June 2011 – Model Solution

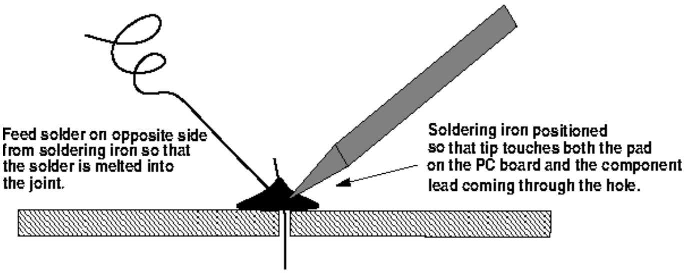
1a. An input transducer takes an input from the real world, and converts it into an electrical signal. A microphone, for instance.

1b. An output transducer uses electrical signals to produce an output in the real World, for example, a stepper motor.

2a. Feedback in a closed loop system is where an output at the end of the process is fed back into the input to inform or modify the process.

2b. Oscillatory motion is reciprocating motion along a non-linear (often curved) path.

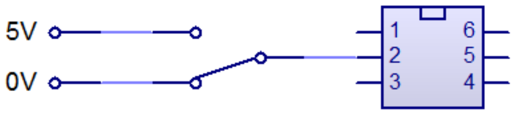
3. Soldering components to a PCB is one method. Prior to soldering, The track needs to be clear of oxidation (it can be scrubbed with wire wool). The component should be pushed through the drilled holes, and the legs bent out to 45˚, so they stay in place. To perform the joint, the soldering iron is placed so it is in contact with both the pad and the component leg for 3s. Solder is then fed slowly into the joint, and finally, the iron is removed.



4a.

A	B	C	Q
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

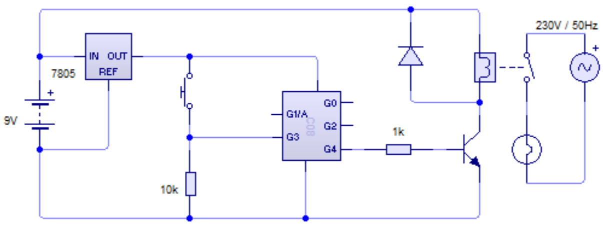
4b.

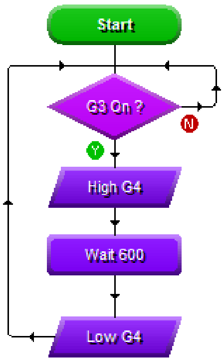


4c. Unused inputs to logic gates need to be grounded, so that the pins aren't left floating, which would cause random outputs and unpredictable behaviour.

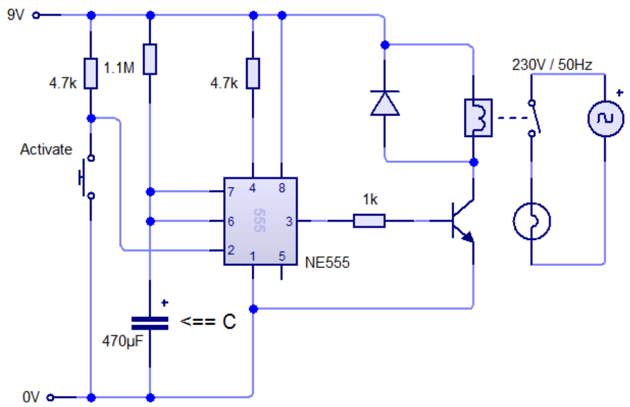
5a i. *(Option Question)* System 1 is to use a PIC chip.

Connected to a relay, this could be programmed to provide a 10 minute high when the PTM switch is triggered.





5a ii. System 2 is similar, but uses a 555 timer set up in a monostable configuration. To calculate a 555 monostable, the formula is T = 1.1 * R * C

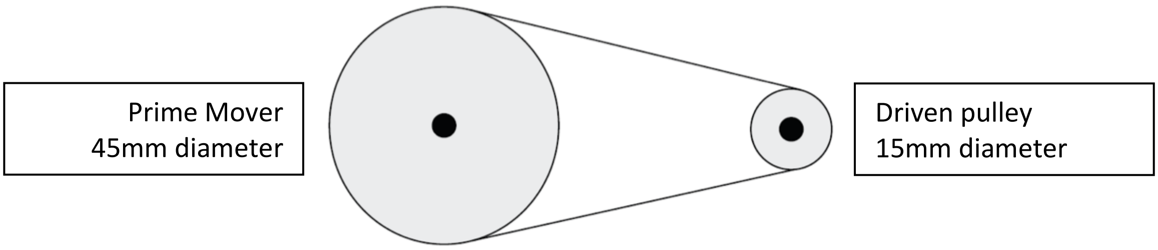


600 = 1.1 * R * 0.000470
600 = 0.000517 * R
600 / 0.000517 = R
R = 1,160,541.6Ω (Roughly 1.1MΩ)

5b. To provide a variation from two to thirty minutes, the PIC solution would be easiest to adapt. In order to do so, a 10k potentiometer could be added to an analogue input in the PIC, and the program modified to first read in the value from the 8-bit ADC. As the values will range from 0-255…

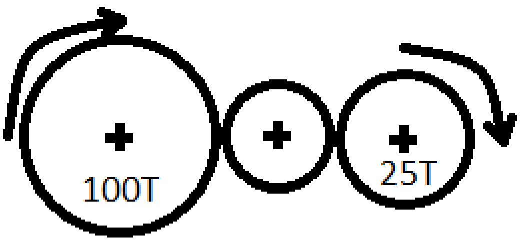
255 / 28 = 9.1, so the program would need re-writing to take the reading, and divide it by 9, then use this value multiplied by 60 (seconds) to get the wait time.

6a i. *(Option Question)*
System 1: Belt and pulleys. This method uses friction between the pulleys and belts to transmit rotary motion to linear then back to rotary.
In order to accelerate the motion, a smaller diameter output pulley should be used in comparison to the input one.



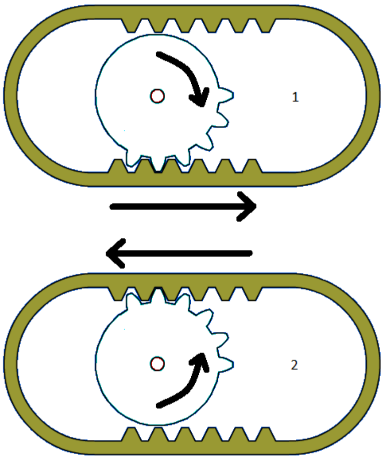
In the example sketched above, the driver has a 45mm diameter pulley, connected to a 15mm pulley. 45:15 = 3:1 ratio. So in this case, the torque is reduced three times, but the speed is amplified three times over.

System 2: Compound gear chain. This uses friction between the teeth of the gears in the chain to transmit the drive from one shaft to the next.
In order to accelerate the motion, a smaller number of teeth is placed on the driven gear. The number of teeth on the idler gear is inconsequential, but its presence has the effect of preserving the direction of the rotary motion.



In this example, there is a 100:25 = 4:1 ratio between the input and output gears. As such, the output will be four times faster than the input. Using double gears would allow for even higher ratios to be achieved, as the individual ration pairs are multiplied together.

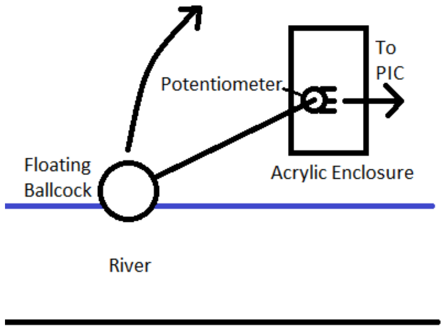
6b. *(Option question)*



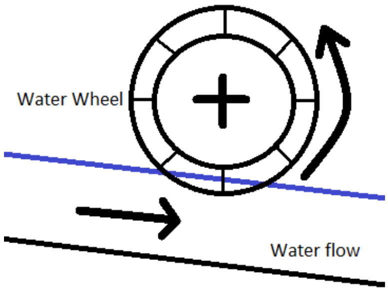
By using a rack and pinion reciprocating linkage as shown in my sketch, reciprocating motion of the notch cut into the sliding link (shown in green) can be converted into clockwise motion.

As the slider moves to the right (labelled 1), the gear rotates clockwise as the teeth move around until the end of the rack. At the end of the rack, the first tooth on the gear will be meshed with the top rack.

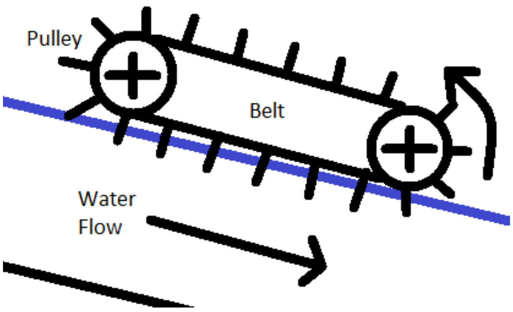
As the reciprocation starts moving left (2), the teeth mesh fully into the rack, and the gear continues to rotate clockwise.



7a. This design uses a floating ballcock attached to a potentiometer on an extension arm. As the water level rises, this will cause the potentiometer to turn so that the movement can be detected on the analogue input of the IC, and interpreted as an integer.

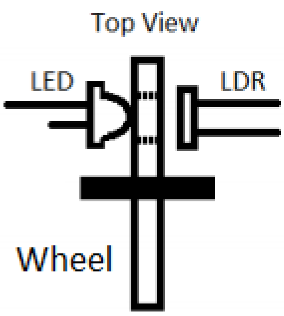


7b. Method 1. This design uses the linear flow of the water to power a water wheel, connected to a shaft (which will be the output). Each of the paddle-blades inside the wheel will rotate sufficiently to engage the next blade.



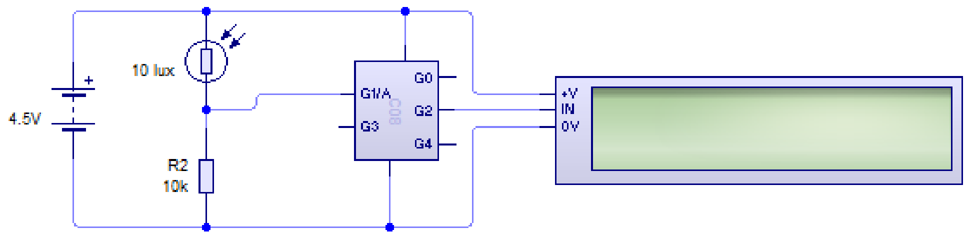
Method 2. This design uses a belt and pulley system, with perpendicular individual blades attached at regular intervals.

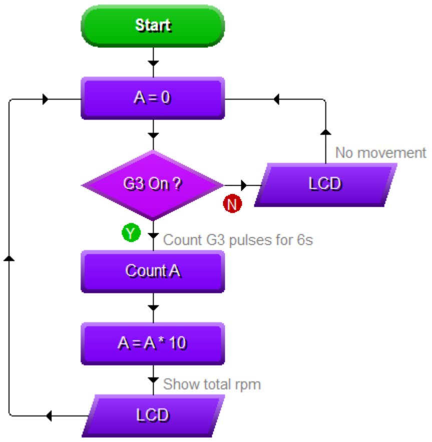
As the water flows past the blades, they will be pushed along, which in turn will rotate the output pulleys.



7c. This system uses a simple LED / LDR pair, whereby an LED (or laser) shines straight at the side of the wheel. The wheel itself has a single small hole drilled straight through one of its sides, so that once per rotation the LDR is illuminated. This signal can then be detected by an analogue input on a PIC.

7d. In terms of the control circuit, a simple PIC design will accept the digital inputs from the micro-switch (*expressed as a PTM in this diagram*). A program can then be written to count the pulses, and display then on an LCD display. Alternatively a series of 7-segment displays connected together with 7-segment decode ICs could have been used, but this solution requires less components, and provides a more attractive output.

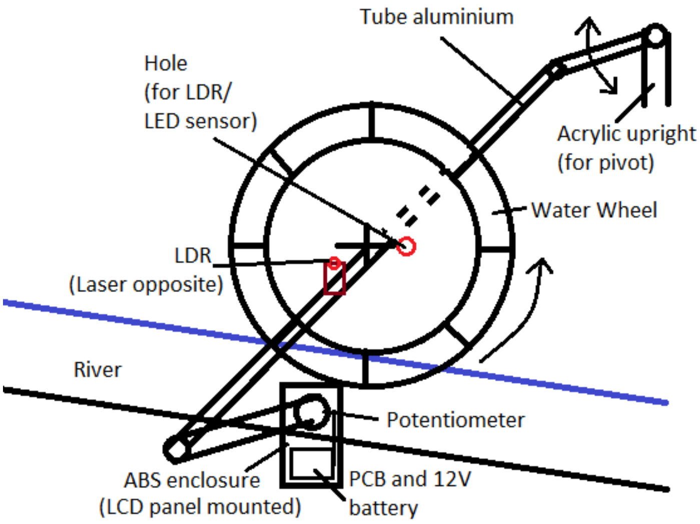




In order to display rpm, the program would work as follows:

1. If the input isn't triggered, display "No movement".
2. Once movement is detected, count the number of pulses from the micro-switch over a 6s period, and store in variable A.
3. Multiply A by 10 (to give an rpm value).
4. Display the value.
5. Reset A to 0, and restart the program.

7e. This solution combines the water level and speed into a single assembly. The water wheel will be made hollow (and from a light-weight material), so it will be buoyant. I could then connect this to the LDR/laser outlined in the previous question to simultaneously read the water height and speed. The wheel may need to be quite large to have this amount of buoyancy.



I would make the wheel from 1mm PVC, which I would water-jet cut, then bond together with a thermoset (e.g. Aryldite). The frame could be made from 1mm aluminium, spot-welded together (aluminium won't rust in water). I could also use aluminium for the pivot supports either side of the bank (or thick laser-cut acrylic). For the PCB/battery enclosure, I'd use laser-cut ABS, which would offer weather and impact resistance in case it is bumped by passers-by.

The system would be placed either side of the river bank. Prior to mounting, a hole would need to be excavated, and concrete poured to give a solid foundation. Rawl plugs and bolts could then be used to firmly secure the control side to the ground. For the far-side of the river, I'd sink my uprights into the concrete while it set to secure it.

To calibrate the water level, I'd need to simulate the height of the river level bar at different known positions (likely before fitting the wheel), and hard-coding these onto the PIC program so that they could be looked up as the level changed.

The only modification to my original circuit design would be to connect the potentiometer to the power rail, ground rail and the wiper to an analogue input.

To indicate the water level, the second line on the LCD display (I'd use a 16x2 display) could be used to show the water level to the nearest cm, and update it once every 6s when the rpm count is updated.

Additionally, I could include a warming LED/buzzer when the water level gets dangerously high, to notify members of the public in the event of a problem. I could also connect a 3rd party dialler, to send an SMS message via the mobile network to the Environment Agency in the event of unusually high flow or water level.