

Re: Sprinkler pump controller

Source: <http://sci.tech-archive.net/Archive/sci.electronics.design/2007-03/msg04831.html>

- *From:* ehsjr <ehsjr@xxxxxxxxxxxxxxxxxxxx>
 - *Date:* Sun, 18 Mar 2007 21:46:26 GMT
-

jfma@xxxxxxxxxxxxxxxx wrote:

I've had this idea for a while but don't know exactly how to go about it. I'm thinking of a micro controller of some sort but I haven't worked with micro controllers and not sure which one would be best for this project. I also want to make this project into something that I could use on a resume, something . I am an electronics engineering technician (AS-ET).

This is the basics of what I have now. I have a water retention system (55 gallon drums manifold together) that has an MPXV501GP (1.45 PSI) to detect the water level in the system. I have a rain sensor mounted on the gutter.

What I want to be able to do is to turn on the sprinkler pump after it hasn't rained for a number of days. I also want the pump to come on when the system level goes above a narrow high window and off again below that same window. I also want it to shut down the pump if it goes below a bottom window and able to come back on above that lower window when it hasn't rained for the preset number of days. Upper and lower windows could be detected with 2 op amps for the upper and 2 for the lower. I've done something similar to this before but was a number of years ago and I can't find the documentation as to what I did and parts I used. They probably don't make those parts anymore anyway.

The newest data book I have is a TI linear volume 3 from 1992. Do they have hard copy data books any more or they all on CD?

"Program" it first at the functional level, before choosing the ucontroller. Two conditions do not "go through" the ucontroller: When the low window bottom sensor is dry, the pump is off regardless of any other conditions; when the high window high sensor is wet, the pump is on regardless of any other conditions. That said, you still might want to have the uc look at them for errors, like the low window

Re: Sprinkler pump controller

low sensor dry but the other window sensors wet. Once you have laid out all the functions you want you should be in a better position to pick the uc. But it really doesn't matter, as you indicated that you haven't worked with them.

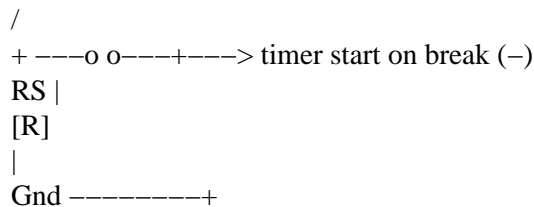
So, what you really want is someone to provide a complete solution, with the uc already chosen and the code written, or someone with experience to tell you which uc to use so that you can then learn how to use that particular uc family. That experienced person will be better able to meet your needs after you have spelled them out. For example, you did not address resetting the timer, error checking the sensors, what to do if the pump is running and it starts raining, whether the timer needs to run during a power failure, what the power on reset state should be and who knows what else.

I'm having a hard time figuring out how this project demonstrates the need for a uc, but I think you could demonstrate the parts cost advantage of one.

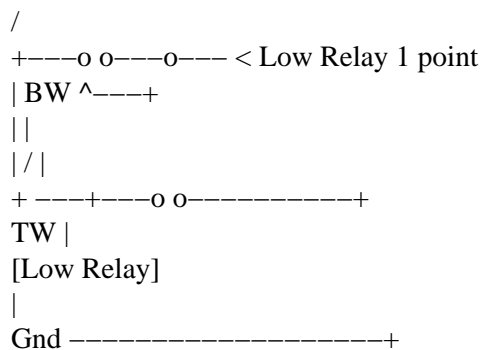
I'll show a hardware solution below. The purpose is really background to show that the person reading the resume may think "why in the hell is he showing such a simple use of a uc, when so much more could be done?"

Excluding the timer, a simple hardware solution is a couple of relays. I'll draw the comparator outputs as switches (saves drawing the op amps) All switches made when wet.

Your rain sensor (RS – on when wet) :



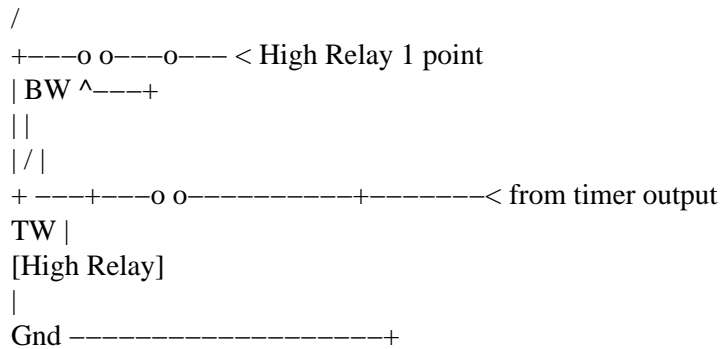
Your low window level top of window (TW) and bottom of window (BW) sensors (on when wet):



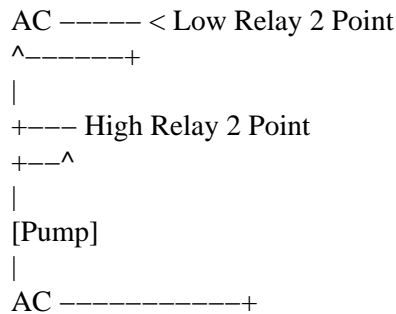
Re: Sprinkler pump controller

When the tank is filling, BW closes first as the level rises. When the water reaches the top of the window, TW closes and the relay energizes. It stays energized, if the water level drops, until BW opens.

Your high window level top of window (TW) and bottom of window (BW) sensors (on when wet):



Operates the same as the bottom window sensors. In addition, the relay can be controlled via the timer. That meets your condition of the pump operating when the timer calls for it, regardless of the status of the upper window sensors.



The point of all that is to show that the uc is not providing a whole bunch of function that mandates its use over hardware. You want the resume to sell a product – you – to the company.

Maybe it would be sufficient for your project to use the uc as the basis for your timer – I don't know. It seems to me a waste of uc power to use it only as a timer in this. Neither the hardware nor software solution can be given until timer reset is defined, but the hardware solution might be a counter like a CD4060

Re: Sprinkler pump controller

and support components or maybe cascaded ICM7242's.
I think it would be a better project if you could do all the "thinking" with a uc – and show enough need for "thinking" that the person reading the resume is impressed. As stated, your project doesn't do that, in my opinion.

Suggested additions: add zones, with individually selectable on times like 4 hours every other day or so many minutes daily or once a week or whatever, sensor logical error checking, maybe an "it's been too dry for too long" alarm to a human operator, etc. The more practical flexibility you put into it, the better.

Ed