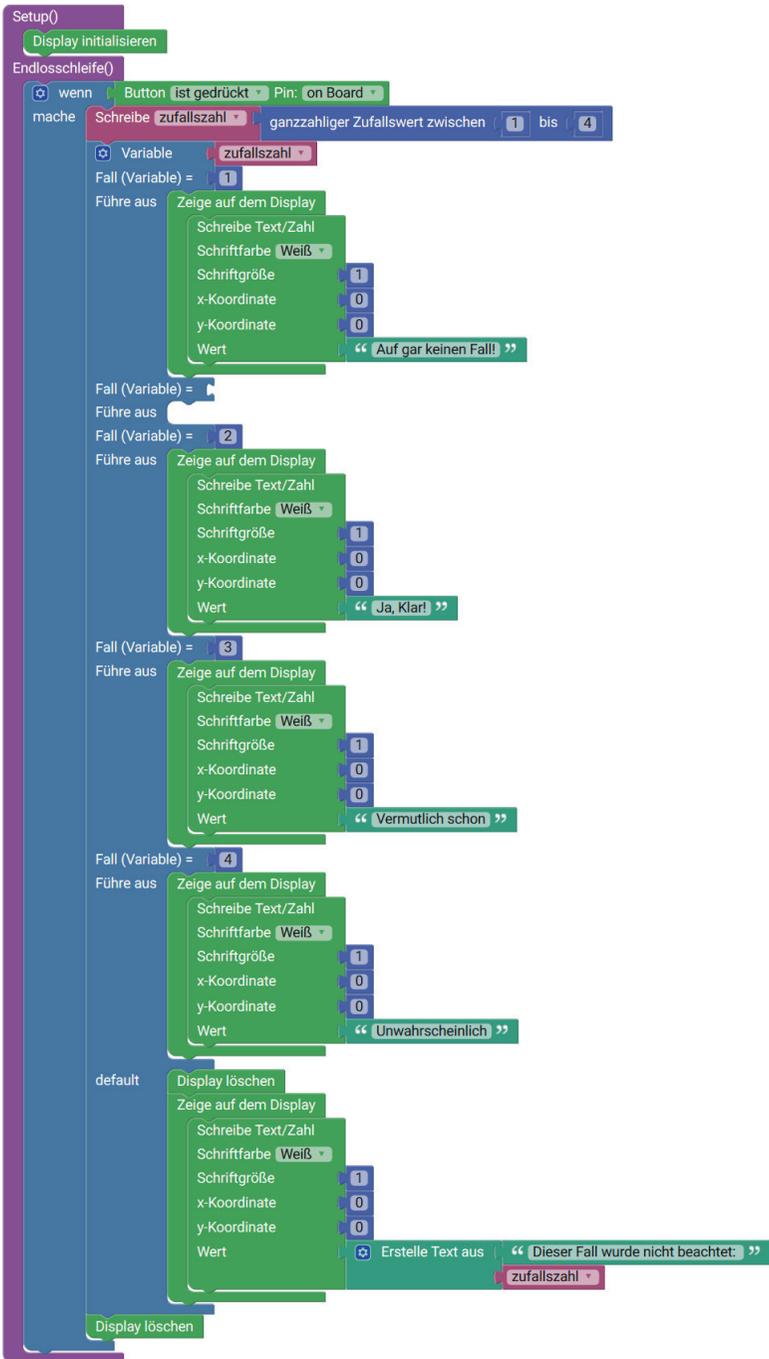


Abb. 6-7 Ein Vorschlag für ein mögliches senseBox-Orakel

6.6 Aufgaben und weiterführende Ideen

- Im ersten Teil des Projekts hast du einen Würfel mit sechs Seiten programmiert. Kennst du auch Würfel mit mehr als sechs Seiten? Dann erweitere doch dein Programm entsprechend.
- Die Ausgabe der Zahl beim Würfeln kann nicht nur auf dem Display erfolgen, schließe zum Beispiel eine LED an und lasse diese eine gewisse Anzahl oft blinken, um die Augenzahl des Würfels anzeigen zu lassen.
- Benutze den Beschleunigungssensor, um den Würfel auszulösen.
- Manipuliere das Orakel mit einem zweiten Knopf. Wenn dieser zusätzlich gedrückt wird, soll immer die gleiche vordefinierte Antwort erscheinen. Damit kannst du deine abergläubigen Freunde manipulieren.