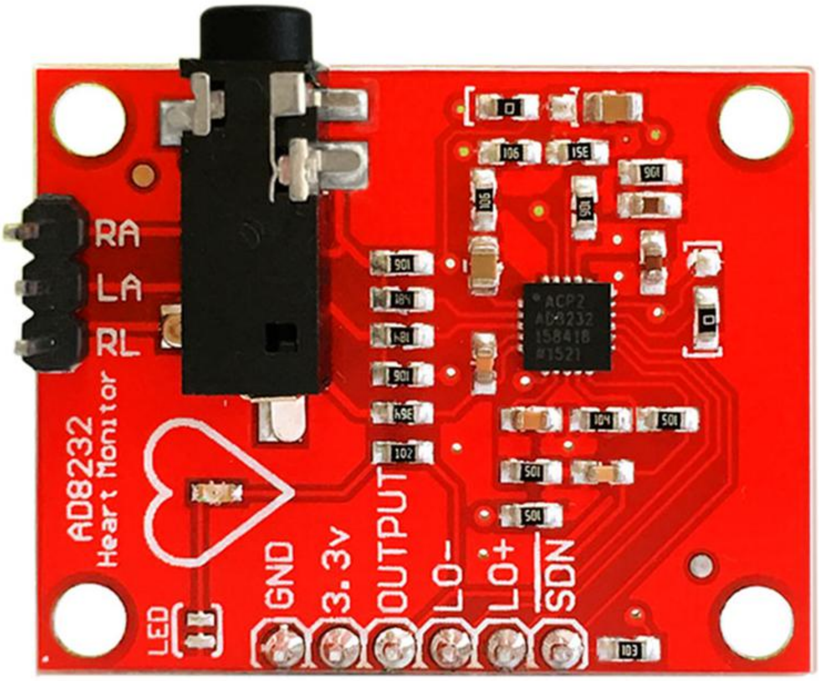
MD0339 Keyes AD8232 Electrocardiogram Monitoring Sensor Module



1. Introduction

AD8232 is an integrated front end, suitable for monitoring heart rate through adjusting bio-electricity signal of heart. It aims at monitoring different vital signals and is a analog front end of heart rate monitor, featured low power consumption and single lead.

2. Technical Parameters

Supply Voltage: DC 3.3V

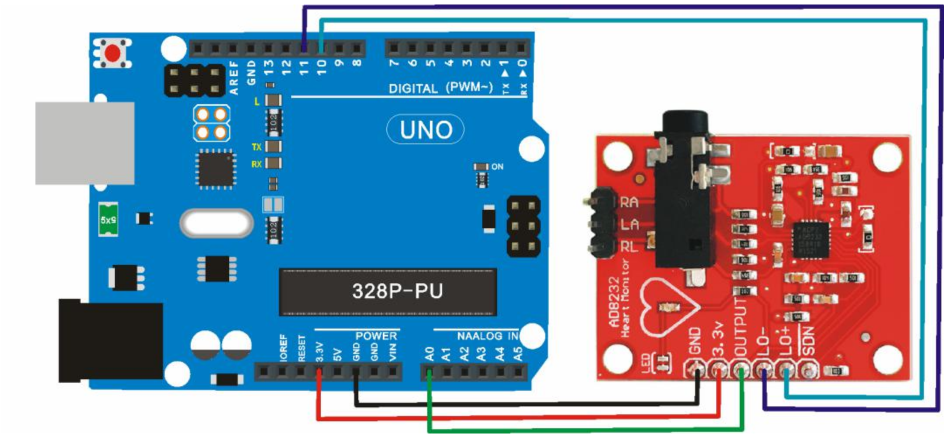
Output: Analog Output

Interface(connecting RA LA RL): 3PIN 2.54 pin header or earphone socket

Size: 36mm * 31mm * 18mm

Working Temperature: -40 ℃ — +85 ℃

3. Connection Diagram



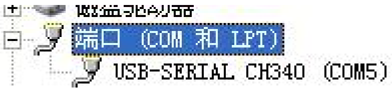
4. Sample Code

1>First upload program to UNO board with Arduino IDE

```
 void setup() {
   // initialize the serial communication:
   Serial.begin(9600);
   pinMode(10, INPUT); // Setup for leads off detection LO +
   pinMode(11, INPUT); // Setup for leads off detection LO -

 }

 void loop() {

   if((digitalRead(10) == 1)||(digitalRead(11) == 1)){
     Serial.println('!');
   }
   else{
     // send the value of analog input 0:
       Serial.println(analogRead(A0));
   }
   //Wait for a bit to keep serial data from saturating
   delay(1);
 }
```

2>Use processing.exe software to download program.myPort = new Serial(this, Serial.list()[2], 9600);

Serial.list()[2],9600,of this program here, check serial port of current computer



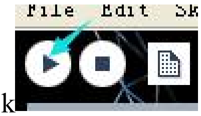
For example, there only one port showed like this; when there no second port,Serial.list()[0],9600. When there are COM1 and COM5 on computer,COM5 is of UNO of testing AD8232 board and Serial.list()[1],9600. So reasoning like this in turns.

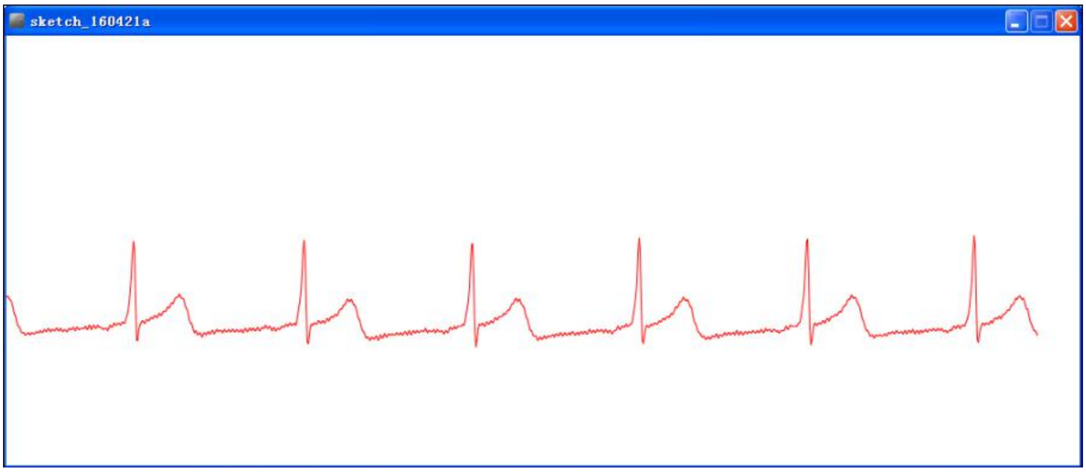
```
 /******************************************************************************
 Heart_Rate_Display.ino
 Demo Program for AD8232 Heart Rate sensor.
 Casey Kuhns @ SparkFun Electronics
 6/27/2014
 https://github.com/sparkfun/AD8232_Heart_Rate_Monitor

 The AD8232 Heart Rate sensor is a low cost EKG/ECG sensor.   This example shows
 how to create an ECG with real time display.   The display is using Processing.
 This sketch is based heavily on the Graphing Tutorial provided in the Arduino
```

```
IDE. http://www.arduino.cc/en/Tutorial/Graph

Resources:
This program requires a Processing sketch to view the data in real time.

Development environment specifics:
   IDE: Arduino 1.0.5
   Hardware Platform: Arduino Pro 3.3V/8MHz
   AD8232 Heart Monitor Version: 1.0

This code is beerware. If you see me (or any other SparkFun employee) at the
local pub, and you've found our code helpful, please buy us a round!

Distributed as-is; no warranty is given.
******************************************************************************/

import processing.serial.*;

Serial myPort;               // The serial port
int xPos = 1;                // horizontal position of the graph
float height_old = 0;
float height_new = 0;
float inByte = 0;


void setup () {
   // set the window size:
   size(1000, 400);

   // List all the available serial ports
   println(Serial.list());
   // Open whatever port is the one you're using.
   myPort = new Serial(this, Serial.list()[2], 9600);
   // don't generate a serialEvent() unless you get a newline character:
   myPort.bufferUntil('\n');
   // set inital background:
   background(0xff);
}


void draw () {
   // everything happens in the serialEvent()
}
```

```
void serialEvent (Serial myPort) {
  // get the ASCII string:
  String inString = myPort.readStringUntil('\n');

  if (inString != null) {
    // trim off any whitespace:
    inString = trim(inString);

    // If leads off detection is true notify with blue line
    if (inString.equals("!")) {
      stroke(0, 0, 0xff); //Set stroke to blue ( R, G, B)
      inByte = 512;  // middle of the ADC range (Flat Line)
    }
    // If the data is good let it through
    else {
      stroke(0xff, 0, 0); //Set stroke to red ( R, G, B)
      inByte = float(inString);
     }

     //Map and draw the line for new data point
     inByte = map(inByte, 0, 1023, 0, height);
     height_new = height - inByte;
     line(xPos - 1, height_old, xPos, height_new);
     height_old = height_new;

      // at the edge of the screen, go back to the beginning:
      if (xPos >= width) {
        xPos = 0;
        background(0xff);
      }
      else {
        // increment the horizontal position:
        xPos++;
      }
  }
}
```

5. Result

Open above program with processing.exe , and click . Place three sensors on body,that is, stick RA around tummy, LA to left chest, RL to right chest. Then display the following wave pattern and LED on the PCB is blinking with heart beating. So the board is normal.



sketch_160421a