In this tutorial, I want to describe simple Infrared (later IR) obstacle avoiding scheme. It might be interesting to those who are just entering amazing world of robotics and wants to know more obstacle sensors.

There are many ways to realize remote object detection. The simplest is just IR Light Emitting Diode (later LED) and phototransistor. IR LED emits light with wavelength approximately 850 nanometers. Light reaches obstacle and reflects back. There it is picked up with phototransistor. Voltage in measurement point changes (fig. 1), and this change is proportional to picked up light intensity. Comparator or ADC might be used to convert it to digital form and decide, if there is an obstacle in front of us. For example if 5 Volts supply is used: 4.5 Volts mean obstacle is far, and 1 Volt means obstacle is close (as phototransistor current flows, voltage dropout on resistor grows according to Ohm's law, and voltage in point of measurement falls).

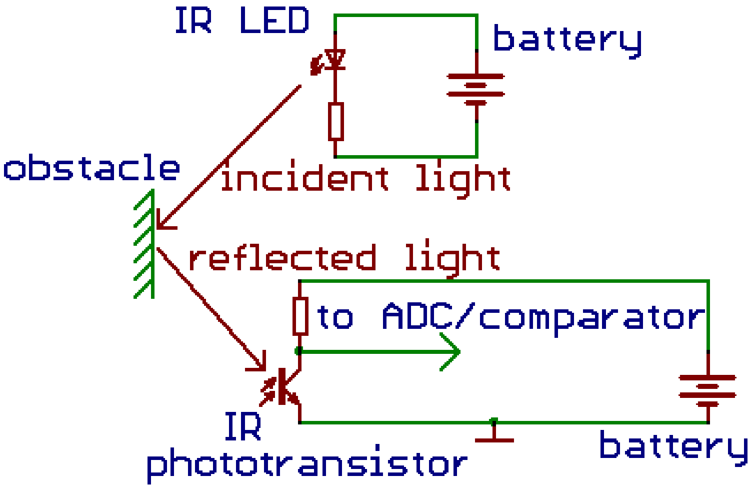


Fig (1). Simplest IR bumper schematic.

This schematic has big drawback. First, the Sun is powerful IR source, so there will be false readings on sunlight (fig. 2). Also any light bulb emits IR light intensely, so false readings again.

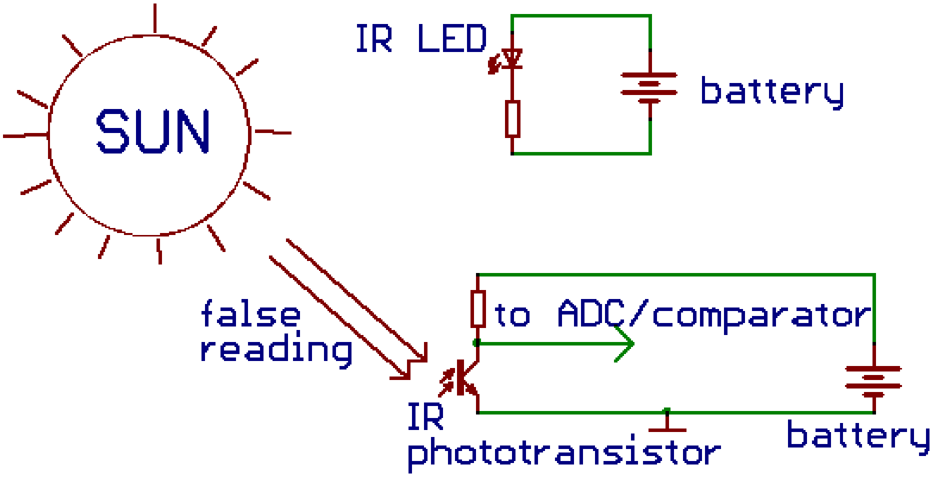


Fig (2). Sunlight impacts on IR phototransistor. False reading occurs.

There is one way to partially cut-off ambient light with color filter and let only LED light to reach phototransistor. The simplest filter it a piece of color photo film, exposed to fluorescent light, and then developed. Film would look black, but it let light with wavelength 820-1050 nm to pass through. However, even with such improvement this schematic lacks of stability.

There is better approach to improve the design. We can modulate IR LED like shown (fig. 3). Just turn LED on and off to do this. NPN transistor is used not to draw too much current from microcontroller (later MCU) pin, as it can be above 50 milliamps during impulse.

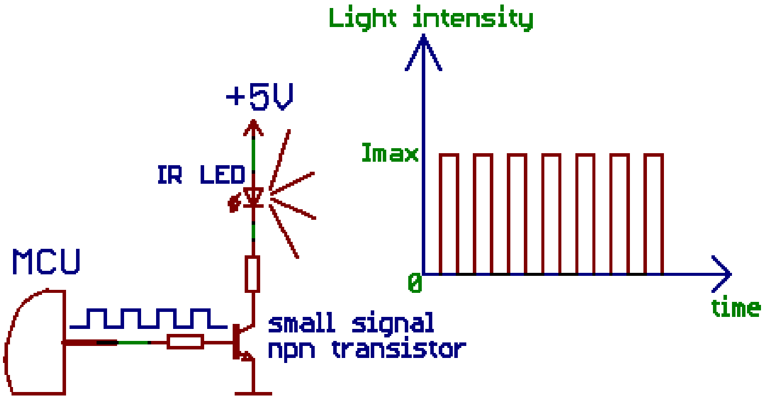
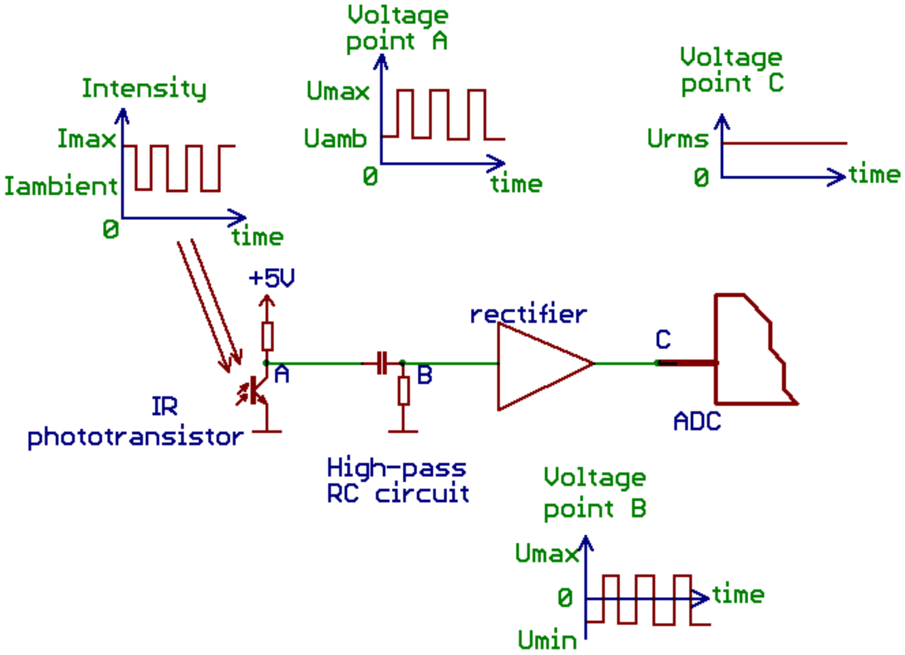


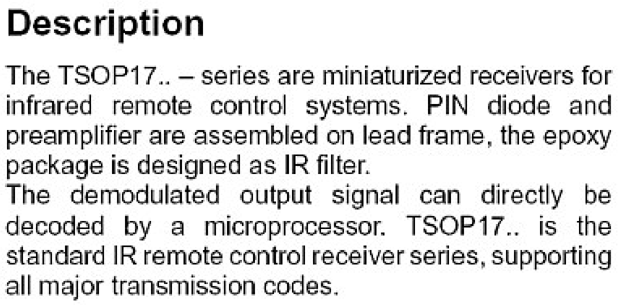
Fig (3). Modulation of light intensity using NPN transistor

Light intensity will be modulated, and voltage at the output of receiver will also be modulated. Now, ambient noise is directional current (later DC) signal, but transmitter signal is alternating current (later AC). Capacitor is used to block DC and get voltage proportional to AC only. After it, signal is amplified and rectified to make it suitable for MCU recognition (fig. 4).



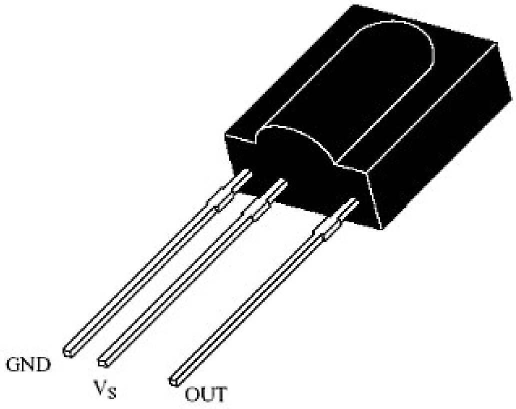
Fig(4). Improved IR bumper schematic

This way, non-changing ambient light can be rejected. Looks complicated, isn’t it? However, all this is implemented in integrated circuits (later IC). They are conventionally used as receivers in remote control.
In this tutorial TSOP1736 integrated circuit will be described.

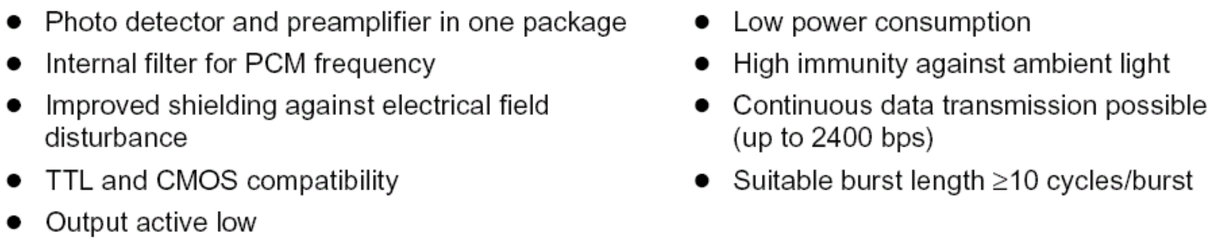


Description

The TSOP17.. – series are miniaturized receivers for infrared remote control systems. PIN diode and preamplifier are assembled on lead frame, the epoxy package is designed as IR filter.
The demodulated output signal can directly be decoded by a microprocessor. TSOP17.. is the standard IR remote control receiver series, supporting all major transmission codes.



It also provides several features:



- Photo detector and preamplifier in one package
- Internal filter for PCM frequency
- Improved shielding against electrical field disturbance
- TTL and CMOS compatibility
- Output active low
- Low power consumption
- High immunity against ambient light
- Continuous data transmission possible (up to 2400 bps)
- Suitable burst length ≥10 cycles/burst

And it’s block diagram:

Block Diagram

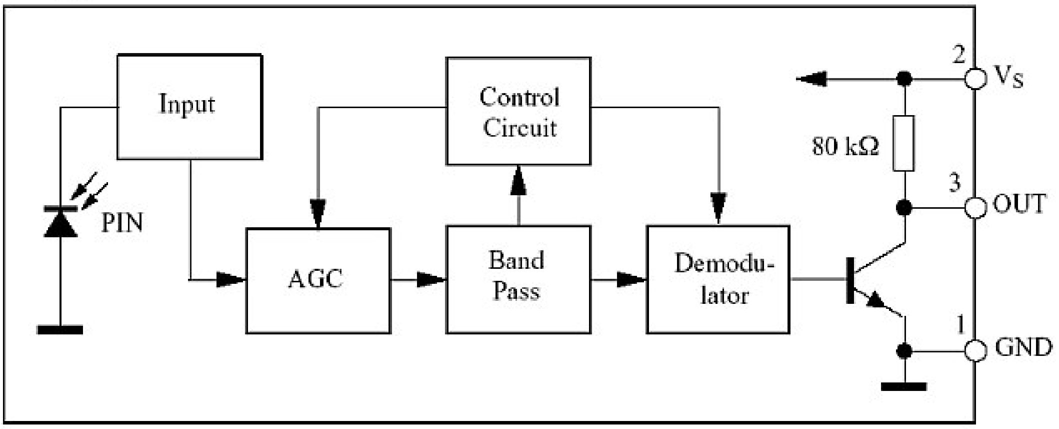
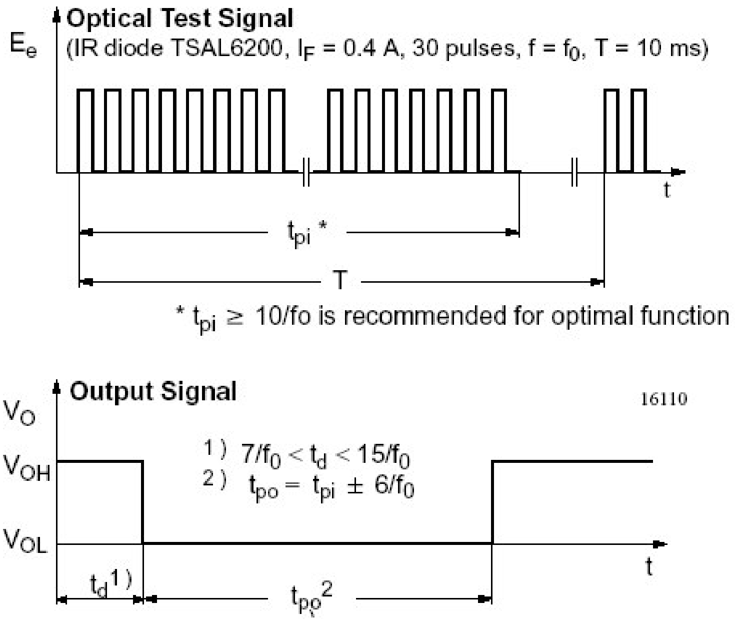


Fig (5). TSOP17.. block diagram

As you can see, it looks quite like schematic on fig. 4. However, instead of rectification, demodulation with band pass filtration is used. TSOP 1736 sense 36 KHz square wave light intensity, and there are other frequencies available. This prevents different remote controlled devices to interfere with each other as they work on different frequencies.

As TSOP sense 36 KHz modulated light, it turns output low. It is because of the output stage, which is transistor switch. It will keep output low for some time and then again rise high. It not just sense 36 KHz but also determine if it continuous 36 KHz signal, or a burst of square waves. It rejects continuous 36 KHz like an ambient light.

These diagrams explain TSOP behavior:
TSOP reaction on single burst containing 30 pulses. Burst repetition time (T) is 10 ms. Square wave frequency is 36 KHz.



Optical Test Signal
(IR diode TSAL6200, I$_F$ = 0.4 A, 30 pulses, f = f$_0$, T = 10 ms)
E$_e$
t
t$_{pi}$ *
T
* t$_{pi}$ ≥ 10/fo is recommended for optimal function
Output Signal
16110
V$_O$
V$_{OH}$
V$_{OL}$
1) 7/f$_0$ < t$_d$ < 15/f$_0$
2) t$_{po}$ = t$_{pi}$ ± 6/f$_0$
t
t$_d$1)
t$_{po}$2

Fig. (6). TSOP reaction on single burst with 36 KHz modulated light intensity

Reaction on several bursts one after another

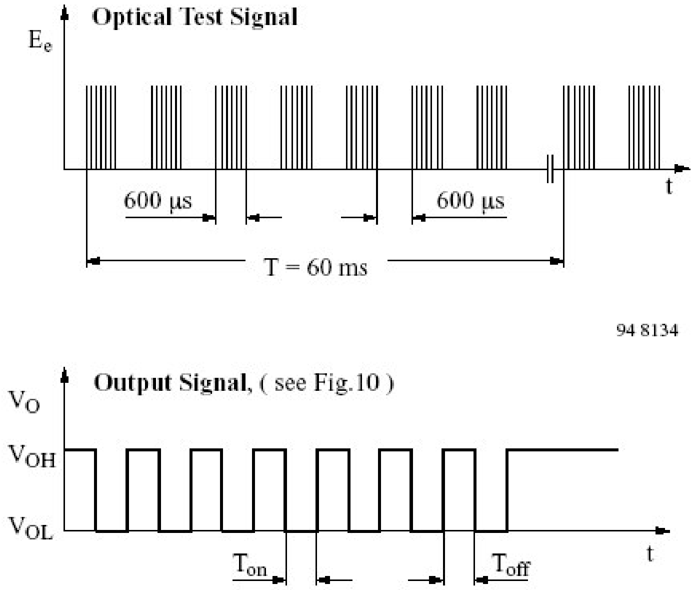
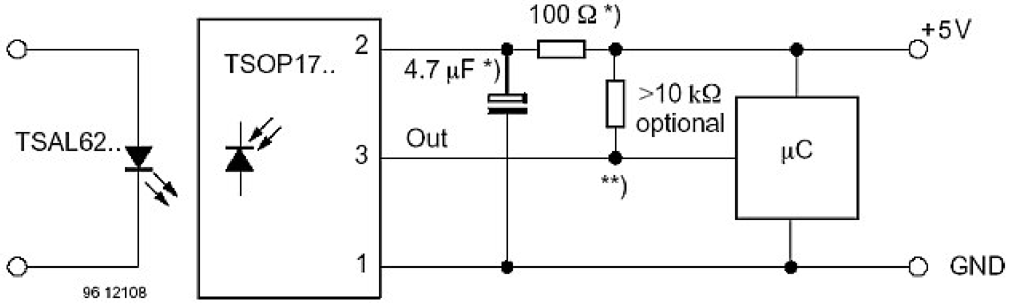


Fig. (7). TSOP reaction on repeating bursts 36 KHz modulated.

Application circuit is simple enough. The rule of thumb is not to skip resistors and capacitor, because in noisy digital environment false readings are possible. 100-Ohm resistor and 4.7-mkF capacitor form low-pass filter to stop power supply disturbances.

Application Circuit



*) recommended to suppress power supply disturbances

**) The output voltage should not be hold continuously at a voltage below 3.3V by the external circuit.

Fig. (7). Application circuit.

TSOP is used typically with high-intensity IR LED. For example, TSAL 6200, TSAL5100 and TSAL5200 are widely used. The best way to turn it on and off is npn transistor as shown at fig 3, because it can consume high current (up to 400 mA, depending of the current-limiting resistor value) and MCU pin may not to be able to supply it. Current can be easily calculated as $(V_{sourse}-1.95)/R_{curr.lim}$. If IR LED is soldered close to TSOP false detections may occur through diode side radiation. This can be eliminated using opaque LED coating: black paper, black thermal shrink tube, etc. Example is shown at figure 8.

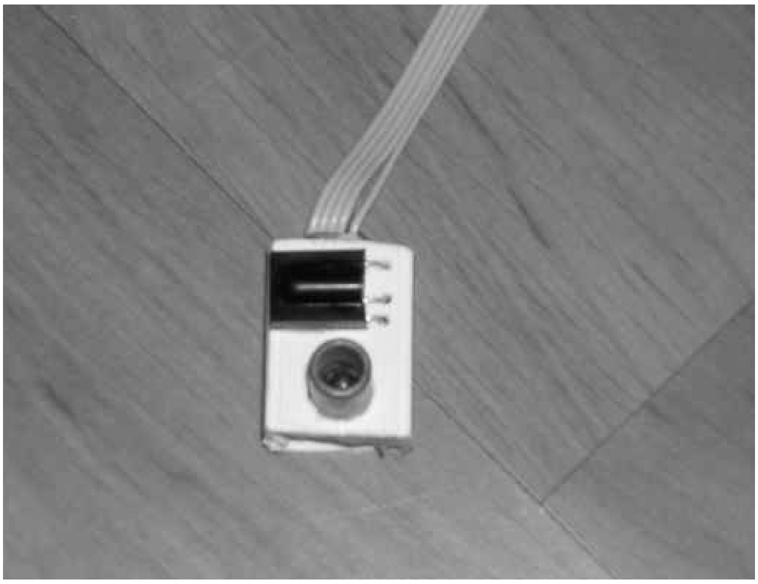


Fig. (8). Assembled bumper.

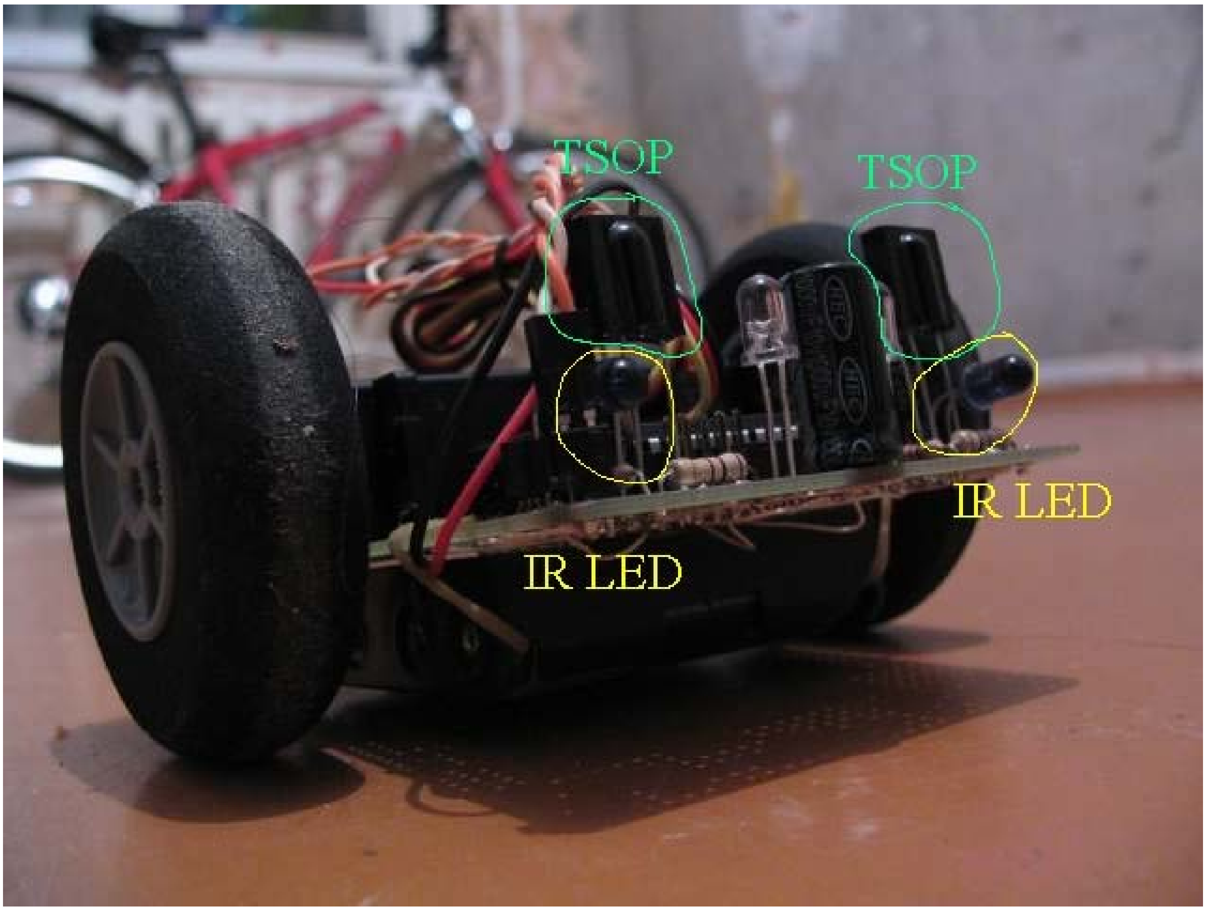


Fig. (9). Simple 2-wheel robot with two IR bumpers in front.

Cylinder in front is a lens, focusing invisible IR radiation to the photodiode. For this IR bumper two MCU pins are required. One pin is used to generate 36 KHz square wave bursts 20-30 pulses each. Another pin is used to monitor output of TSOP. Square wave is generated using timers. Remember, that for 36 KHz square wave, you need to toggle LED twice faster:

```
          |1/72K |
          ______          ______
__________|      |________|      |________
          |<   period     >|
            1/36000 sec.
```

Here is a little example, how to interface TSOP-based sensor. Program is the very "dumb" – its goal is just bring understanding of sensor behavior rather than show software tricks.

The schematic used is shown on figure 10.

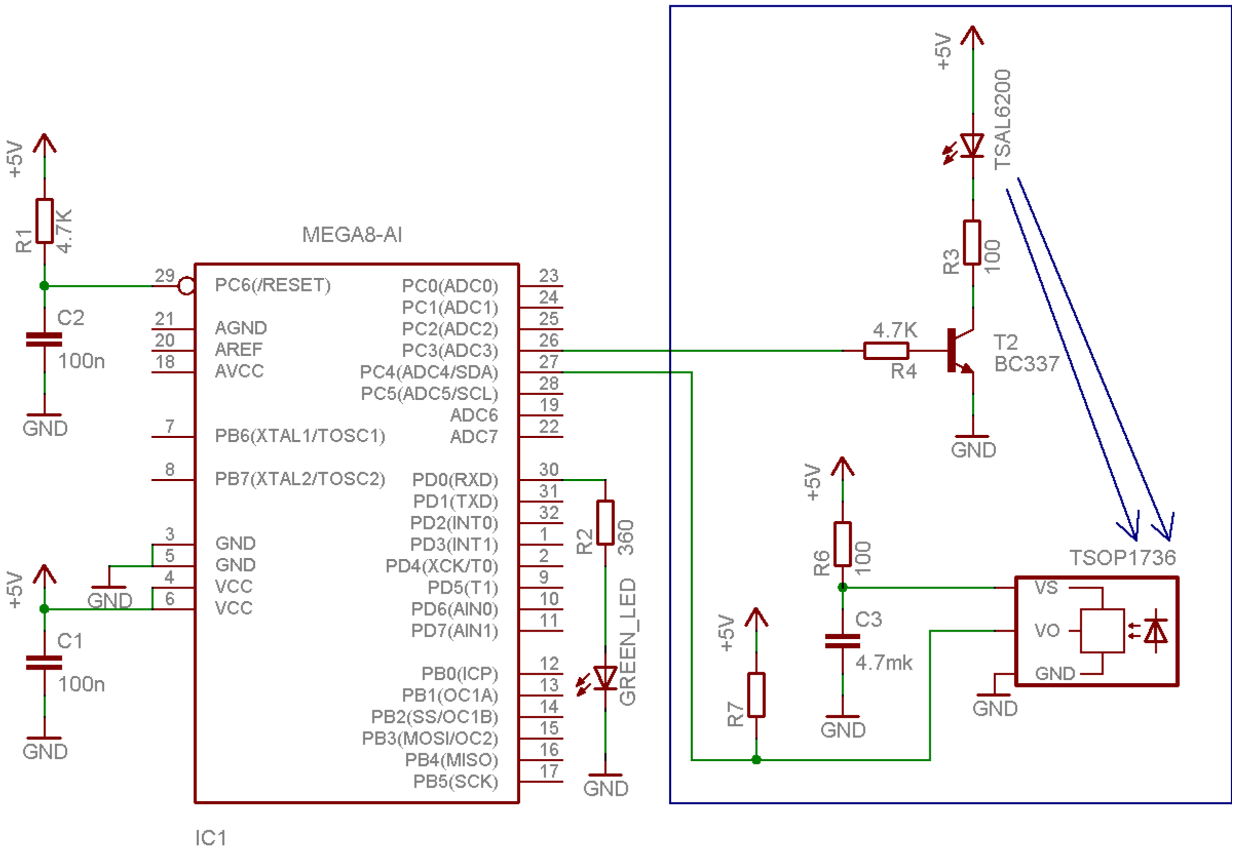


Fig. (10). Schematic that is used in program example.

WINAVR, Atmega 8, clock 8 MHz

```
// One IR bumper is used. IR LED is connected to PORTC3,
// output of TSOP is connected to PORTC4.
// visible wavelength LED is connected to PORTD0 to indicate
// obstacle detection


// global variables declaration
volatile static unsigned char flag=0b00000000 ; //semaphore variable
volatile  unsigned char  pulses=0;                    // variable to count pulses


// initial setup
```

```
DDRC=0b00001000; // configure outputs
DDRD=0b00000001;

// timer 2 is set up to generate 36 KHz square wave, output compare mode.
TCCR2=0b00001010;
OCR2=139;

TIMSK|= (1<<(OCIE2));  // timer 2 output compare interrupt enable
sei();                            // global interrupt enable


// timer 2 interrupt routine
ISR(TIMER2_COMP_vect)
 {PORTC=PORTC^(flag); // if bit 3 of “flag” is set, PORTC3
                                                    // will toggle. Otherwise, it will not.
 pulses++;  //increment variable to count sent pulses
  };


// main program

 int main (void) {
 while(1){

// first, four “if” sentences are used to realize bursts 25 pulses each with time
// between bursts, equal to length of 60 pulses on PORTC2 and PORTC3.
 if ((pulses>=25)&&(bit_is_set(flag,3))) {
        pulses=0;  //
        flag=flag&0b11110111;} // start count from beginning
if ((pulses>=60)&&(bit_is_clear(flag,3))) {
        pulses=0;
        flag=flag|0b00001000;};  // start count from beginning


// Visual obstacle indication. Turn on green LED if TSOP received signal
if (bit_is_clear(PINC,4))  {PORTD=(PORTD|0b00000001);} // TSOP output
low means it receive modulation. Turn on indicator LED.
 else {PORTD=(PORTD&0b11111110);};  // else turn off.
 }; // end of “while” loop
 }; // end of main loop
```

As a conclusion, this simple IR bumper can be used as main navigation sensor together with encoders or even without them in simple 1-MCU robot. However, it also can be used together with SHARP rangefinder or sonar as emergency, or test sensor, or to scan many directions same time. It’s advantage is it’s price-less than 1$ per sensor.

Part of pictures (5-7) was stolen from TSOP17xx datasheet, Vishay Semiconductors, USA.

Figures 8-9 taken from www.roboforum.ru