How to use Sanyo 2Y0A21

The Sanyo 2Y0A21 is a low-cost infrared based distance sensor. It has an effective range of 10 to 80cm. The output signal voltage ranges from 3.1v to 0.3v

What you need:

- Sanyo 2Y0A21 IR Distance Sensor
- 1x 1M Ohm resistor
- 1x 2M Ohm resistor
- platform to connect to (ie. breadboard)

Wiring up to BeagleBone Black

The Beagle Bone Black board has seven ADC pins that could be used to interface with the distance sensor, but it requires the signal voltage to be limited to 1.8v or below. Before it can be connected, we must first construct a voltage divider.

A voltage divider is a signal voltage pull down hardware setup that allows higher voltage sources to be interfaced with another voltage source. To get the desired output voltage, you need calculate the desired voltage using the equation:

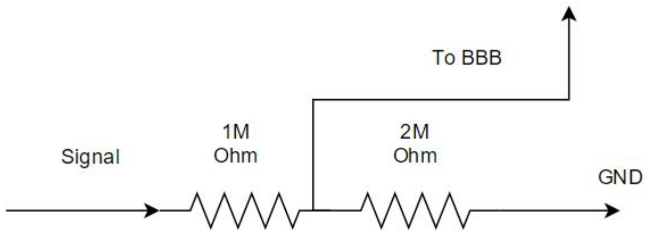
$$V_{out} = V_{in} \cdot \frac{R_2}{R_1 + R_2}$$

Image sourced from Sparkfun

Your choice of resistance to generate the desired ratio is important, the high the resistance of the resistor pair, the less undesired voltage drop will occur under load.

For this instruction, we will be using 1M and 2M Ohm resistor. With R2 being 2M Ohm and R1 as 1M Ohm, the output would be ~2.067v, but after factoring the voltage drop under load, it should be around 1.786v.

To wire the circuit:



To connect the new votlage to BBB, you need to first select a pin to connect to, the options available are AIN0, AIN2, AIN6, AIN4, AIN5, AIN3, AIN1.

P9

DGND	1	2	DGND
VDD_3V3	3	4	VDD_3V3
VDD_5V	5	6	VDD_5V
SYS_5V	7	8	SYS_5V
PWR_BUT	9	10	SYS_RESETN
GPIO_30	11	12	GPIO_60
GPIO_31	13	14	GPIO_40
GPIO_48	15	16	GPIO_51
GPIO_4	17	18	GPIO_5
I2C2_SCL	19	20	I2C2_SDA
GPIO_3	21	22	GPIO_2
GPIO_49	23	24	GPIO_15
GPIO_117	25	26	GPIO_14
GPIO_125	27	28	GPIO_123
GPIO_121	29	30	GPIO_122
GPIO_120	31	32	VDD_ADC
AIN4	33	34	GNDA_ADC
AIN6	35	36	AIN5
AIN2	37	38	AIN3
AIN0	39	40	AIN1
GPIO_20	41	42	GPIO_7
DGND	43	44	DGND
DGND	45	46	DGND

P8

DGND	1	2	DGND
GPIO_38	3	4	GPIO_39
GPIO_34	5	6	GPIO_35
GPIO_66	7	8	GPIO_67
GPIO_69	9	10	GPIO_68
GPIO_45	11	12	GPIO_44
GPIO_23	13	14	GPIO_26
GPIO_47	15	16	GPIO_46
GPIO_27	17	18	GPIO_65
GPIO_22	19	20	GPIO_63
GPIO_62	21	22	GPIO_37
GPIO_36	23	24	GPIO_33
GPIO_32	25	26	GPIO_61
GPIO_86	27	28	GPIO_88
GPIO_87	29	30	GPIO_89
GPIO_10	31	32	GPIO_11
GPIO_9	33	34	GPIO_81
GPIO_8	35	36	GPIO_80
GPIO_78	37	38	GPIO_79
GPIO_76	39	40	GPIO_77
GPIO_74	41	42	GPIO_75
GPIO_72	43	44	GPIO_73
GPIO_70	45	46	GPIO_71

Image from DigiKey

If you are using the ZenCape, please note that AIN0 is already occupied by the potentiometer.

Configuring Beagle Bone Black

To setup the Beagle Bone Black, you configure the pins for ADC. For our example, we are using AIN0.

SSH into BBB:

To Enable the ADC pins on the BBB:

#echo BB_ADC > /sys/devices/bone_capemgr.9/slots

To read the voltage:

#cat /sys/bus/iio/devices/iio:device0/in_voltage0_raw

Parsing the input

To parse the input we need to translate the signal characteristic:

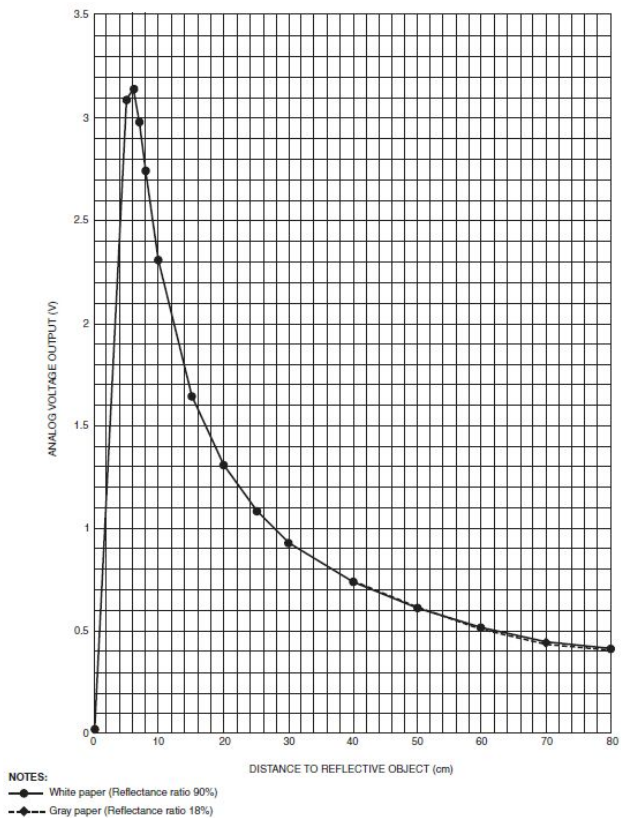


Image from Sanyo Datasheet

The curve fit formula for the sensor (sourced from elinux.org page) :

cm = 41.543 * (Voltage + 0.30221) ^ -1.5281

With the input voltage being 1.786v with a resolution of 4095 (BBB's ADC resolution), voltage becomes:

voltage = 1.786 * (adc_read / 4095) / (3.1 / 1.786)

The combined final formula becomes:

cm = 41.543 * ((1.786^2 * (adc_read / 4095)) + 0.30221) ^ -1.5281

Sources/References:

Sanyo GP2Y0A21TYK Datasheet:
http://www.sharpsma.com/webfm_send/1208

Formula for IR Sensor:
http://elinux.org/ECE597_Project_Robot_Control

BBB Pinout diagram:
http://www.digikey.com/en/articles/techzone/2013/sep/~/media/Images/Article%20Library/TechZone%20Articles/2013/September/BeagleBone%20Black%20Brings%20Arduino-Style%20Connectivity%20Simplicity%20to%20Embedded%20Linux/article-2013september-beaglebone-black-brings-arduino-fig3.jpg