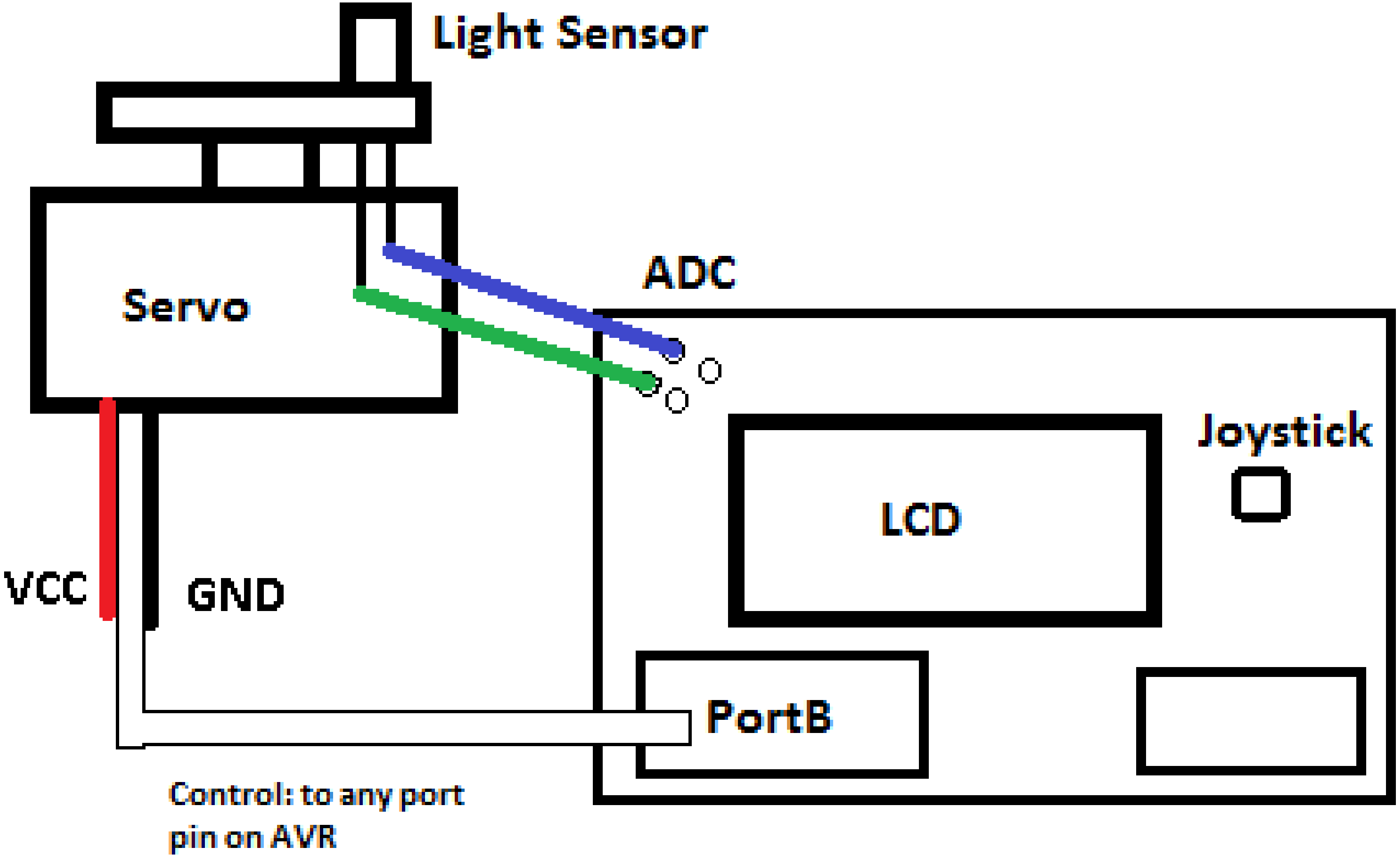
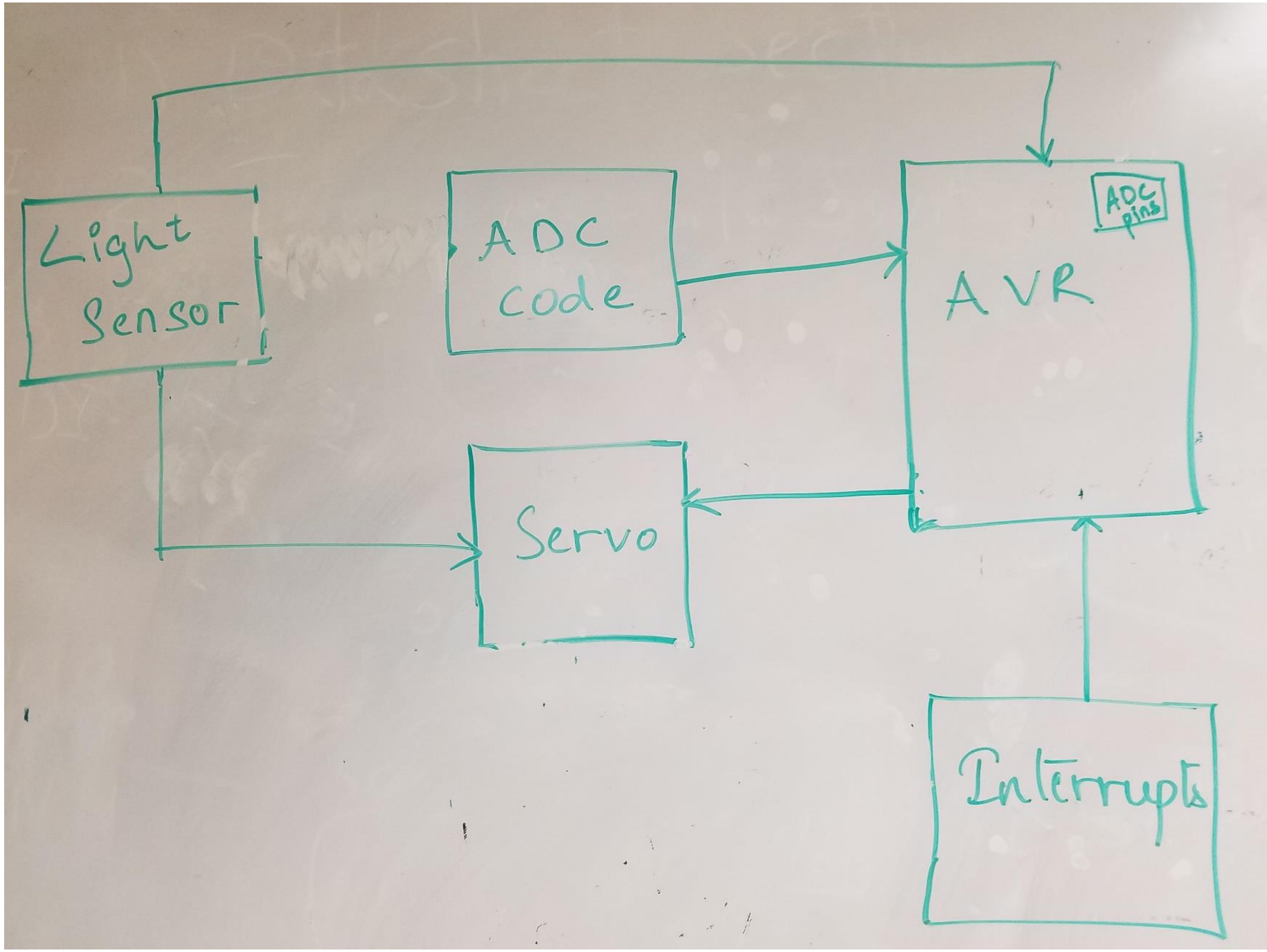
HW Pinout connections



HW4 Block Diagram



HW4 clarifications

- 1. There are multiple parts that they must get it working: 1.Read light from ADC, 2.use interrupts, 3. use timer and then do the sweep. Where should they start from?
- Ans: They should start on interrupts, as it is the basis for the project (inputs define the mode and the start). ADC code is also easy and available (discussion 9) and mostly complete, they just need to add/modify a couple lines. This can be found on both the Piazza post and the homework prompt.
- Create state machine with functional test interrupts. Then separately work on the ADC and motor code. Once those are functional, insert them into the test interrupts

- 2. Why do you need interrupt?
- Ans: Interrupt is for pressing right on the joystick, to initiate a full sweep (can be done at any time). This allows the processor to save resources and only go into code upon an interrupt trigger.
- 3. Also, why can't you use software counter instead of hardware timer (or at least initially to get other parts working). Is the timer used to slow down clock for sweeping?
- Ans: The use of the hardware timer seems to be there just so they learn a new concept. It is a good thing for them to understand. Software timers could also make this difficult to do with the interrupts, as the interrupt code will be independent from their Main code. The timer is used to slow down the clock to allow the motors to function.
- They may use the software counter. A project used no timers whatsoever, instead the delay functions for pulse width modulation. That though was mainly because that is what the blue servos required.

HW4 clarification

- What is the whole flow ?

The system has two modes: follow the light (FTL) or avoid the light (ATL). Pressing up or down changes between these modes. The LCD (driver code available in discussion 6) should reflect the mode (FTL or ATL). Pressing right will start a full sweep. While in full sweep behavior, the servo will move 180 degrees and the light sensor (through the ADC) will take light measurements every 20 degrees. The LCD should also display the angle of most/least light in degrees. When done with full sweep, the system will go into local sweep behavior. Local sweep behavior moves +/- 10 degrees from its current position taking light measurements at both points and moving to the point with the most/least amount of light. Again, the LCD should display the most/least light angle in degrees. The end result should look like the servo arm moving back and forth and the arm with the light sensor either following or avoiding the light, while the LCD displays angles in degrees. It should also be noted that at any point, the local sweep can be interrupted by pressing right on the joystick, which will put the system back into full sweep behavior.

HW4 clarification

- The servo in the prompt was supposed to use the PWM (specifically timer0).
- Example (this could be 180 degree for your servo, this is just for example use):
- 184 degrees is because that servo moves in 23 degree increments, instead of 20 (23 * 8 = 184). 500us and 2500us was found from experimentation with the blue servo and from reading its datasheet.
- The frequency to do this was found to be 500 us to move counterclockwise ~23 degrees, and 2500us to move the servo right ~23 degrees. This means in 8 movements in one direction, the servo moves 184 degrees. Interrupts are implemented into the system,
- The full sweep will sweep a full 184 degrees, taking light readings at every point. After the full sweep is complete, the servo will move to the point that had the most/least light and will start a local sweep. When doing a local sweep, the servo (assuming it is not at 0 or 184 degrees) will sweep clockwise and counterclockwise 20 degrees (if at 0 or 180 degrees,