

# Kurzvorstellung

## Projekt „Flappy Bird“

### im Zuge der LV Embedded Systems Software 1

von  
Mathias Metzler  
Benjamin Pal  
Thomas Klima

#### Kurzfassung:

Wir werden als individuelles Projekt das bekannte Spiel „Flappy Bird“ auf dem OLED Click-Modul implementieren. Als zweites Click-Modul wird eines der verschiedenen Sensor-Boards gewählt werden, welches den Input des Benutzers erfassen soll.

#### Gegebene Randbedingungen / Project constraints:

##### Project Constraints

1. Utilize the CY8KIT-5LP Evaluation Board incl. 2 Click modules **(mandatory)**
2. Each Click module should be used to either provide measurement values and/or to display measurement values **(mandatory)**
3. The Evaluation Board and the Click modules should fulfill a specific project description **(mandatory)**
4. Each Click module should use a different communication interface **(optional)**

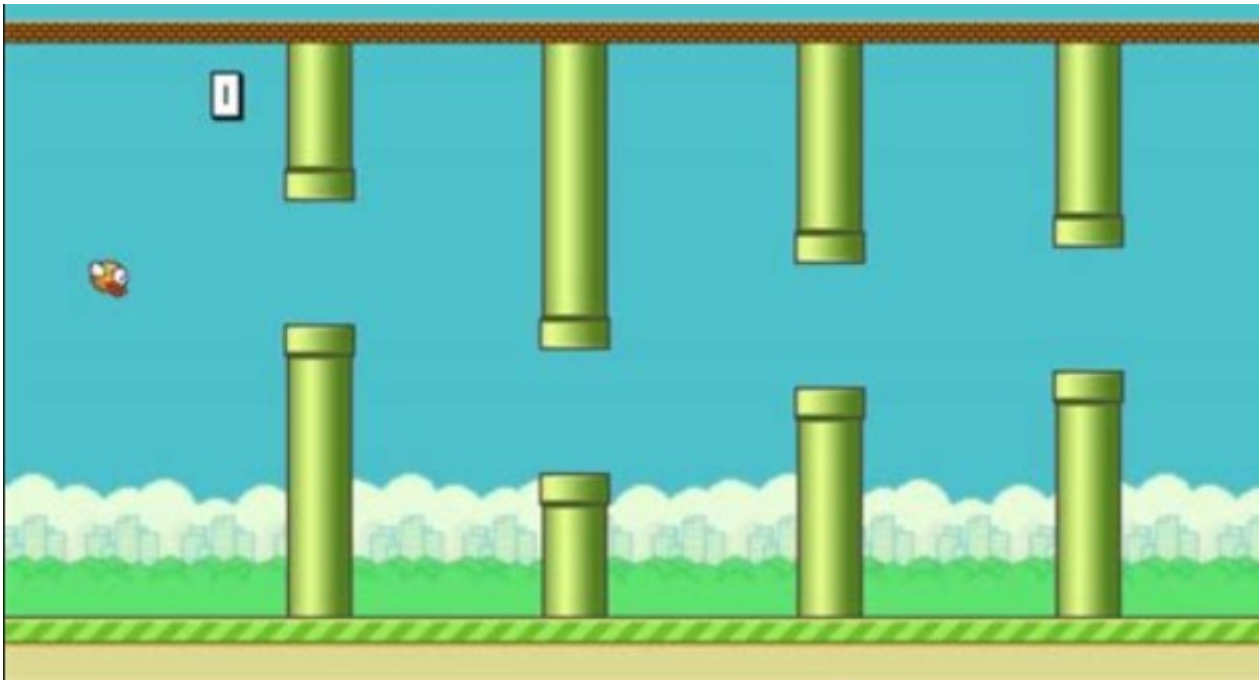
- 1, Zu verwendende Hardware
  - CY8KIT-5LP Evaluation Board
  - OLED C Click Module
  - eines der Sensor Click Module

- 2, Verwendung aller Module
  - Erfüllt, da eines der Module Sensordaten liefert, das OLED Modul der Ausgabe dient.

- 3, Spezifikation
  - s.u.

- 4, Unterschiedliche Kommunikationsschnittstellen
  - Das OLED Modul verwendet SPI, also wird ein I<sup>2</sup>C Sensor gewählt werden müssen

# Grobe Spezifikation



In „FlappyBird“ wird ein Level, bestehend aus undurchdringlichen Säulen von rechts kommend auf den Spieler / Bird zugescrollt. Wird eine der Säulen berührt ist das Spiel vorbei. Der Spieler muss im rechten Augenblick einen input geben, damit der Bird flattert („flap“) und damit den Säulen ausweicht.

Anpassung an unsere Hardware:

Da wir noch nicht wissen, welche Geschwindigkeit und Farbwiedergabe das OLED Display bietet, kann man noch nicht festlegen mit welcher Qualität die Grafik ausgeführt werden wird. Es kann sein, dass wir auf Funktionen wie „draw\_rectangle“ mit eintöniger Färbung zurückgreifen müssen. Falls möglich können wir auch versuchen Sprites bzw. Texturen am Display zu zeichnen.

Auch die Bildwiederholrate ist noch nicht abzusehen, da ohne Prototypen-Hardware auch der Treiber nicht getestet werden konnte, angestrebt wird jedoch ein Wert über 30fps.

User input kann im einfachsten Fall über einen Taster erfolgen, wir werden jedoch versuchen den Beschleunigungssensor oder das Mikrophon zu verwenden. Nachdem wir keine präzisen Werte / Amplituden benötigen, sondern nur mit einem fixen Schwellwert verglichen wird auch keine aufwändige Signalverarbeitung benötigt – wir können uns hier auf eine niedrige Latenz konzentrieren.

Sound kann mit einem externen Lautsprecher eventuell realisiert werden, jedoch erst wenn die Grundfunktionalität steht.

Es wird mehrere Tasks geben:

- 1, DisplayTask – sendet periodisch die aktuellen Grafik-Daten
- 2, WorldSimulationTask – kümmert sich um Scrollen, Kollision und den Vogel
- 3, InputTask – verwaltet die Sensordaten / User input
- 4, MiscTask / Main? - Sonstige Tätigkeiten wie Punktezählen, Sound usw

Die Kommunikation zwischen den Tasks muss erst geplant werden, es bietet sich aber natürlich eine TaskQueue an für den UserInput an, eine Semaphore für den Grafik Zugriff usw..

