

Computer Automation Techniques

Arthur Carroll

Three Types of Computers

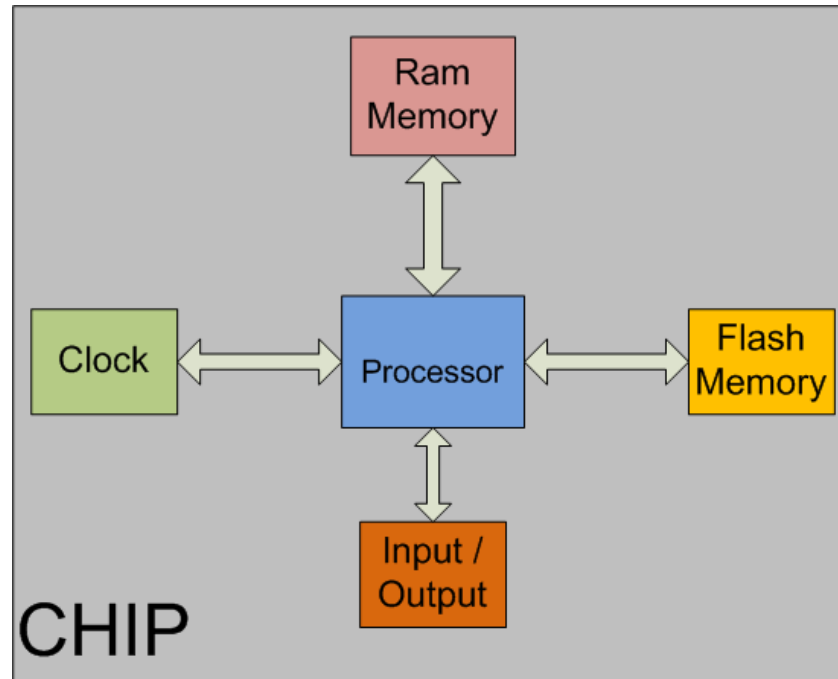
- Micro-Controller
- Single Board Computer
- Desktop Computer

The Micro-Controller

- Small inexpensive DIP or surface mount chips
- Roughly 10-100 MHz
- A complete, stand alone computer on one chip.
- Common Types: AVR, ARM, PIC
- Current Industry Leader: Atmel AVR (ARM close second, PIC is fading out)
- Can be programmed in C code or assembly language
- Usually programmed using mostly C, and a little assembly for certain functions
- Can only run one C or assembly program at a time



Micro-Controller



- *Input / Output includes digital I/O, analog in, and serial ports
- *Clock can also be an external crystal, which is more accurate than the internal clock
- *Flash Memory can be expanded with external chip

Micro-Controller Applications

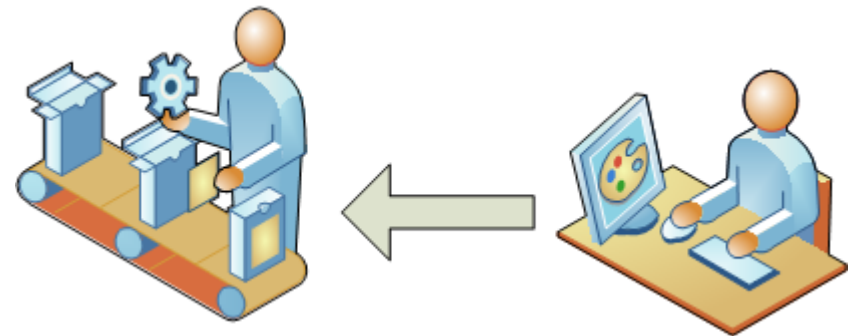
- Consumer Electronics: Alarm System, MP3 player
- Automotive: Engine Control Unit, ABS system
- Medical: Heart Rate Monitor, Robotic arm
- Military: SAM, small UAV(bomb clearing robot)
- Many more....

Advantage of AVR over logic

Multiple assembly lines for different chips



One assembly line making the same chip with different programs



Microcontroller Programming Environments

AVR

- Paid Industry Standard: Code Vision AVR
- Free Industry Standard: Win AVR / AVR - GCC

ARM

- Paid Industry Standard: Keil / IAR (tie)
- Free Industry Standard: ARM - GCC

ARM vs AVR

- ARM usually has a slightly higher clock speed when compared to the equivalent AVR
- AVR has is easier to use and can supply more current to external devices
- Not always a clear advantage between one or the other
- AVR is best to learn with because of simple assembly syntax

Common AVRs

- Attiny 2313/ Atmega8



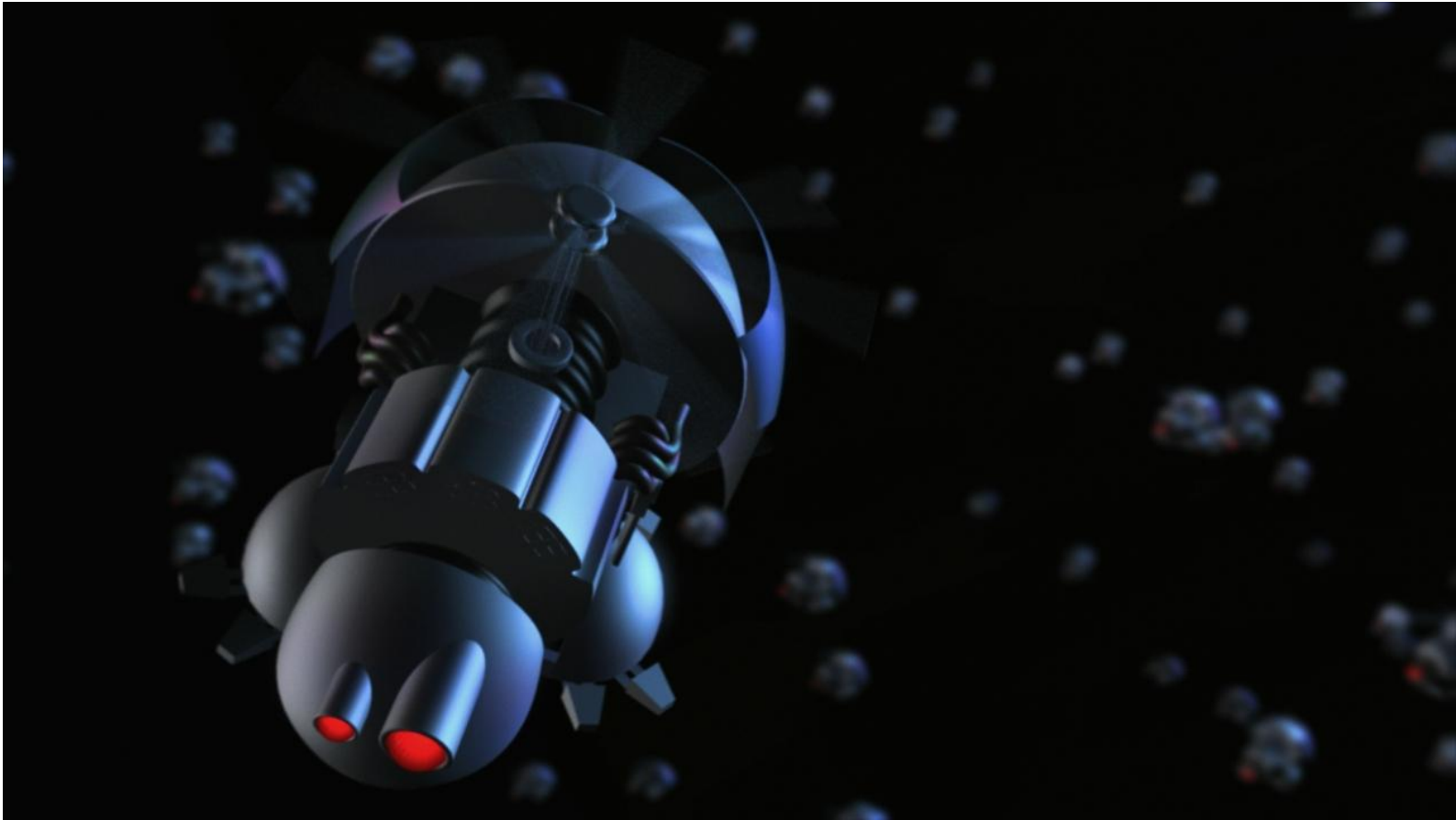
http://www.sparkfun.com/commerce/product_info.php?products_id=360

- Atmega128



http://www.futurlec.com/ET-AVR_Stamp.shtml

Future of AVR

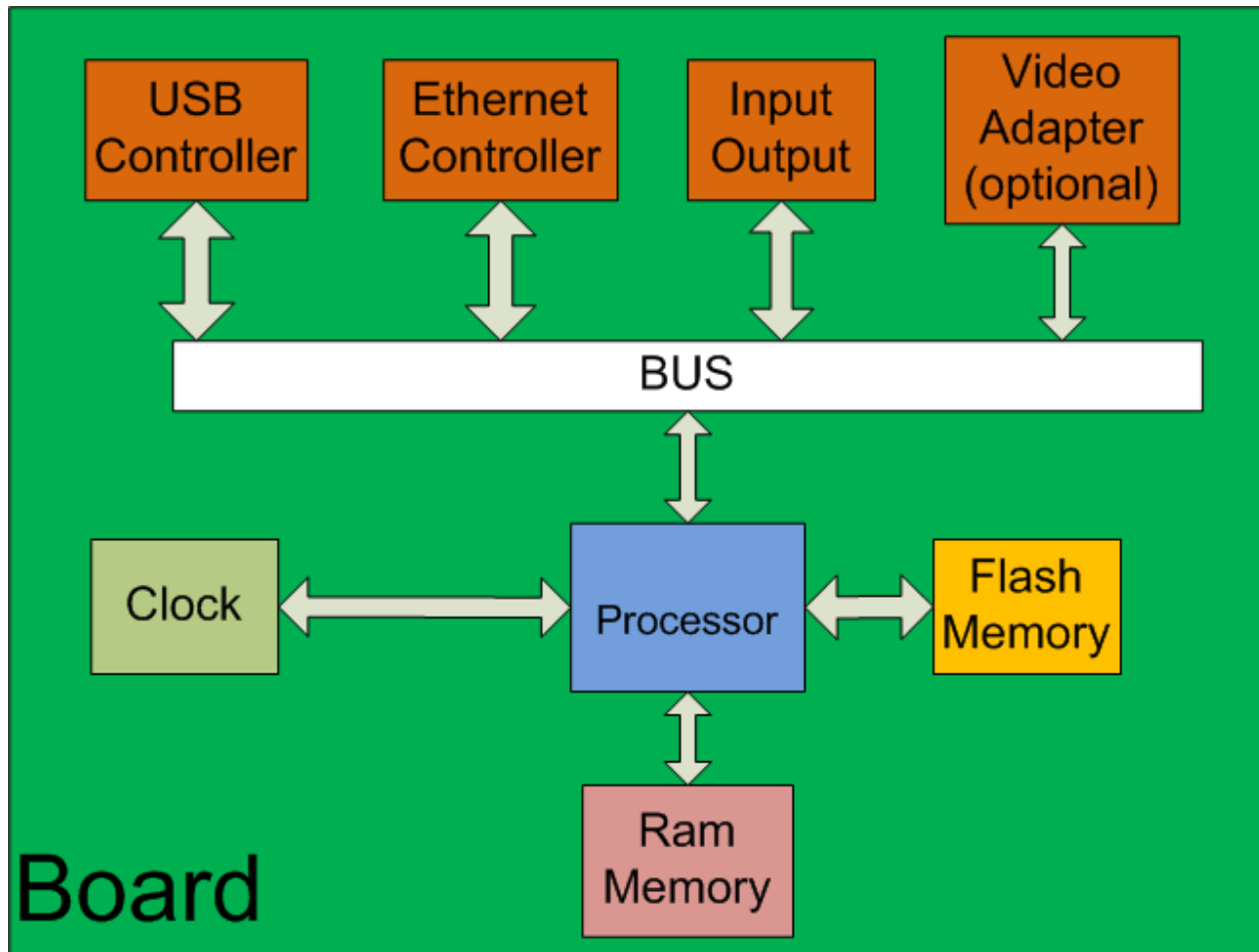


The Single Board Computer

- Relatively small
- Runs fully functioning OS
- Clock speed 300Mhz – 1.2GHz
- Complete computer on one board
- Common Types: TS7800, Gumstix
- No clear leader, relatively new technology
- Generally programmed exclusively in C / C++
- Terminal or basic LCD GUI interface(cell phone)



Single Board Computer Layout



I/O usually 100+ pins and 2-4 USARTS

Single Board Computer Applications

- Consumer Electronics
 - Helicopter
 - Boeing 747
 - Luxury Car
- Manufacturing Plants
- Military
 - Advanced Mobile Robots(super sonic UAV)
 - Abrams Tank

Advantage of SBC over AVR

- Internet:
 - Remote login, send email, communicate with other computers
- USB port:
 - control with of the shelf Gamepads (ie. XBOX 360 or Playstation)
 - GPS
 - Keyboard / mouse, virtually anything USB
- Can run multiple programs at once.
- Can compile and execute C / C++ code, allowing remote updates.
- Large library of existing programs ready to deploy, free and paid.

Operating Systems

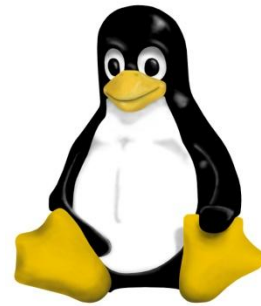
- Single Board Computers can run a full operating system
- Windows CE vs Linux
- Linux is most common, because the Windows license brings up cost
- Allows multiple programs to run simultaneously
- Abstracts many functions you would otherwise have to write(clock, internet protocols, device drivers)

Windows CE



- Not free, no source code available.
- Not a full version of windows, more limited than XP
- Excellent DirectX graphics interface.
- Visual Studio 2008 Support, the best IDE available.
- Not considered robust, prone to crashing. This is because the OS is written as multiple programs that communicate with each other.
- X86 family of processors only

Linux



- Free and includes source code. GNU
- Full desktop version of Linux, can run code written for desktop version
- Extremely robust, almost never crashes, OS written as one single program
- Tools not as developed as Windows, but not very far behind
- Written by volunteer community, and supported by hardware manufactures
- X86 and ARM processors, ARM usually has higher speed and lower power consumption

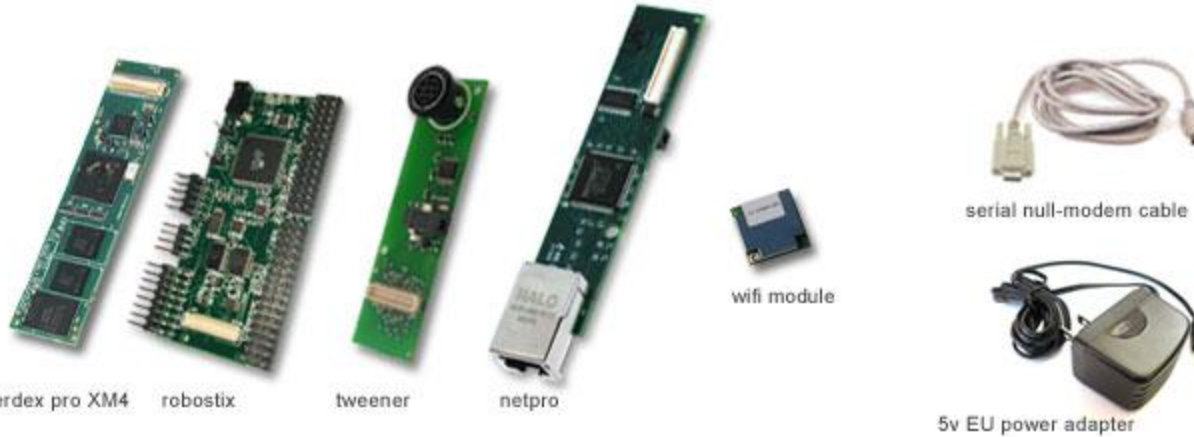
Debian Linux



- Most common embedded OS
- The very popular Ubuntu is based on Debian
- You can download and install Ubuntu with Wubi like an application from XP. Making it easy to try and learn.
- Has large library of free software
- Not as user-friendly, requires more knowledge (but getting better every day)

Most Common SBC Model

➤ **robotics wifi pro pack FCC** with robostix, 10/100, wifi, USB host & microSD storage.



- http://gumstix.com/store/catalog/product_info.php?products_id=203

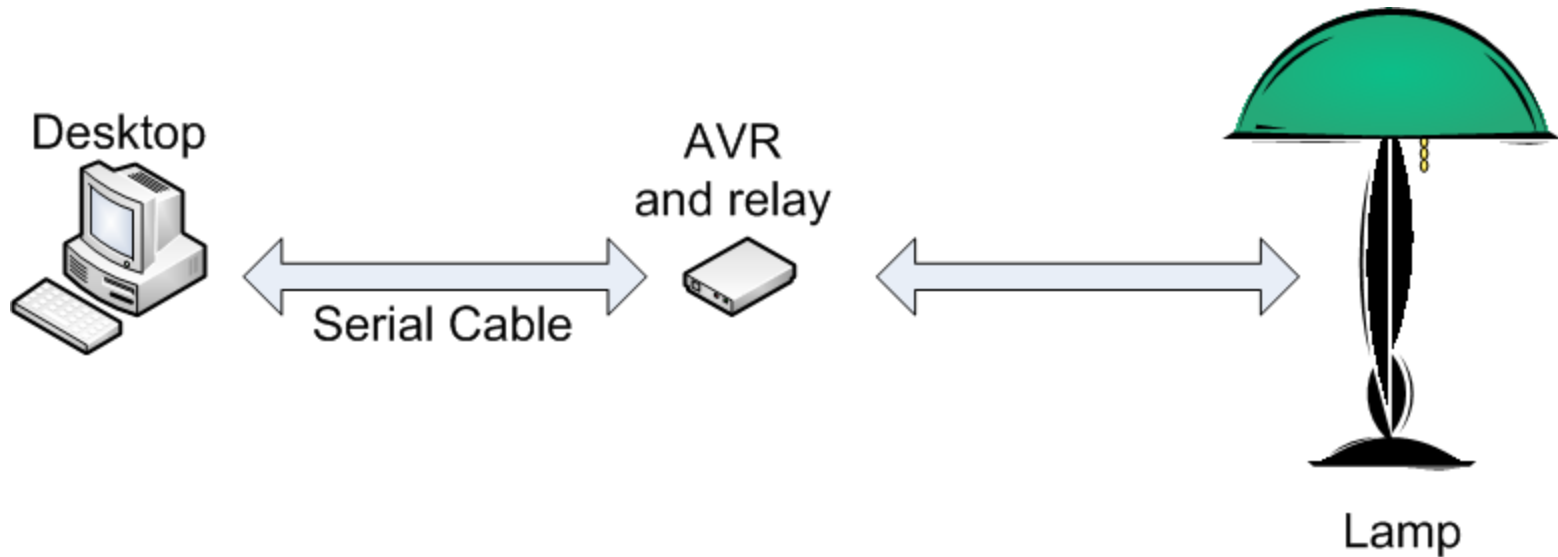
Future of SBC



Desktops

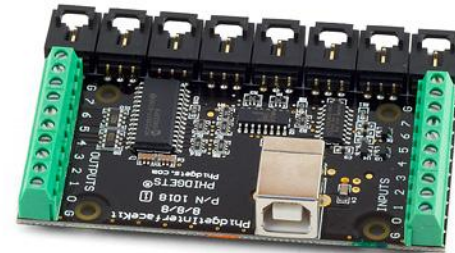
- Extremely fast processor and large memory capacity
- Generally not mobile, but more so lately with invention of solid state hard drive
- Built in GUI interface, can easily run GUI based programs that interface with the outside world
- Market Leader: Intel X86
- Programmed in any language you choose

How to control a lamp with your Desktop

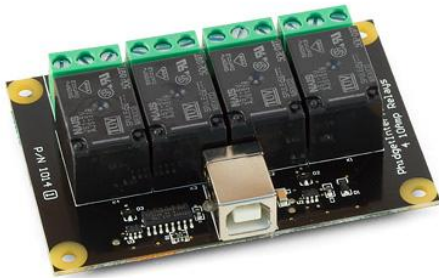


Phidgets

- Off the shelf solution
- Many expansion boards
- Easy to use



Phidget I/O Board



Phidget Relay Board



Humidity / Temp sensor expansion board



Dimmer switch expansion board

Similarities

- The clock speeds and processor names are constantly changing, but the fundamentals of C / C++ remain the same
- C / C++ works on every computer, with the exception of device specific code
- There are many ways to accomplish the same task
- A lot of overlap between categories
- You will see that C / C++ for the PC and AVR are exactly the same, in fact, C / C++ is the same for any processor

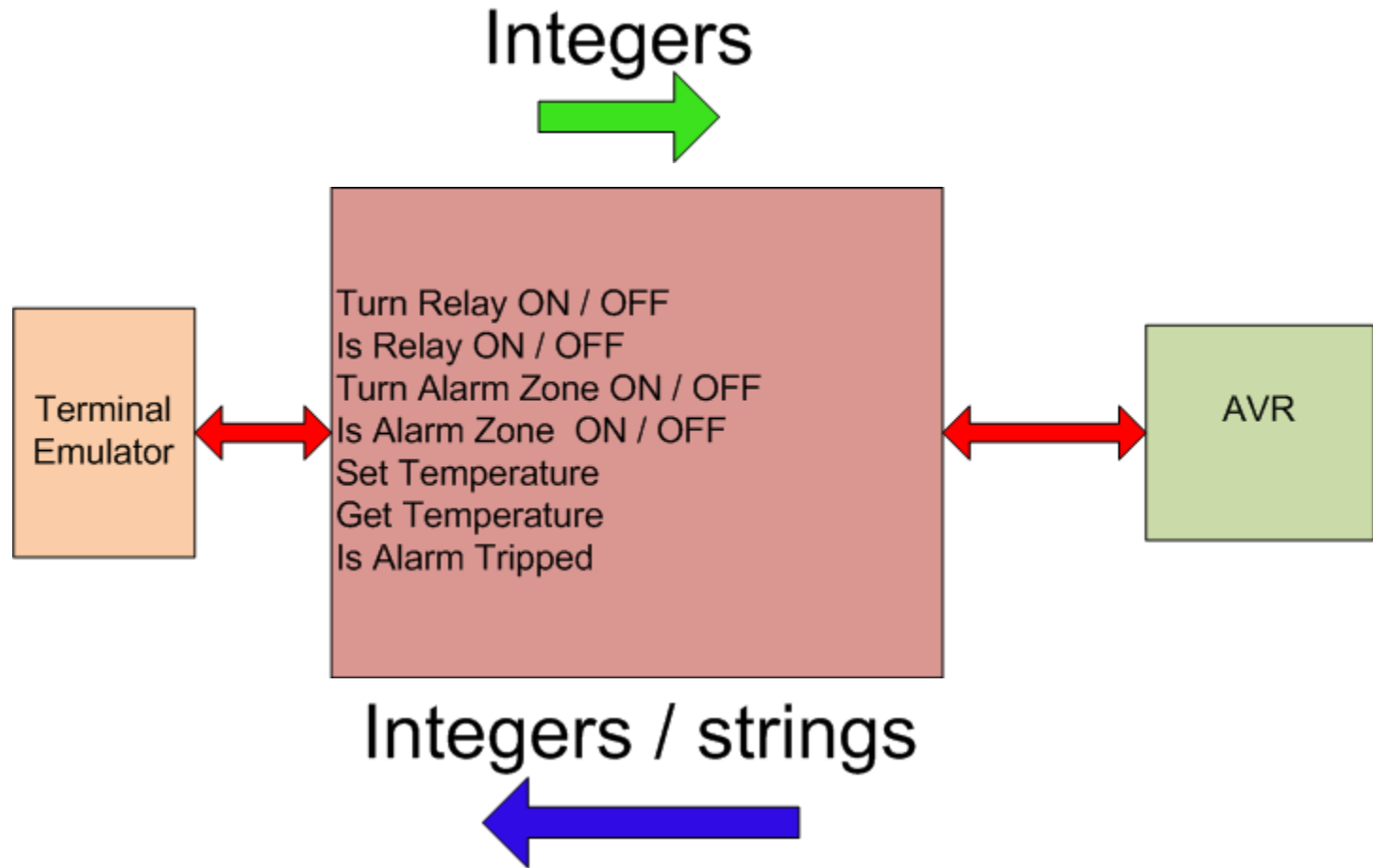
Final Project

- You will build a phidget and use it to control a smart house
- The phidget will be similar to the diagram in slide 22
- You will write the code for the AVR, then build and demonstrate the circuit

Minimum Hardware Requirements

- Inputs: 8 pins representing Temperature input value in hexadecimal(DIP switch), at least one alarm trigger switch, RS-232 port connected to PC
- Outputs: At least 1 relay output for 120Vac appliances, 8 pins representing Temperature output value in hexadecimal

Minimum Software Requirements



Phidget Protocol

Hyper Terminal

- “1” = turn relay ON
- “2” = turn relay OFF
- “3” = is relay ON
- “4” = is relay OFF
- “5” = what is the temp
- “6” = set the temp

AVR

- Return any value asked for from PC
- Do any task requested from PC
- If alarm zone is tripped immediately print message to terminal
- Periodically print state of all inputs / outputs to terminal
- Ask for temp if sent “6”

Controlling the AVR through a terminal emulator

- You will mainly use the `scanf()` function to read numbers from the Usart
- Use Codevision to setup the serial port:
 - 9600 8-N-1 (look it up)
 - Use Putty if you have vista
- Example of using serial port and `scanf()` in book

Minimum documentation

- Flow chart of AVR program
- C / C++ source code for AVR(compiles in Codevion AVR)
- Compiled binary of program
- All files should be zipped into one file(.RAR, .tar.gz, .zip...)
- Physical demonstration to a TA during the week of each due date