Checkers Checklist

Commitment
Display Checker Board through VGA
- display b&w image from camera
- display color image from camera
- display generated board (like pong lab)
- display pieces on the board

Play against Computer
- create the checkers AI
- get connection over lab kit (USB communication)
- display the checkers from the computer on the display

Have game clock with button to move
- debounce signal
- divider module (to 1Hz for timer)
- game timer module
- display outputs from module

python to serial port (USB connection to Labkit)
- verilog code
- wiring on labkit
- computer driver/script

Goal
Have physical Checker board
- create pieces and board on the wall (red and green)
- process correct frame (set up camera to view board, correct height and dimensions)

Project computer pieces on board
- process board configuration signal to board layout
- filter computer’s pieces
- assign correct pieces (kings or not) to each space
- show image on display
- project the compiled image

read in human pieces through camera
- detect colors
- filter
  - downsample
  - MSB’s for colors red and green
- find where the player has checkers
- translate to board configuration signal
play through to kinged pieces, different colors
  - detect green and red
  - assign different bits to different colors (in sent board configuration)
  - handle the different capabilities of each piece

**Stretch Goal**
synchronize camera and projector each time
  - get frame for projector
  - make button signal to know it’s synchronized
  - system for aligning camera with the board
serialize moves, as opposed to 96 bit string
  - serializer module
  - de-serializer module
checkers program features (undo, suggested move…)
  - undo button
  - undo software
  - suggested move button
  - suggested move calculation
  - suggested move arrow/display
fancy checkers pieces
  - save into memory
  - recover from memory to display