General Purpose Sensor Activated Relay Controller with LED Display.

Background

This project uses the 24-pin BS2-IC as the brains of a general purpose, sensor driven relay controller with a 4-digit LED display.

A sensor is used to control up to four "event point" relays by using the BS2. For example, an RPM sensor could be programmed so that each relay provides a difference function at a difference speed. The same may also be done with sensors for humidity, pressure, distance and so on. The main restrictions, with an analog sensor, is that it put out no more than +5vdc and, if it requires an operating voltage, then that voltage has to be +5vdc. If other than +5vdc is required for the sensor (such as for the Vcc), then it must be supplied externally or from the optional LM350 regulator (more on this later).

As for digital sensors, since their output values are less "generic" than 0 to 5v analog sensors and have their own protocols, I'll leave their interfacing up to the user. I have, however, used the demo project with the DS1620 digital temperature sensor and it works fine. I have to admit that I tested the DS1620 by simply heating it up with my 53-year-old Weller soldering gun (pretty crude, I know), but its operation is the same as the LM34 analog sensor. You'll find the code attached for the DS1620 as well as for the LM34.

In addition to the relay board, a 4-digit LED display can be connected so that you can monitor the input that is controlling the relays, such as RPM, temperature, voltage and such.

Temperature Alarm Demonstration Project

The demo project, described here, is a temperature alarm with four programmable event point relays and a four digit LED display. In this example, the input is supplied by the LM34 analog temperature sensor. I used the LM34 because it's easier to mimic its analog output then the DS1620 digital output; they both work the same, however. Keep in mind that the LM34 puts out a linear 10mv per degree F, so that at a temperature of 100F its output voltage is 1000mv, at 150F its output voltage is 1500mv and so on.

Note: The LM34 used here can read up to maximum of 300F, but others in the LM34 family may have a lower maximum operating temperature, so be careful with the LM34 that you use.

Here, I'm reading from 0 to 5.0 VDC or 0F to 500F, (but really limited by the LM34 to 300F); however, as mentioned above, just about any sensor that provides 0 to 5 VDC over the range the user is interested in could be used. For example, 0 to 5 VDC might be RPMs; pressure; humidity; MPH; or simply 0 to 5 VDC for voltage. As for the four "event point" relays, they could be used to control up to four difference functions that are to occur from 0 to 5vdc; they don't have to be alarm functions. The point I want to make is that this project may be adapted for uses other than as a temperature alarm, and the key to that is using an analog sensor that puts out 0 to 5vdc over the range you're interested in. Digital sensors, as mentioned, have their own quirks and I'll leave them up to the user, but there's no reason why they can't be used with this Relay Controller and LED Display.

OK, lets see how the temperature alarm works.

Relay Board.

Note: (See the attached code and the schematics for both PCBs)

The Basic Stamp outputs P6 through P15 are programmed to go HIGH starting with an input voltage from the LM34 of 1100mv. This 1100mv, of course, represents a temperature of 110F. As the temperature increases in 10F increments, each pin, in sequence, goes HIGH until BS output P15 goes HIGH at 200F. Naturally, the temperature range as well as the temperature at which a BS output (P6 through P15) goes HIGH may be changed in the code. In this demo, four of the 10 BS outputs are used to control four relays……… more on this a little later.

Also, in order to monitor the input, I have used a 4-digit LED display so when the LM34 outputs 1100mv (which will be at a temperature of 110.0F), the LED display will read "110.0".

Anyway, as the voltage from the LM34 increases (due to increasing temperature) each event point pin (the event point pin numbers are not the same as the BS output pin numbers), starting at pin 1 (110F) will go HIGH -- in 10 degree F increments -- and stay HIGH until the pin 10 goes HIGH at 200F. As the temperature (i.e., sensor voltage) goes down below 200F each pin in sequence, and in 10-degree increments, starting at pin 10 and continuing to pin 1, will go LOW. Of course, these temperature points are easily changed in the code and could be anywhere from 0F to 500F, limited by the sensor used and the Vref.

Now, if you look at the relay schematic, you will see four DPDT relays. Each relay is controlled through an NPN transistor which has a "flying lead" connected to its base resistor. Each of the four flying leads may be connected to any four of ten event point connection pins which will go HIGH as described above. The relays may be used to control audible alarms, visual alarms, fans and such. Of course, the relays may be used to control anything within their rating; they're not limited to alarm functions only. The relays are rated 1A at 30VDC so heavy loads would need to be switched through power relays, solid-state switches (SSS) or such. A SPDT output is available from each relay on terminal blocks TB1 through TB4.

As an example: For use as an alarm one relay may control a visual alarm that goes off at, say, 130F. The second relay may activate at 140F and control an audible alarm. The third relay may, through an external relay or SSS, control a cooling fan, which turns on at 150F.

As the temperature goes down, each alarm will deactivate in sequence. So, when the temperature goes below 150 deg, the fan turns off; when below 140 deg, the audible alarm turns off; finally, at 110 deg, the visual alarm turns off. No doubt the sequence in which the alarms deactivate can be changed in the code so that, say, the audible alarm turns off before the visual alarm and the fan turns off last; but this is something for another day.

WARNING: Having various devices turning OFF and ON automatically may create an unsafe condition. The individual user will have to determine whether or not unsafe conditions may occur and then take the necessary precautions to avoid them.

In addition to the above, when any of the four event points are reached, a corresponding red “event” LED will illuminate on the control panel to show an event condition. Any of the four event point relays (and any device controlled by that particular relay) may be disabled through four toggle switches on the control panel. However, even if a particular event point relay is disabled, its corresponding red “event” LED will still light indicating an event condition exists even though no event relay is activated. When an “event” is disabled as above, a corresponding yellow “caution” LED comes on to remind the user that that particular event point relay has been disabled and will have to be reset manually as required. This disable feature may be used to prevent unwanted operation of a particular function (such as a fan) while at the same time giving an indication (by means of the red event LED) that an event point has been reached.

There is a DB9 for programming the BS2 and also a 3 position terminal block to give access to 3 I/O pins for a digital sensor input (with the 0831 removed, of course). If you need access to more than 3 pins for a digital sensor you can connect “flying leads” to the relay connection pins as needed, but you’ll have to give up some “event points”.

Now, a description of the power supply.

You will see two voltage regulators here. One, a LM350, supplies a reference voltage to the 0831 A/D/C if you’re using other than 5.0v as Vref, or, alternately, it could be used to supply Vcc to a sensor that uses other than 5v. This regulator can be left out if you’re using 5.0v for Vref and use a Vcc of 5v. (I recommend that you read StampWorks Experiment 29 for a description of the 0831 reference voltage and how to use it.)

The second voltage regulator, the LM1084-5, supplies 5 volts for the 0831 Vcc, 5.0 volts for Vref, the LM34 sensor, the relay circuits and also for the LED display (on a separate PCB). This regulator is pre-programmed for 5 volts and is rated at 5 amps. Total current draw from the 5 volt regulator indicates a heat sink will be needed here and a standard TO-220 sink with fins will do fine.

Power for the project is supplied by a 9 VDC wall-wart rated for 1 amp.

LED Display

The LED Display is pretty straightforward, although it can be a little tricky to get the MC14489 LED driver to do what you want it to. I recommend that you look through Experiment #31 in the StampWorks book for more information on this. As mentioned, the 5-vdc for the LED Display comes from the 5-amp regulator on the Relay Board.

By the way, the LED display may be used stand-alone for projects other than described here. It's pretty versatile once you get the hang of setting up the MC14489 and it does provide up to four digits. It's also a lot easier to read than the typical LCD display.

Does It Work?

Attached is a picture of my breadboard setup (yeah, I know……it's pretty gross). The 24-pin BS2 is on one breadboard as are the MC14489 LED driver, 0831 A/D/C, 4-digit LED display and the two voltage regulators (LM350 and the LM1084-5). You may note a small blue pot on the board. This is used to generate an adjustable test voltage (0 to 5.0v) that mimics the voltage from an analog sensor and is handy for testing purposes. The small vertical board that's plugged into the DB9 is Parallax's USB to serial converter used for programming the BS.

The second breadboard has the four relays, relay and LED driver transistors, four red event LEDs and four yellow caution LEDs.

The PCB off by itself to the left has a DS1620 digital temperature sensor on it and is used to connect the DS1620 to the breadboard for testing.

Additional info.

Most of the ideas regarding the Basic Stamp, programming the BS, interfacing the MC14489 to the LEDs and using the ADC0831 came from Jon Williams' StampsWorks manual from Parallax. I especially recommend going through Exercises #28 and #31 in the manual for more detailed information and also Exercise #29 on using the DS1620.

Many of the parts mentioned are available from Parallax, such as the USB to serial converter, LM34, ADC0831, 2N3904, DS1620 &c. Other items are readily available on the Internet.

The schematics were drawn using DipTrace v1.5, Extended Edition.