Sensor Fusion: Ultrasonic Sensor

Microcontroller --> PGA460 --> Murata Ultrasonic Sensor

**Abstract**

Ultrasonic sensors operate on the principle of time-of-flight for the purpose of distance measurement. The time-of-flight principle refers to a measurement of time taken by a particle to travel through a medium. The duration is then used to calculate the distance between the ultrasonic sensor and the barrier by taking d = v\*t/2, for which v is the speed of sound (a constant) and t is the duration the signal takes to travel to the barrier and back to the emitter. The division by 2 in the calculation ensures that the result is the distance between the sensor and barrier, and not the total distance traveled by the signal.

**Materials**

Microcontroller: Arduino Mega

Ultrasonic Driver/ Transducer: PGA460

Ultrasonic sensor: Murata ma58mf14-7n

Step up transformer and signal amplifier circuitry integrated into the system

**Vision:**

Establish communication between microcontroller, transducer, and ultrasonic sensor. Eventual multiplexing between one transducer and multiple ultrasonic sensor.

**Procedure**

In order to interface with the ultrasonic sensor and process returning signals, an embedded system of a microcontroller, transducer, and ultrasonic senor is needed. Due to specific operating settings (voltage, current, etc) a step-up transformer is needed between PGA460 and ultrasonic sensor. Listed below are summary of specifications of PGA460 and Murata ultrasonic sensor.

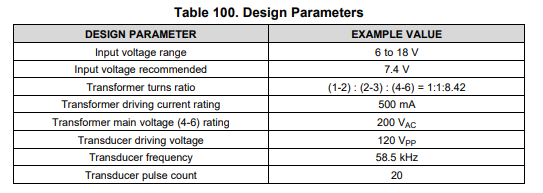
PGA460

Vin; 6-28V, optimum 7.4 V

Baud rate: up to 119200

Serial communication: SERIAL\_8N2

Distance range: Preset 1 (<1 meter) and Preset 2(>1 meter)



Murata ma58mf14-7n

Vin: 80-120Vpp (<20 pulses per second)

Center Frequency: 58 kHz

**Plans going forward:**

First stage:

* + Initializing settings for PGA460.
  + Establishing communication between PGA460 and microcontroller

Second stage:

* + Interfacing PGA460 with sensor
  + Sensor to pulse signals given by PGA460
  + Sensor to detect reflected signal by barrier
  + PGA460 to process reflected signal to derive distance
  + Communicating result back to microcontroller

Third Stage:

* + Driving multiple sensors with one transducer (Multiplexing)
  + Transmitting distance results back to microcontroller
  + Embedded system to run at rate more than 4Hz

Initialization of PGA460

* Register configuration settings for threshold values
* Upon power boot or power cycle, commit configuration settings

Hardware Wiring

