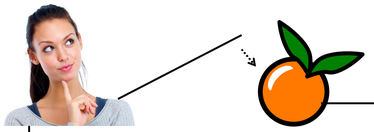
What is Computer Engineering?



How does the hardware work?

The MaKey MaKey is a printed circuit board that uses an ATmega32u4 microcontroller running Arduino Leonardo firmware. The MaKey MaKey provides convenient ways to hook up inputs, and the microcontroller handles how to react to them.

By completing a simple circuit, the Arduino can recognize which input was triggered. Then, it can send key presses, mouse clicks, or mouse movements. We use high-resistance switching so you can close a circuit even though materials like oranges, coins, and pencil lead. For this project, we simply send key presses like 'r', 'd', 'f', and 'g' to our Raspberry Pi whenever you complete a circuit.

Debouncing:

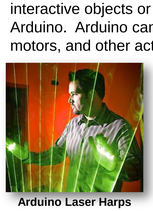
Pushing a button isn't as smooth as you'd think! In reality, the signal "bounces" up and down before it finally makes a good connection.

The Arduino on the MaKey MaKey has been programmed to wait for the input signals to fully settle before registering a new button press.

Cables

What else can Arduino do?

Arduino is an open-source prototyping platform intended for anyone interested in creating interactive objects or environments. The MaKey MaKey is just one of the projects based on Arduino. Arduino can receive input from a variety of sensors and can send signals, control lights, motors, and other actuators. Some other cool projects include:



Arduino Laser Harps

Open Source GameBoy

Color Combo Door Lock

Dell 2000FP

(20.2" x 16.5" approx)

DVI and Power out back

What is a Raspberry Pi?

The Raspberry Pi is a credit-card-sized computer developed in the UK by the Raspberry Pi Foundation to promote learning basic computer science. It is cheap, robust, and portable. It's a perfect machine for making custom projects and products around school, home, or work.

How does the software work?

We run Raspian, a optimized version of Linux, on our Raspberry Pi. This way, the Raspberry Pi is in charge of the software-side of things. We've compiled this Bomberman game using C++, a common programming language you'll learn here at BYU! Then, we've configured Bomberman to recognize the keyboard letters sent here by the Arduino so you can compete in Bomberman with play-doh, coins, or anything else conductive!

How powerful is it?

The GPU is capable of 1Gpixel/s, 1.5Gtexel/s or 24 GFLOPs of general purpose compute and features a bunch of texture filtering and DMA infrastructure. This means that graphics capabilities are roughly equivalent to the original Xbox's level of performance. Real world performance is like a 300MHz Pentium 2, only with much, much swankier graphics.

Cable s

What else can a Raspberry Pi do?

Basically, a Raspberry Pi is a powerful, compact, $30 computer! Because it can run Linux, you can use it for nearly any software projects like the three (of *hundreds*) shown below!

Pandora Jukebox

Mountable Web Server

Home Media Center

Hardware

Computer Engineering is all about interfacing two key components, the first of which is hardware. Believe it or not, hardware is more than just what is found within computers! Here at BYU you will learn about several hardware components that are fundamental to computing in a variety of different applications.

Microcontrollers:

Microcontrollers are designed for use in embedded systems. Generally, they are lower power and more cost-efficient than typical microprocessors, making them ideal for applications such as robotics, automobile control systems, and medical devices. They are usually programmed in assembly or C. A great example of this is the MaKey MaKey which using an ATMega32u4 microcontroller.

FPGAs:

A Field-Programmable Gate Array is another type of integrated circuit that is most often used to perform complex digital computations very quickly. They are programmed with the use of Hardware Description Languages (like VHDL or Verilog) which are capable of describing a digital circuit. Ask us about the cool Space and CERN projects involving FPGAs going on at BYU right now!

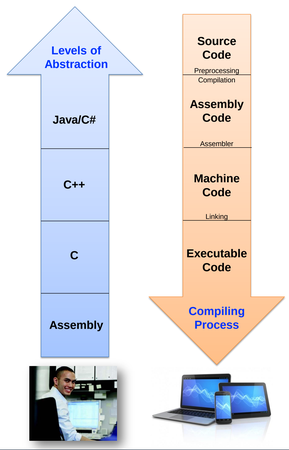
Microprocessors:

All modern CPUs are microprocessors. It is a programmable device that accepts digital data as input, processes it according to the instructions stored in its memory, and provides results as output. Here at BYU you will learn how processors operate, design your own simple LC-3, and see how techniques like hyper-threading, caching, and pipelining improve performance.

You'll also learn about ASICs, DSPs, GPUs, and create your own powerful systems within Computer Engineering!

Software

The second key component to Computer Engineering is software. Although hardware is where the rubber meets the road in computing systems, it would not be very useful if there was no software for it to run. Computer Engineers get to learn a variety of different programming languages that run on hardware at different levels of abstraction. They also understand how programs go from code to execution. In other words, you get to develop some really sweet stuff!




```
//*************************************************************************************************
//*************************************************************************************************

void CBomber::Die (void)
{
    if (theDebug.CanBombersDie())
    {
        // If the bomber has not won the current match
        if (!m_Victorious)
        {
            // Make the bomber start dying : set the dead state
            // and reset the timer.
            if (m_Dead == DEAD_ALIVE)
            {
                // If the bomber was doing something with a bomb
                if (m_BombIndex != -1)
                {
                    // If the bomber is just lifting this bomb
                    if (m_BomberState == BOMBERSTATE_LIFT ) {
                        // End the lifting and make the bomber hold the bomb
                        m_pArena->GetBomb(m_BombIndex).SetBeingHeld();
                        m_BomberState = BOMBERSTATE_WALK_HOLD;
                    }

                    // If the bomber died while holding or throwing a bomb
                    if (m_BomberState == BOMBERSTATE_WALK_HOLD ||
                        m_BomberState == BOMBERSTATE_THROW)
                    {
                        // Make him throw the bomb (with no bomber throwing animation)
                        MakeBombFly(BOMBFLIGHTTYPE_THROW);
                    }
                    // If the bomber died while punching a bomb
                    else if (m_BomberState == BOMBERSTATE_PUNCH )
                    {
                        // Make him punch the bomb now (with no bomber punching animation).
                        // Yes, that's strange, since he is dying now. But we have to release the bomb
                        // otherwise it won't explode (bug tracker #2160381). Besides, this can only happen
                        // during the first animation sequence (ALTHOUGH IF THE
```

The Bomberman game you're playing is *thousands* of lines of C++ code. Here is a snippet of the code that tells our computer what to do when a Bomberman gets burned.

In Computer Engineering, you'll learn how to develop software using object oriented design, write clean code, and optimize programs for performance!

Why be a Computer Engineer?

Being a Computer Engineer is the best! You'll be set apart from the competition because of your understanding of both hardware and software in computing systems. Not only do you get to create innovative technology, you get to work on some of BYU's own award-winning projects such as:

Robot Soccer Competition

Light Suits

Robotic/Machine Vision

Not to mention, there is an expected **21% employment growth** ending in 2018. In 2013, the graduating class of Computer Engineers experienced an **average starting salary of $70,900!**

If you want to be in high demand, make a good living, guide technology forward, and love your job…

Apply Now!

Visit http://ee.byu.edu

Cables

BYU | ELECTRICAL & COMPUTER ENGINEERING