Agenda

- History of the Raspberry Pi
- Hardware
- ARM Processor
- Programming
- Applications
- Raspberry Pi vs BeagleBone
History of the Raspberry Pi

- The purpose of creating these affordable, programmable computers was to promote basic computer science to be taught in schools.
- Created by the Raspberry Pi Foundation (UK)
- Sold over two million units within two years of having licensed manufacture deals with element 14/Premier Farnell and RS Electronics.
Why you should use one

- Low cost and low power
- Simple to use
- Educational device for youths and hobbyists to learn about programming
- Versatile for many different projects/applications
- Compact Size
- Immense amount of resources for different projects
Hardware

- 4 Models: A, A+, B, B+
- 85.6mm x 56mm x 21mm
  - The size of your student ID
- ARM1176JZF-S Processor
  - Typical clock size is 700 MHz, performing at approximately 40 MFLOPS
    - Can be overclocked to 1GHz without any issues
- Includes VideoCore IV graphics processor with 1 billion pixels per second
- 512 MB of RAM is built into the board, not replaceable or upgradeable
- Includes multiple built on I/O ports
  - 100 MB/s Ethernet port
  - HDMI port and RCA port
  - Audio Jack
RISC Architecture

- Low Transistor Count
- Low Power Consumption/Heat Production
- Used in most mobile devices
  - Phones
  - Laptops
  - Small Digital Devices
- Raspberry Pi has similar requirements to mobile devices
- Architecture allows for various Unix OSs
  - Raspberry Pi can utilizes nearly all Linux distros
Instruction Set

- 3 Instruction Sets
  - 32 Bit ARM
    - Single Instructions
    - Handles Data
    - Organizes Processor Segments
  - 16 Bit Thumb
    - Specialize in branch range and address space
    - Used with ARM for rapid interrupts
    - Used for Digital Signal Processing
  - 8 Bit Java
    - Jazelle Technology
    - Deals with complex Java bytecodes
Major Processor Segments

- Integer Core processes integer values
  - 40 total 32-bit registers
  - Three Pipelines
    - ALU, MAC, and Load/Store
  - ALU handles all arithmetic, logic, shift, and saturation operations
  - MAC handles all Multiply operations
    - 32x16 multiplier
    - Accumulator
Major Processor Segments

- Load/Store Unit handles all load and store operations sent from the Integer Core, and decouples these instructions from the MAC and ALU pipelines.
- Prefetch Unit handles all instruction calls
  - Utilizes both types of branch prediction
  - Combined with the Branch Target Address Cache (BTAC) results in nearly zero wasted cycles (Dynamic)
  - Also handles branches not in BTAC with normal branch predictor (Static)
Major Processor Segments

- Memory Management Unit organizes all memory calls, in order to make them more efficient, lowering system delays.
- Vector Floating Point Coprocessor (VFPC)
  - Core of process is integer
  - Floating Point Operations done here
    - Single and Double Precision
    - Eight single or four double elems
- Vector Interrupt Control (VIC) Interface
  - Handles all interrupts
  - Deals mainly with external systems
  - Request Signal allows faster interrupt
Pipelining

- 8 stage pipeline
- Datapath consists of three pipelines:
  - ALU, shift, or Sat pipeline
  - MAC pipeline
  - Load or store pipeline
- Fetch stages can hold up to four instructions. Branch prediction performed on instructions ahead of execution of earlier instructions
- Issue and Decode stages can contain any instruction in parallel with a predicted branch
- Execute, Memory, and Write stages can contain a predicted branch, an ALU, or multiply instruction load/store multiple instruction, and a coprocessor instruction in parallel execution.
Pipeline Stages

Fe1: 1st fetch stage
Fe2: 2nd fetch stage
De: Instruction decode
Iss: Reg. read and issue
Sh: Shifter stage
ALU: ALU operation
Sat: Saturation stage
WBex: Writeback Mul/ALU

MAC1: 1st multiply acc. stage
MAC2: 2nd multiply acc. stage
MAC3: 3rd multiply acc. stage
ADD: Address generation
DC1: Data cache 1
DC2: Data cache 2
WBls: Writeback from LSU
Programming on a Pi

- Raspberry Pi supports multiple Linux distributions
  - Ubuntu
  - Openelec
  - OSMC
- The installed ARMv6 also handle many languages
  - Python
  - C
  - C++
  - Java
  - Ruby
  - Scratch
- More programming languages can be installed with proper support tools
Applications

- While originally used to teach kids about programming, the Raspberry Pi has evolved to be used for a multitude of different applications
  - Automated Light Control System
  - Game System Emulator
  - Surveillance System
  - Universal Remote Control system
  - Automated House system
  - Music Streaming Speakers
Surveillance Camera
Game Emulator
Remote Control Car
## Raspberry Pi vs BeagleBone

<table>
<thead>
<tr>
<th>Feature</th>
<th>Raspberry Pi</th>
<th>BeagleBone Black</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cost</strong></td>
<td>$35</td>
<td>$45</td>
</tr>
<tr>
<td><strong>I/O Pins</strong></td>
<td>8 Pins</td>
<td>65 Pins</td>
</tr>
<tr>
<td><strong>Power Draw</strong></td>
<td>260-350 mA</td>
<td>210-460 mA</td>
</tr>
<tr>
<td><strong>Processor Speed</strong></td>
<td>700 - 1000 MHz</td>
<td>1000 MHz</td>
</tr>
<tr>
<td><strong>RAM</strong></td>
<td>512 MB</td>
<td>512 MB</td>
</tr>
<tr>
<td><strong>Video Outputs</strong></td>
<td>HDMI, Composite</td>
<td>Micro-HDMI</td>
</tr>
<tr>
<td><strong>Audio Outputs</strong></td>
<td>Stereo over HDMI, 3.5mm</td>
<td>Stereo over HDMI</td>
</tr>
<tr>
<td><strong>Onboard Memory</strong></td>
<td>SD Card</td>
<td>2 GB On-board, MicroSD</td>
</tr>
</tbody>
</table>
Sources

- http://makezine.com/magazine/how-to-choose-the-right-platform-raspberry-pi-or-beaglebone-black/