Chris Staymates
EEL5666

Special Sensor Report

Sensor: Hamamatsu UVTron Pyro Detector with Driving Circuit
Source: Acroname Products
Website: http://www.acroname.com/robotics/parts/R67-UVTRON.html
Total Cost: $70.00

Introduction

When trying to sense flames there are many considerations to take. The least of which is not how to differentiate the heat generated by a fire, from that say of a human. There are many sensors on the market that do an adequate job of detecting heat sources but cannot make the differentiation. An example of such a sensor is the Eltec 442-3 heat sensor. However, I settled on purchasing the Hamamatsu UVTron from Acroname due to its ability to truly detect flames. This sensor is also available with a drive circuit that condition the output of the bulb to a pulse train, as well as a shield that limits the viewing angle of the bulb.

Operation

The UVTron differentiates heat generated by humans and fire by detecting the wavelength of the ultraviolet light emitted from the flickering flame. Figure 1 shows the spectrum of UV light in the atmosphere. The first shaded area marked UV Detection Range is where the Hamamatsu operates (185 to 160nm).

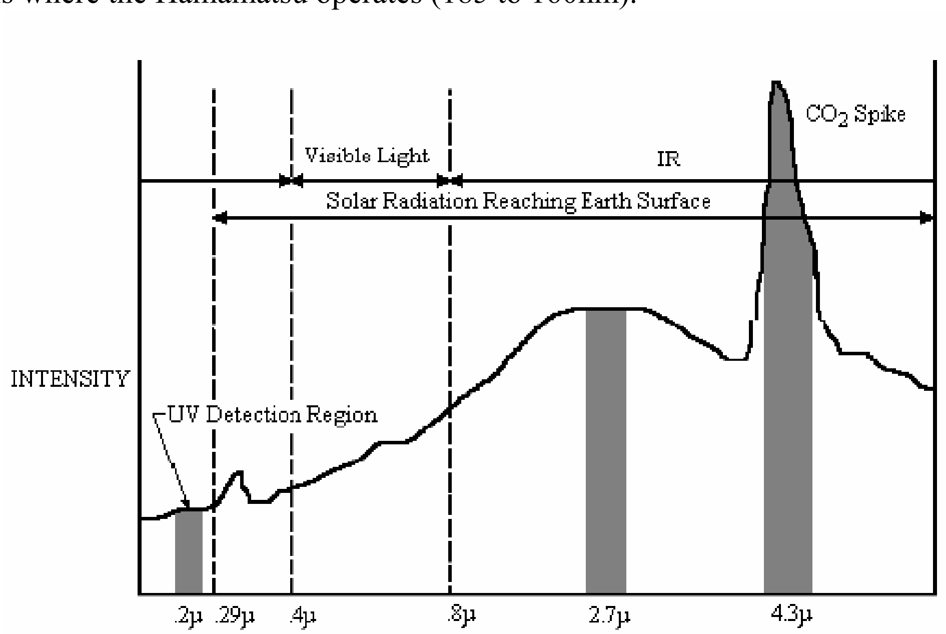


Figure 1: spectrum of UV light

http://www.scottinstruments.com/custom_file_uploads/Guide%20to%20Flame%20detector%20Selection.pdf

The detection process involved the excitation of photoelectrons on the cathode side of the UVTron bulb to the anode side. The excitation of the photoelectrons is caused by the ultraviolet light and are accelerated due the electric field generated. This process creates a current across the bulb and is the basis of the voltage discharge associated with the bulb. (*Please note that complete details on the operation of the bulb can be found in the Acroname documentation for the part.*)

Interfacing

Power
Getting power to the drive circuit is also a very simple task. The circuit can regulate power from 10 to 30V using an onboard 7805 regulator. This power is hook up to the + pin on the board. Ground is wired to the – pin. By soldering a single header onto the hole marked O you can give the board pre-regulated power.

Output
The drive circuit that accompanies the bulb converts the electric discharge into 10ms pulses. These pulses are available on pins 1, 2 and 3. Because a pulse is generated when a bulb event occurs, it is not useful to measure the distance between pulses as this time will vary. The drive circuit also offers an inverted output as well as the non-inverted output. Both are shown in Figure 2.

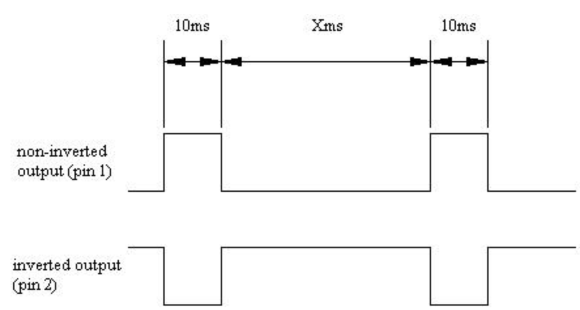


Figure 2: Drive Circuit Outputs

There is one more output available from the drive circuit that is a open-collector output (pin 3). This means that it can drive different voltages rather than 0 and 5V. A pullup resistor in nesseccary for using this output. Placing a capacitor across the pins marked CX will lengthen the pulse. See Table 1. The maximum pulse width is 100s.

Lengthening the pulse with a capacitor	
Capacitor value	Pulse Length
1uF	1s
10uF	10s
100uF	100s

Table 1: Lengthening the pulse

Coding for the UVTron

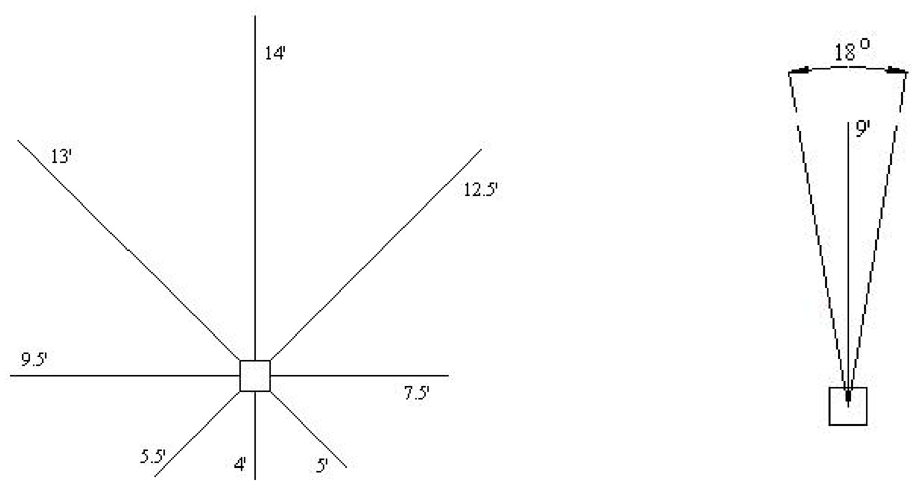
Since the UPTron's pulse train is an output of the drive circuit and an input to the micirprocessor, it makes sense to code for the pulse with an input capture pin. Since there is no way to determine the time between pulses, it soon became difficult to detect a pulse train with any accuracy. Howeverm I soon realized that I do not need to measure then length of any pulse.

Hooking up the output of the circuit to an oscilloscope, I determined that although there is no set distance between pulse, as a long a fire is present, a new pulse always occurs at least 70ms after the last one. I could have used this value in measurement, but I realized that as long as I receive another pulse before this time is up, the fire is still there. I then set up a timeout on a timer unit on the microprocessor. If a pulse arrived before the time out expired then I reset the timeout, else, there was no fire.

The code is completely interrupt driven so the program merely samples a global variable to determine if there is a fire. My test code for this application is shown in the appendix of this report.

Specifications / Verification

The UVTron is specified to be able to detect flames at a distance of 5 meters in all directions. The following data was collected to determine the range of the sensor in my application. Figure 3 shows a direction view of the experimental range of the sensor with the viewing restrictor removed and on



(a) (b)

Figure 3: Range of sensor with (a)shield removed and (b)on

Appendix – Code Listing

```
//global variable to determine if there is a fire
int fire, numover;

//this interrupt handler handles the case when a timeout occurs
//resetting the timeour occurs automatically
//it changes the global variable fire to false
INTERRUPT(SIG_OVERFLOW3){
	cli();
	if (numover == 15)
	{
		fire = 0;
		numover = 0;
	}
	else
		numover++;
	sei();
}

//this interrupt handler handles the case when as edge is detected
//it should reset the timeout by clearing the number of overflows
//it changes the global varable fire to true
INTERRUPT(SIG_INPUT_CAPTURE3){
	cli();
	numover = 0;
	fire = 1;
	sei();
}

void init_fire(){
	DDRE = 0x00;
	ETIMSK = 0x24; //00100100 enable overflow, and input capture interrupts
	TCCR3A = 0x83; //10000011 set to zero on compare A match, fast PWM TOP = OCR3A 16 bit
	TCCR3B = 0x4D; //01011101 look for rising edges only, clock scaled by 1024
	OCR3A = 0xFFFF;
	fire = 0;
	numover = 0;
}

void main()
{
	LCD_init();
	fdevopen(LCD_sendByte,NULL,0);
	init_fire();
	sei();

	while(1)
	{
		LCD_clearScreen();
		if(fire)
			printf("Fire");
		else
			printf("None");
		test_delay();
	}
}
```