Homework # 2

Alec Newport [acn55], Eldor Bekpulatov [eb654]
ECE 5725 - Thursday Lab

1. A Linux pipe exists only as an abstraction for a communication channel between a single parent process and a single child process. A named pipe, on the other hand, exists as a permanent object in the file system and can be used for communication between any number of processes running anywhere on the system.

2. The name of the fifo is the same as the name of the media player (mplayer), so it's difficult to tell which is referencing the fifo and which is referencing the program.

3. - Pin 1 used for 3.3V for powering all systems except backlight
   - Pins 2 and 4 used for 5V for powering the backlight
   - Pin 3 used for SDA for serial communication with the touch screen controller
   - Pin 5 used for SCL for serial communication with the touch screen controller
   - Pins 6, 14, 20, 30, 34, and 39 used for ground
   - Pin 11 used for GPIO17 for SW3
   - Pin 13 used for GPIO27 for SW2
   - Pin 15 used for GPIO22 for SW1
   - Pin 16 used for GPIO23 for SW4
   - Pin 18 used for RT_INT for sending a reset signal to the touch screen controller
   - Pin 19 used for MOSI_3V for SPI serial communication
   - Pin 21 used for MISO for SPI serial communication
   - Pin 22 used for write execution control and serial register select in the serial interface
   - Pin 23 used for SCLK_3V for SPI serial communication
   - Pin 24 used for a chip select signal in the serial interface
   - Pin 27 used for EEDATA for serial communication with the TFT's EEPROM
   - Pin 28 used for EECLK for serial communication with the TFT's EEPROM

4. Max set: GPIO2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27
   Min set: GPIO5,6,12,13,19,26

5. We first had to run the python script in the background because it needs to listen to any inputs from PiTFT and pipe the arguments to the mplayer. We can run the mplayer in the foreground since we are using the PiTFT screen to display the video, and with python listening in the background we can exit mplayer and also terminate the python process.

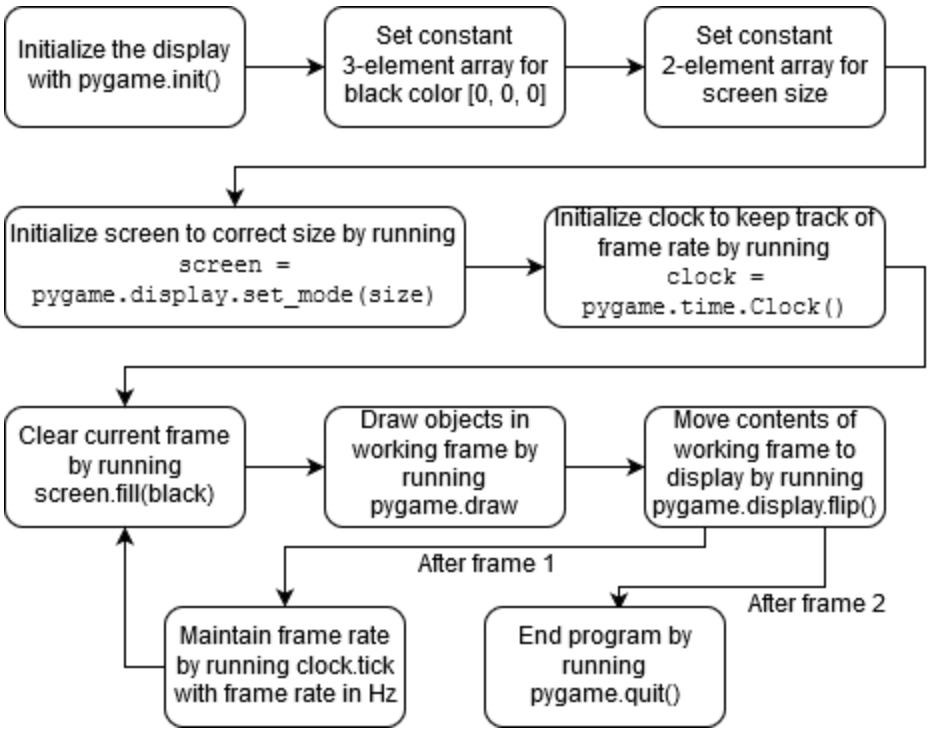
6. The data and time are accurate because the RPi gets them from the internet. If the RPi was disconnected from the internet and an accurate time needed to be maintained, an external, battery-backed real-time clock should be added.

7. If the software set up the GPIO as an output and drove the output high, either by accident or by another user, and the switch was closed, there would be a direct power to ground short without R2 that would blow out the GPIO. R2 prevents the problem because it limits the current output by the pin in this situation. The value of R2 is set to 1k ohm because it limits the current drawn to 3.3mA, a sufficiently low value so as to not damage the pin, even in this situation.

8. A surface is a pygame representation of an image. Surfaces can be stacked together to create more complex images or frames. A rect simply designates a 2D area on the screen using coordinates of the upper left corner of the rectangular area as well as the height and width. It does not represent or contain an image itself. Every surface has an associated rect which defines the location of the surface as well as its 2D area.

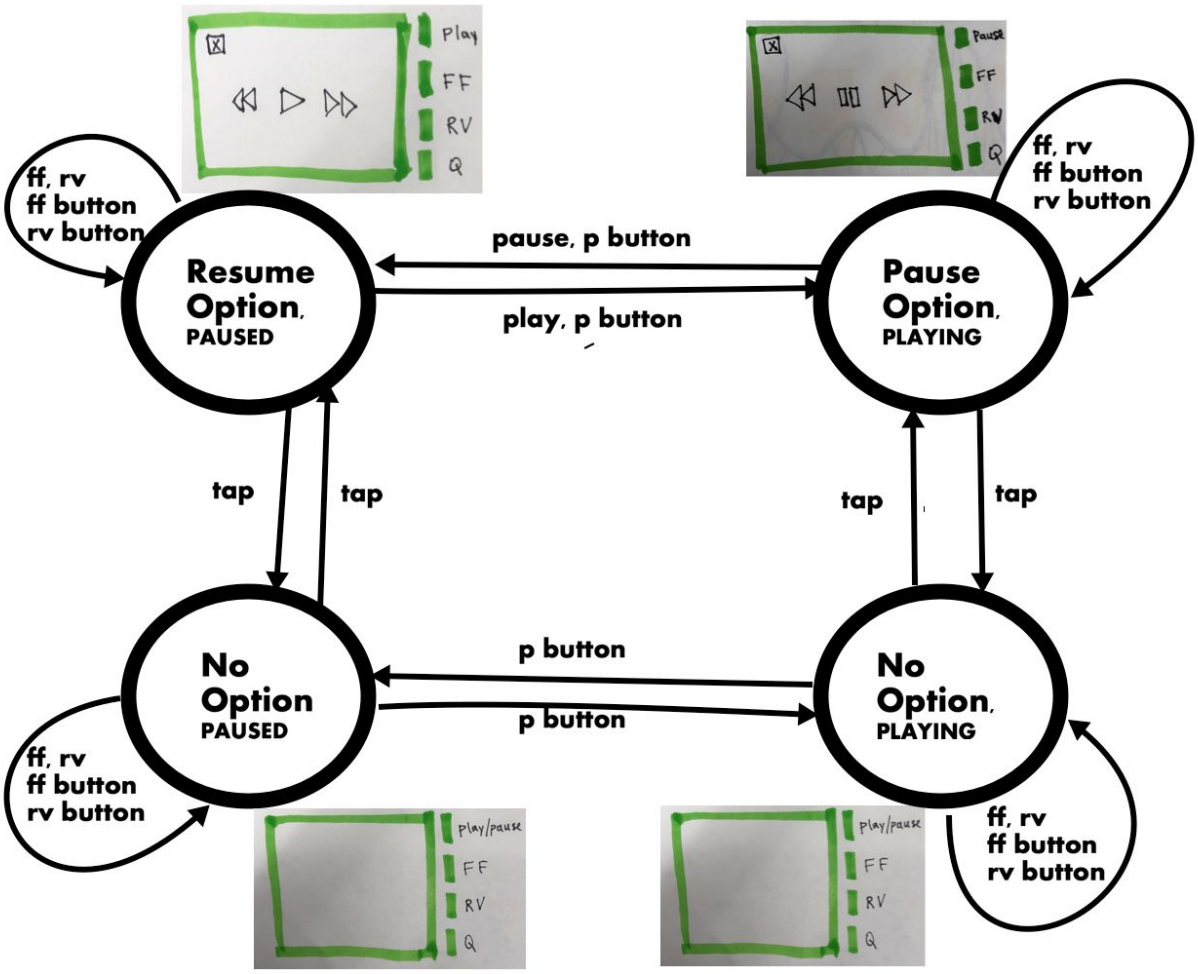
   To establish a touch screen button, a rect would be used as the actual button area understood by the program, while a surface would be used to display the button image on the screen for the user. This is because rect has supporting helper functions to get bound information.

   Here is the process for animating 2 frames of an animation:



9. NEW VERSION: Using a callback routine saves a lot of CPU cycles whereas polling requires an active process to check each independent GPIO pin. Also, polling does not scale well with increasing number of external sources, because the logic for each poll iteration increases linearly. Callback routine uses interrupts and interrupt service routines to handle the incoming data from external sources.

   OLD VERSION: Logic board for the video player interface



10. If start_video is run without first creating the fifo video_fifo, a new file named video_fifo is created into which the commands are written. This file is not a fifo, however, so mplayer cannot receive the commands. The python code video_control.py still receives the button presses and outputs the commands to the file as text, but the commands do not reach mplayer, so it keeps playing the video.