

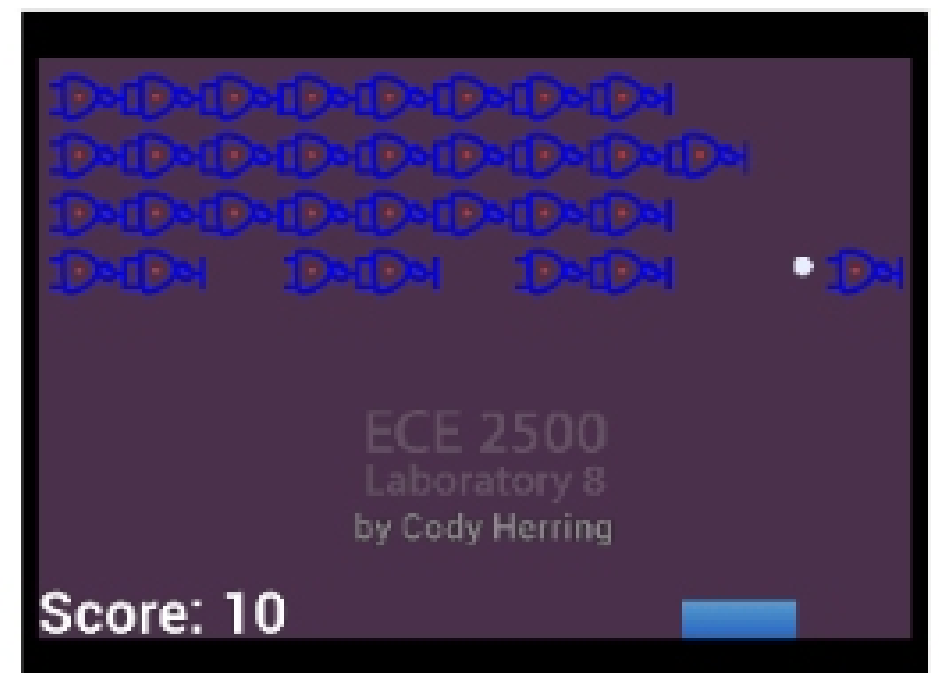
Laboratory Eight

Smartphone Apps

Basic Concepts

Today we will look at writing a software application for the Android smartphone hardware we will be discussing in the lecture. Our application is an arcade game called **breakoutGate**, shown at right. The game uses a software game engine called **AndEngine** and the game is written in **Java**. Your lab instructor will discuss the following concepts:

1. Software applications may be developed on the Android through an Android software development kit (SDK), which is based upon the Java development kit (JDK).
2. To write Java code for the Android, using the Android development tools (ADT), an interactive development environment (IDE) called **Eclipse** (an editor plus more) is provided. All the Android software tools are available at: <http://developer.android.com/sdk/installing.html>
3. Games are typically developed through the use of special software add-ons called engines that provide physics and animation. We will be using code from **AndEngine**.



There is no prelab exercise associated with this lab.

Task One: Load the App and Go

The first thing to with the Eclipse IDE is to load the breakoutGate app into the emulator.

1. Run Eclipse. Go to the home directory and find the Eclipse folder. Run Eclipse. (Eclipse cannot be found by searching "All Programs".)
2. Load the code into Eclipse. The breakout project can be [downloaded](#) from github.com, otherwise your lab instructor will tell you where to find the source code.
3. Start the Android Emulator. From Eclipse, go to Windows -> **AVD** manager. Under Skin size, select **QVGA**, the small size emulator. Close it and next press **Start**. The emulator is slow and it will take a minute to appear. While in the emulator, do a (left) **CTRL F11** to display the screen in landscape mode.
4. Run breakoutGate. Go back to Eclipse and press **CTRL F11** to start the game. It should show you the **NAND** bricks forming **inverter** circuits like shown above.
5. Download the app. Try downloading the app into your own Android phone (you need version 2.3.3 or better):
 - a. Connect your Android with the micro USB cord
 - b. Find breakout.apk in the bin folder on your PC.
 - c. Drag it into your Dropouts Android folder. Click it.

Eclipse Short cuts:

1. **CTRL F11** to begin code execution, i.e. Run
2. **CTRL F** to search for a word. (Never use the replace option. Use Refactor instead.)
3. **CTRL SHIFT O** to eliminate bogus red error flags. (Real errors will remain.)
4. **CTRL SHIFT F** to reformat the code into proper Java form
5. **Project > Clean** forces a rebuild and clears some error flags

Task Two: Some things to try

Here is a list of some things you can modify in the software to alter the game. Please refer to the emulator screen dimensions and variables shown at right.

1. Alter the velocity of the ball. Change the value of **DEMO_VELOCITY** in the beginning of code.
2. Enter your name on the screen. Search for **nameBox** (**CTRL F**) and retype "Cody Herring" (original idea maker student) with your name. Center it on the line.
3. Change the color of the NAND gate brick. Search for "NANDlargeBlue.png" and retype it with "NANDlargeRed.png" (see **gfx assets** picture below).
4. Make SOP NAND circuits. Now lets try some harder things. Staggering every other column of **NAND** bricks downwards by $\frac{1}{2}$ brick will cause the **NAND** gates to form **SOP** circuits. Your instructor will show you how to use an **if else** statement to do this, shifting every other column of bricks downward by $\Delta y = \text{delY} = \text{blockHeight}/2$
5. Making proper form SOP circuits. The **SOP** circuits in Step 4 are not drawn in 2-level proper form. Replace the shifted Red **NAND** gates with Green **OR2B2** gates (see **gfx assets** below).
6. Increase the number of gates. Now replace all of the large gates with small gates from the **gfx assets** directory and increase the number of rows of gates in the display. (If the ball launch is interfered by the lower rows of the gate bricks, lower the ball launching point on the screen.)
7. More ideas? Use Paint to rotate the NAND icons face down. Now make several rows of NANDs, shifting each row to fit. Then move the hungry NANDs downward toward the shooter, like an alien arcade game. Feed more rows down from the top.

