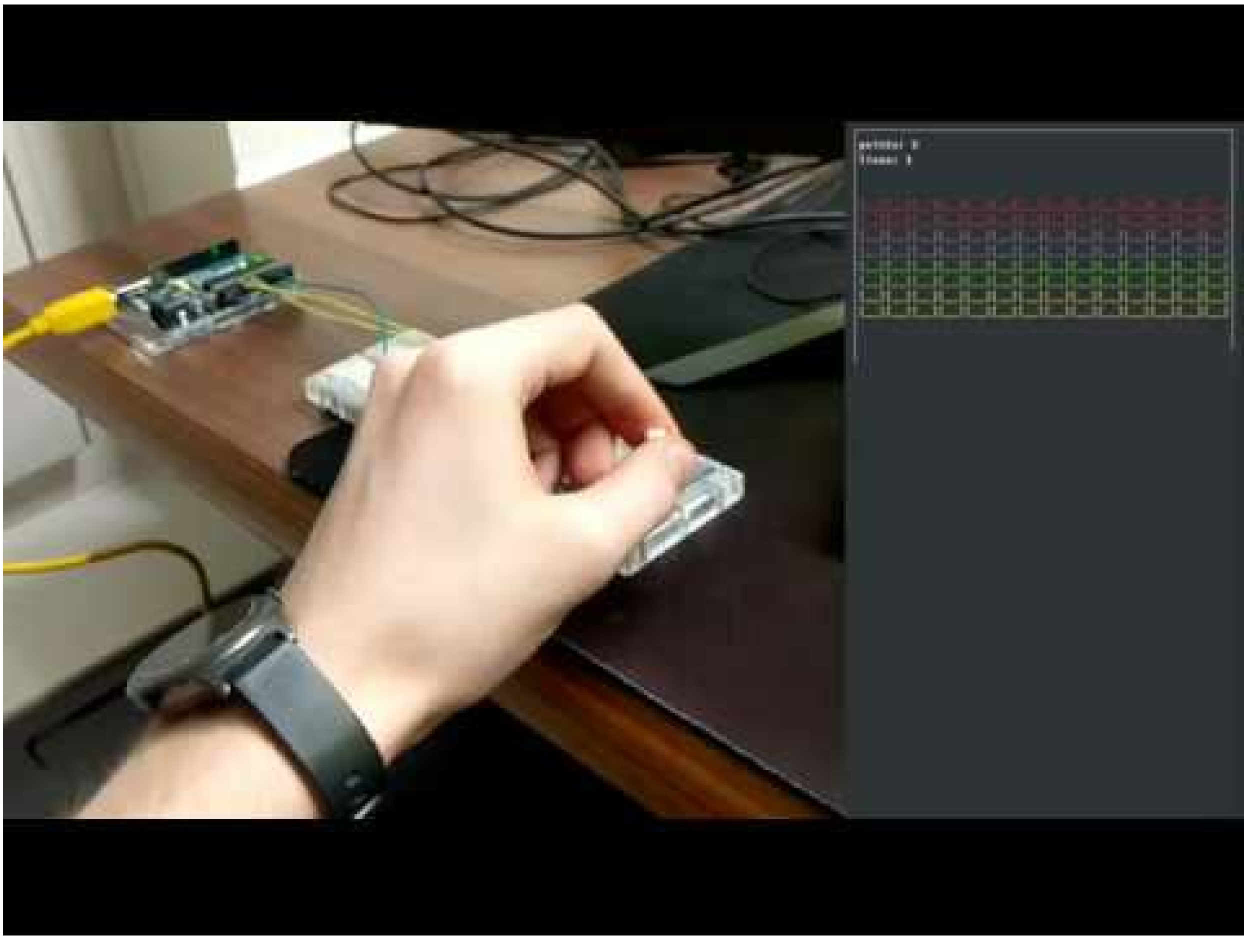
breakout

A breakout clone created in C, using an Arduino as a controller and ncurses to draw the game on screen.

***Warning:* the game flickers quite a bit, as ncurses isn't quite suited to this kind of high-framerate use. If this is likely to bother you (such as if you have epilepsy), don't play it.**

Demo Video



https://www.youtube.com/watch?v=FtQ3F7vpyic

Arduino setup

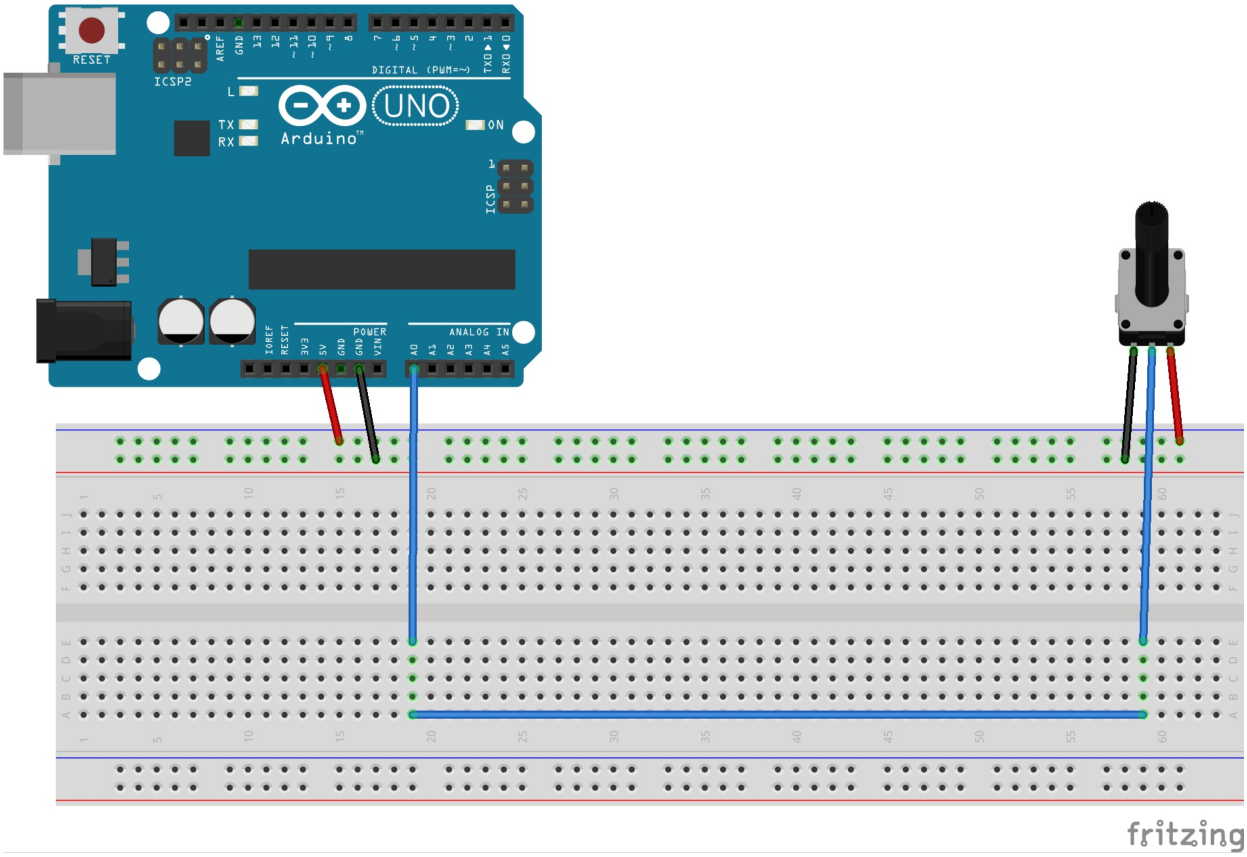
The Ardiuno must be plugged into the computer, and the application in `arduino/arduino/arduino.ino` uploaded to it, using the Ardiuno IDE.

This application sends the current value of the potentiometer to the computer it's connected to every 10ms.

Circuit setup

In practice I put the potentiometer directly on the breakout board, to serve as a sort-of controller.

But any method of connecting the potentiometer to the A0 input, using the 5V supply, will work:



Game setup

Ensure the game is configured to connect to the correct serial port, by checking what port the Ardiuno is connected to in the Ardiuno IDE.

If it isn't `/dev/ttyACM0`, open `arduino/display/breakout.c` and update the `SERIAL_PORT` definition to the correct value.

The game also requires the ncurses library to be installed.

In the project root, compile the game with:

```
gcc arduino/display/breakout.c arduino/display/arduino-serial/arduino-serial-lib.c \
-lncurses -lm -o breakout
```

And run it with:

```
./breakout
```

The game can be exited by pressing:

Ctrl + C

Gameplay

Wikipedia has comprehensive documentation of the original.

This clone implements a subset of its gameplay. Most importantly:

- You have 3 lives to destroy 8 rows of bricks with a ball
- The potentiometer attached to the Ardiuno controls the paddle
- The paddle can be used to deflect the ball back towards the bricks
- If the ball touches the bottom of the play area then you lose a life

There are extra rules concerning when the ball speeds up, how many points you get for destroying bricks, and what happens when the ball hits the top of the screen. Those can be worked out by extensively playing the game, or reading the source code.

License