**SAMPLE CODE**

#include <Wire.h>

#include <SparkFun\_APDS9960.h>

#include <LiquidCrystal.h>

LiquidCrystal lcd(13, 12,6, 5, 4, 3);// Pins used for RS,E,D4,D5,D6,D7

#define APDS9960\_INT 2 // Needs to be an interrupt pin

SparkFun\_APDS9960 apds = SparkFun\_APDS9960();

char\* myMenu[]={"BULB1","BULB2","BULB3"," TV"};

int isr\_flag = 0;

int a=0,b=0,c=0,d=0,e=0,f=0;

int i=2,j=0,k=0,pos=1;

int swtch =7;

byte left[8] ={ 0b10000,

0b11000,

0b11100,

0b11110,

0b11110,

0b11100,

0b11000,

0b10000};

byte right[8]={ 0b00001,

0b00011,

0b00111,

0b01111,

0b01111,

0b00111,

0b00011,

0b00001};

void setup() {

pinMode(A0,OUTPUT);

pinMode(A1,OUTPUT);

pinMode(A2,OUTPUT);

pinMode(A3,OUTPUT);

pinMode(swtch,INPUT);

digitalWrite(A0,LOW);

digitalWrite(A1,LOW);

digitalWrite(A2,LOW);

digitalWrite(A3,LOW);

// Set interrupt pin as input

pinMode(APDS9960\_INT, INPUT);

lcd.begin(16,2);

lcd.setCursor(0,0);

lcd.print("Engineers Garage");

lcd.setCursor(0,1);

lcd.print(" APDS-9960 ");

delay(1000);

lcd.setCursor(0,1);

lcd.print(" GestureTest ");

delay(1000);

lcd.clear();

// Initialize Serial port

Serial.begin(9600);

// Initialize interrupt service routine

attachInterrupt(0, interruptRoutine, FALLING);

// Initialize APDS-9960 (configure I2C and initial values)

if ( apds.init() ) {

Serial.println(F("APDS-9960 initialization complete"));

} else {

Serial.println(F("Something went wrong during APDS-9960 init!"));

}

// Start running the APDS-9960 gesture sensor engine

if ( apds.enableGestureSensor(true) ) {

Serial.println(F("Gesture sensor is now running"));

} else {

Serial.println(F("Something went wrong during gesture sensor init!"));

}

}

void loop() {

while(pos==1){

lcd.setCursor(0,0);

lcd.print("Engineers Garage");

lcd.setCursor(0,1);

lcd.print(" Gesture HA ");

if(digitalRead(swtch)==HIGH){

lcd.clear();

delay(500);

pos=2;

break;

}

}

lcd.setCursor(0,0);

lcd.print("Engineers Garage");

if( isr\_flag == 1 ) {

detachInterrupt(0);

handleGesture();

isr\_flag = 0;

attachInterrupt(0, interruptRoutine, FALLING);

controlAppl();

}

i=2;

k=0;

lcd.setCursor(6,1);

lcd.print(myMenu[j]);

if(digitalRead(swtch)==LOW){

pos=1;

}

if(j<3){

lcd.createChar(2,left);

lcd.setCursor(11,1);

lcd.write(2);

}

if(j>0){

lcd.createChar(1,right);

lcd.setCursor(5,1);

lcd.write(1);

}

if(j==0){

lcd.setCursor(5,1);

lcd.print(" ");

}

if(j>=3){

lcd.setCursor(11,1);

lcd.print(" ");

}

}

void interruptRoutine() {

isr\_flag = 1;

}

void controlAppl() {

if(j==0 && i==1){

if(a==0){

digitalWrite(A0,HIGH);

a=1;}

}

if(j==0 && i==0){

if(a==1){

digitalWrite(A0,LOW);

a=0;}

}

if(j==1 && i==1){

if(b==0){

digitalWrite(A1,HIGH);

b=1;}

}

if(j==1 && i==0){

if(b==1){

digitalWrite(A1,LOW);

b=0;}

}

if(j==2 && i==1){

if(c==0){

digitalWrite(A2,HIGH);

c=1;}

}

if(j==2 && i==0){

if(c==1){

digitalWrite(A2,LOW);

c=0;}

}

if(j==3 && i==1){

if(e==0){

digitalWrite(A3,HIGH);

e=1;}

}

if(j==3 && i==0){

if(e==1){

digitalWrite(A3,LOW);

e=0;}

}

if((k==1 || k==3) && f==0){

digitalWrite(A0,HIGH);

digitalWrite(A1,HIGH);

digitalWrite(A2,HIGH);

digitalWrite(A3,HIGH);

}

if((k==2 || k==3) && f==0){

digitalWrite(A0,LOW);

digitalWrite(A1,LOW);

digitalWrite(A2,LOW);

digitalWrite(A3,LOW);

}

}

void handleGesture() {

lcd.setCursor(6,1);

lcd.print(" ");

if ( apds.isGestureAvailable() ) {

switch ( apds.readGesture() ) {

case DIR\_UP:

Serial.println("UP");

i=0;

break;

case DIR\_DOWN:

Serial.println("DOWN");

i=1;

break;

case DIR\_LEFT:

Serial.println("LEFT");

if(j>0){

j--;

delay(200);

}

break;

case DIR\_RIGHT:

Serial.println("RIGHT");

if(j<3){

j++;

delay(200);

}

break;

case DIR\_NEAR:

k=1;

Serial.println("NEAR");

break;

case DIR\_FAR:

k=2;

Serial.println("FAR");

break;

default:

Serial.println("NONE");

k=3;

}

}

}