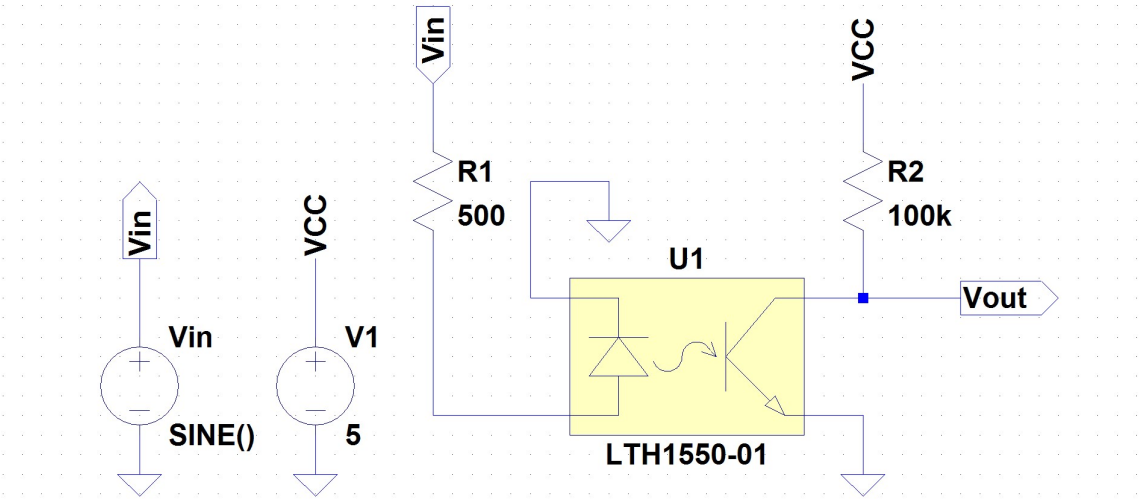
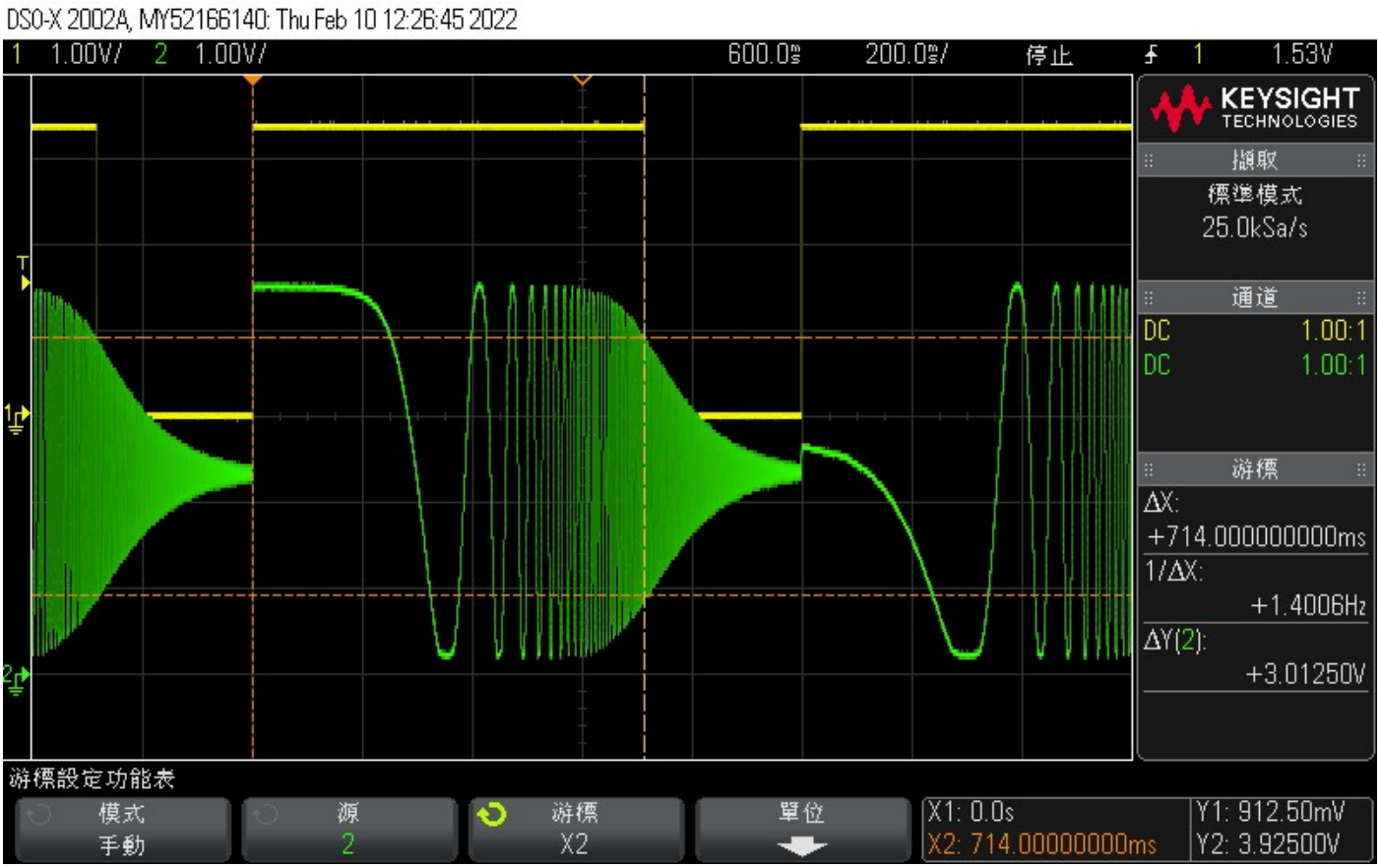
REPORT

Experiment 1: IR Driver and Sensor



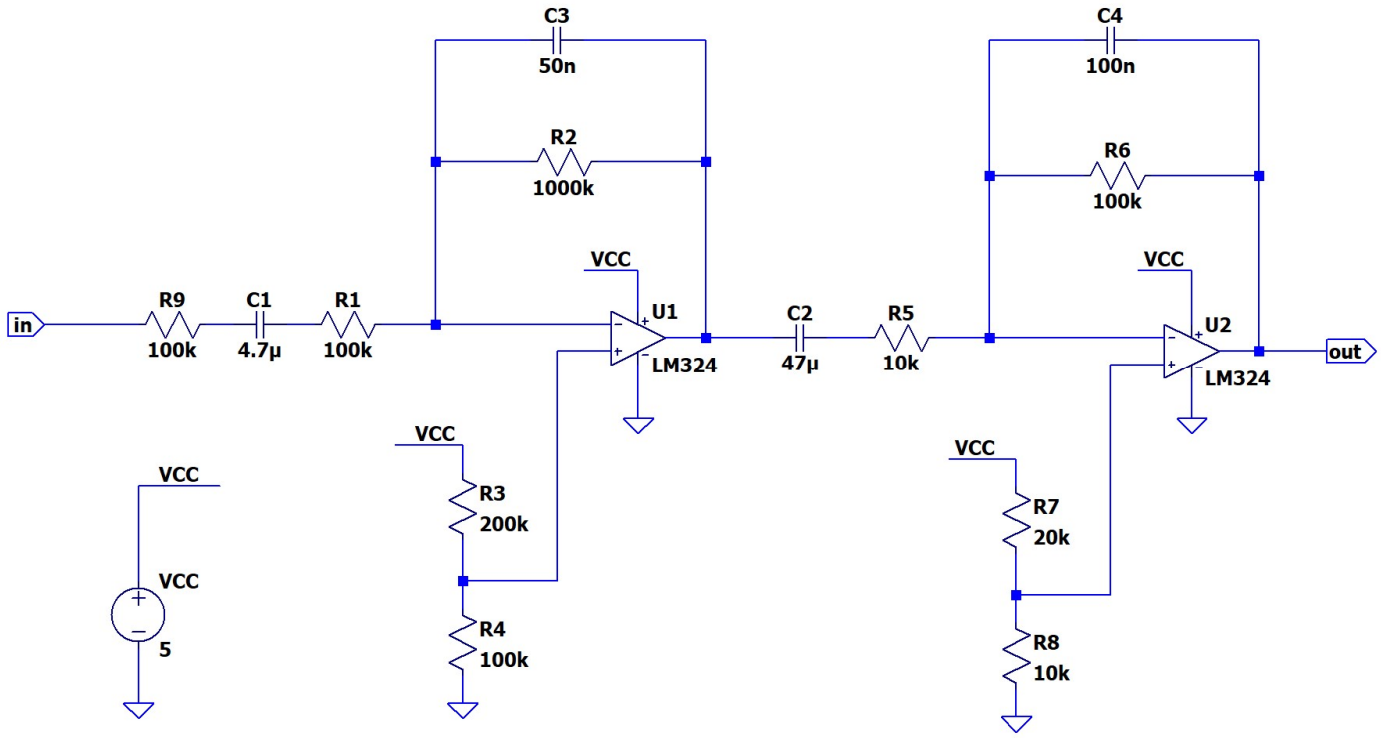
2. AC SWEEP and Bias



$f_{3dB,H}$ (Hz)	Vout average voltage (V)
375	**2.5**

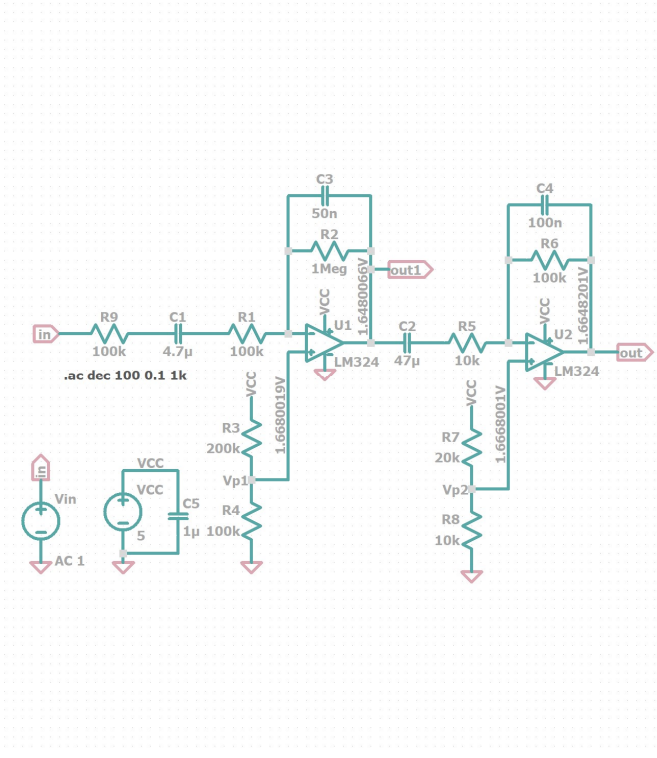

DSO-X 2002A, MY52166140: Thu Feb 10 12:30:48 2022
1 1.00V/ 2 1.00V/
600.0µs 500.0µs/ 停止
1 1.53V
KEYSIGHT TECHNOLOGIES
擷取
標準模式
10.0kSa/s
通道
DC 1.00:1
DC 1.00:1
量測
峰-峰值(1):
3.50V
峰-峰值(2):
4.34V
頻率(2):
103Hz
平均 - 全螢面(2):
2.4710V
量測功能表
源 2
類型: 平均 - 全螢面
新增 量測
設定
清除量測

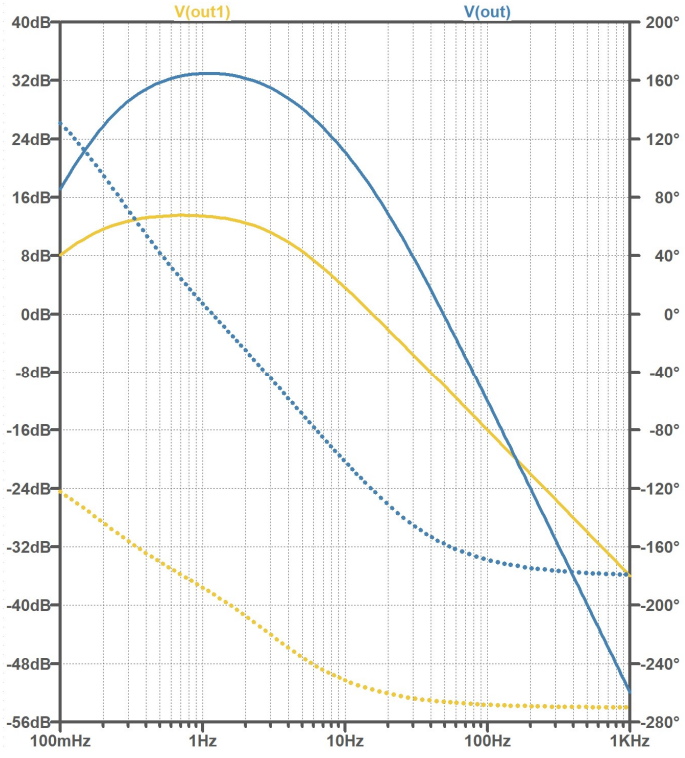
Experiment 2: Filter Stage



Vin : 50mVpp or appropriate value that Vo is not distorted

OSC : DC coupling





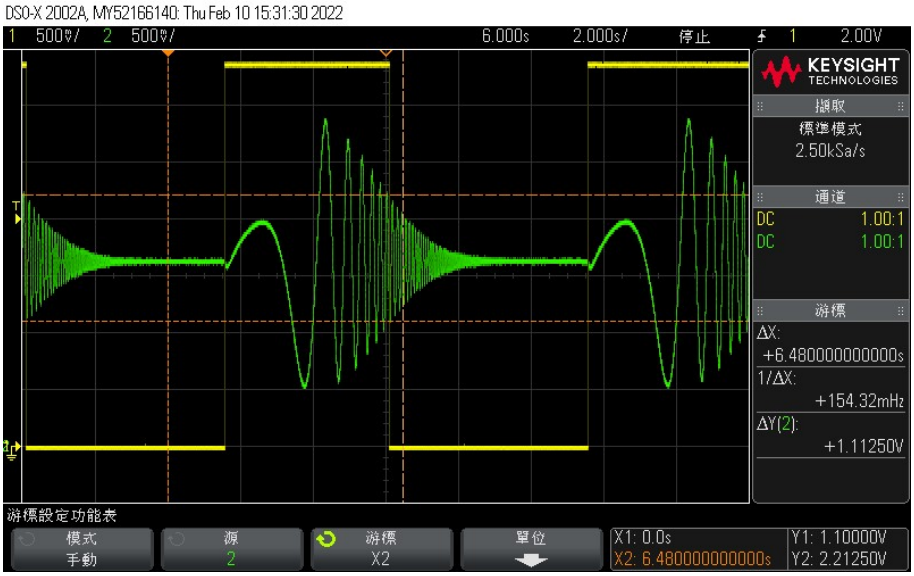
2. DC Bias

U1,V+ (V)	U1,Vout (V)	U2,V+ (V)	U2,Vout (V)
1.59	**1.63**	**1.59**	**1.63**

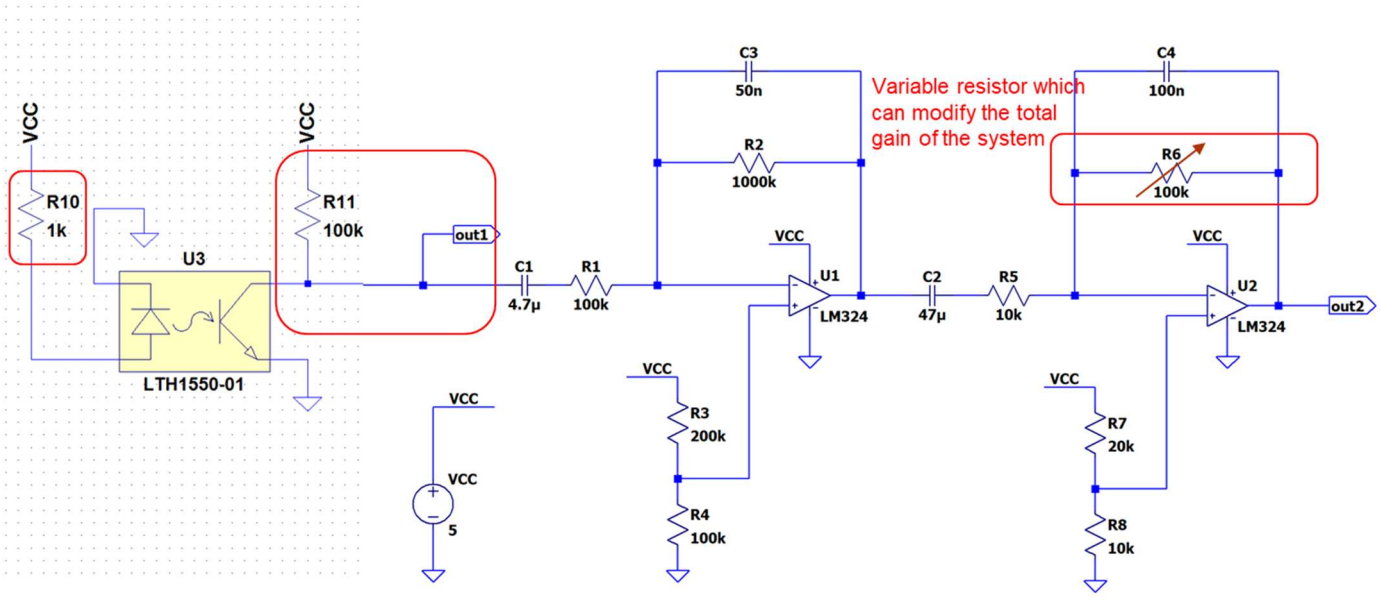
The frequency is too low => measured DC bias might not be correct, use cursor to correct the error

3. AC SWEEP waveform

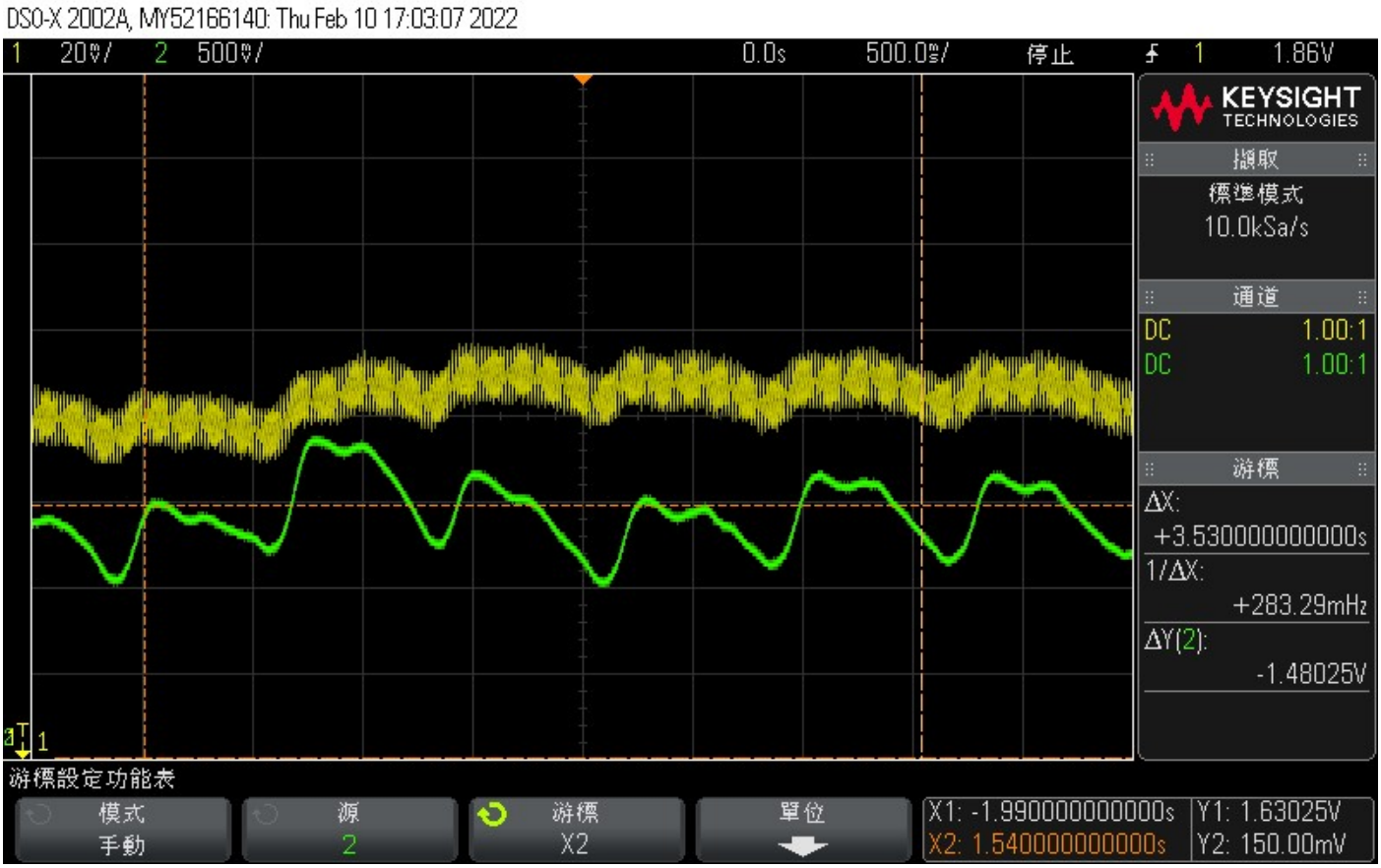
$f_{3dB,H}$ (Hz)____6.5____



Experiment 3: Heart Rate Monitor

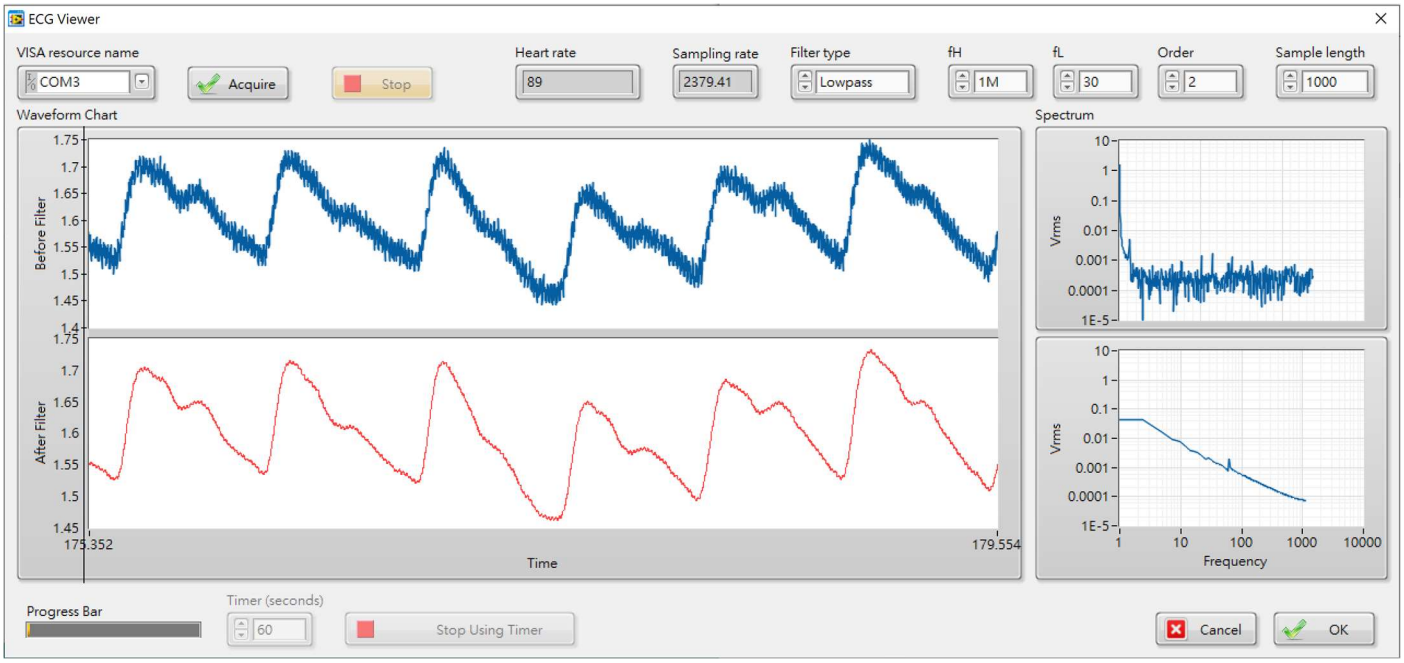


2. Vout1 and Vout2 waveform



3. Record you real time heart rate = 78.741 BPM (beat per minute)

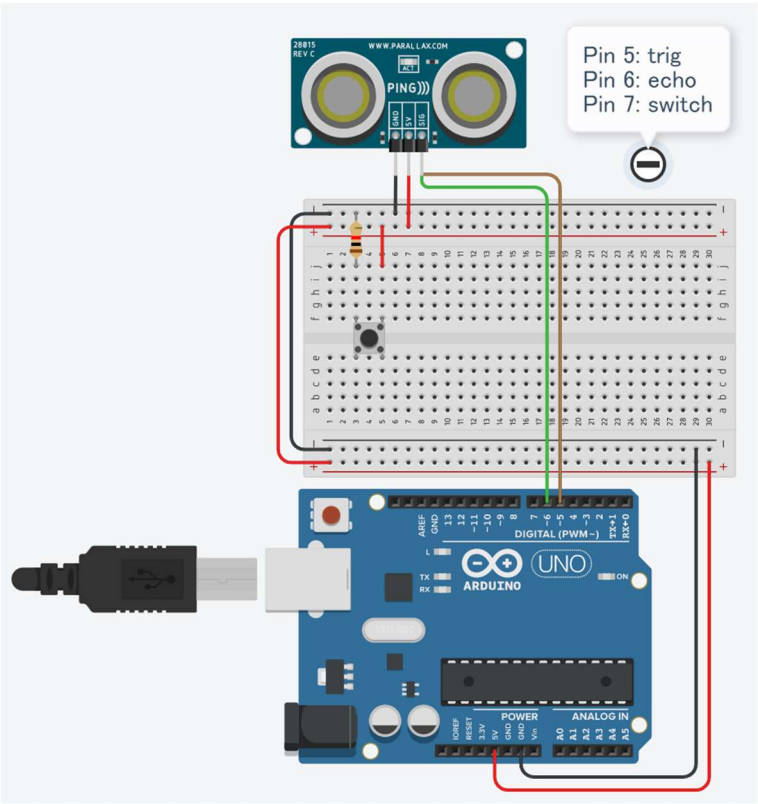
Take a screenshot after reading the stable heart rate.



Experiment 4: Ultrasonic Sensor

#REF code: https://www.block.tw/blog/arduino-hcsr04/

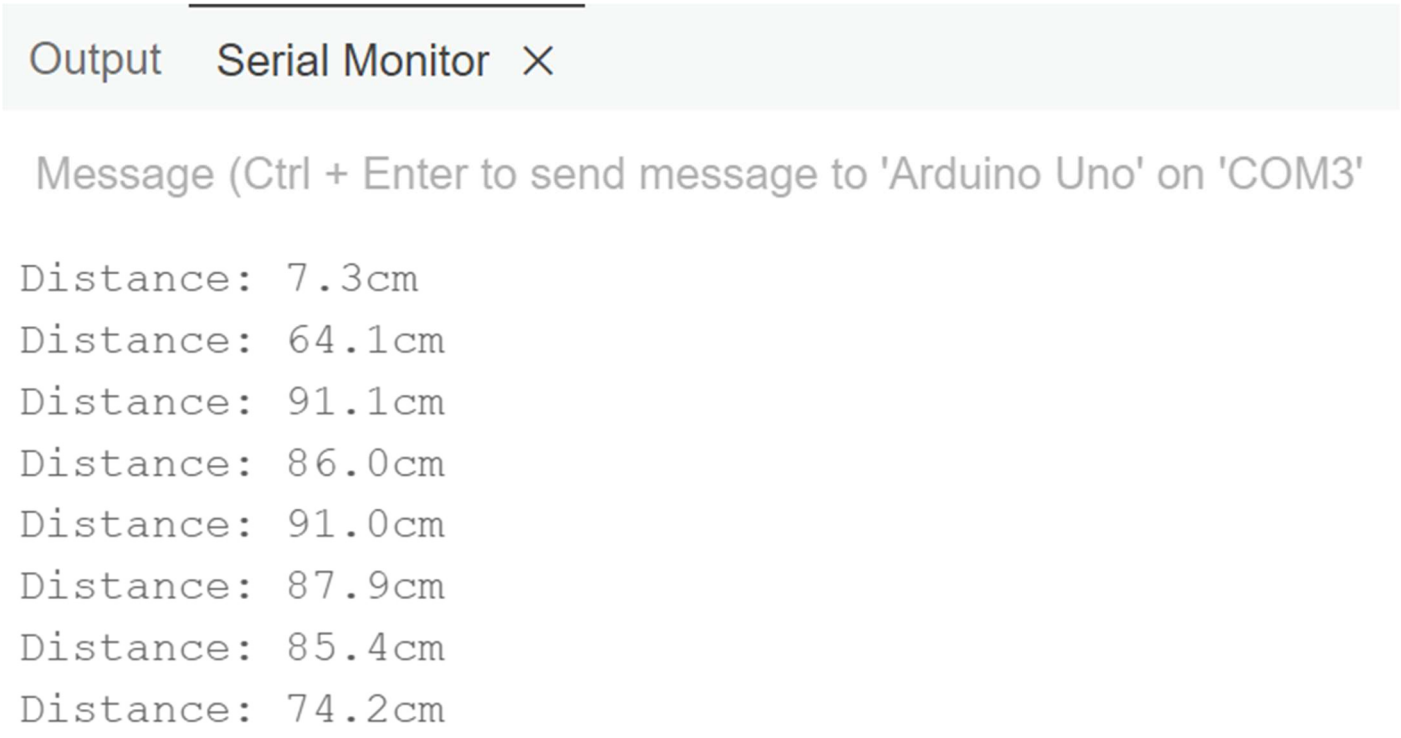
The circuit diagram of your design: (label every port clearly)



The sketch of your design: (copy from the Arduino IDE window and paste here)

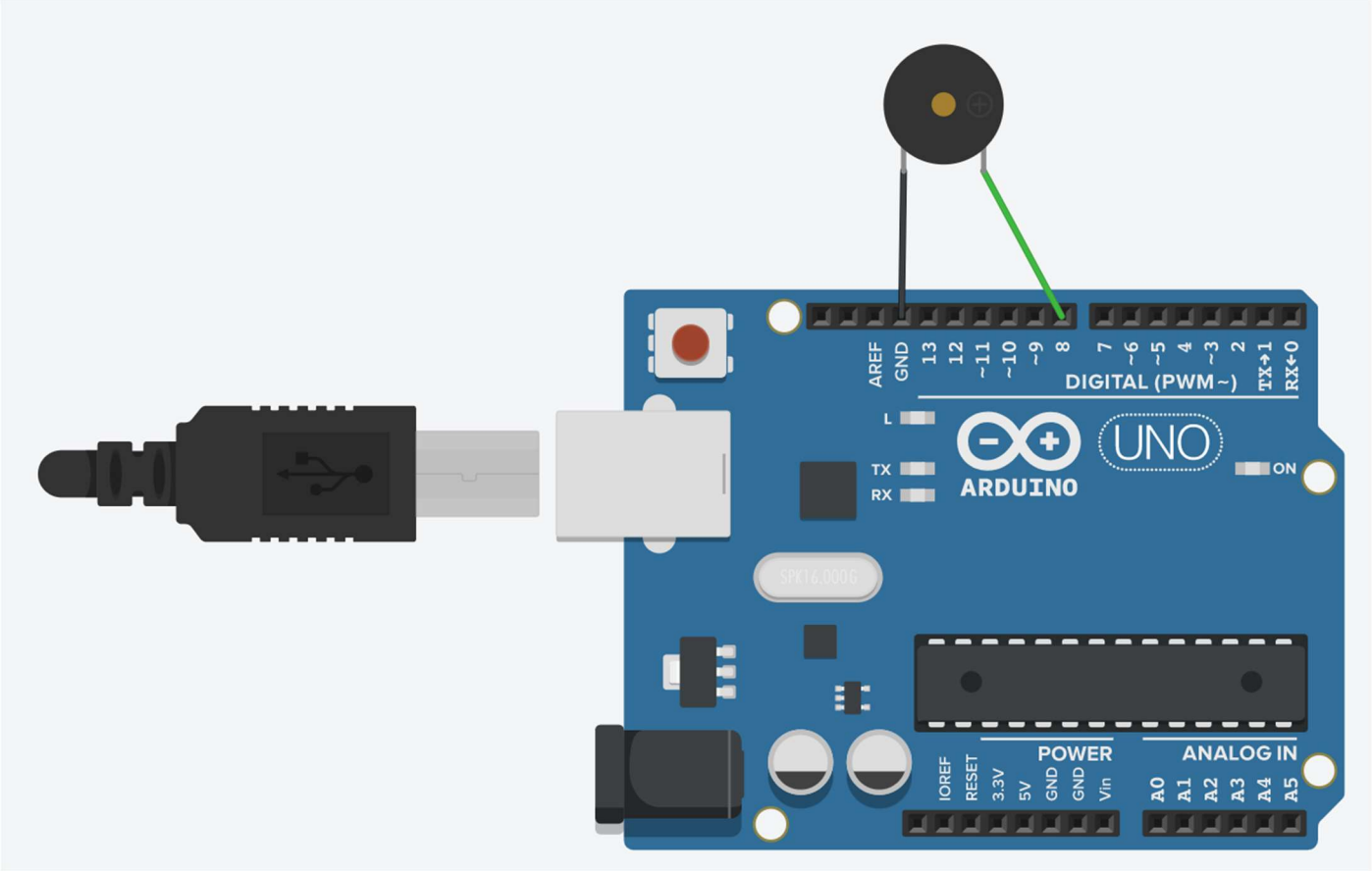
```
//could be improved debounce using rising edge detect+delay
int trig = 5;
int echo = 6;
int btn = 7;
bool sw;
float distance;

void setup()
{
  pinMode(trig, OUTPUT); //Arduino 2 Module
  pinMode(echo, INPUT);  //Module 2 Arduino
  pinMode(btn, INPUT);  //Module 2 Arduino
  Serial.begin(9600);
}

void loop()
{
  sw = digitalRead(btn);
  if(sw)
  {
    GetDistance();
    delay(100);
  }
}

void GetDistance()
{
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  distance = pulseIn(echo, HIGH);
  distance *= 0.017;
  Serial.print("Distance: ");Serial.print(distance,1);Serial.print("cm\n");
}
```

The screen capture of the serial monitor: (show the distance value on the window)



Experiment 5: Melody Generator

The circuit diagram of your design: (label every port clearly)



The sketch of your design: (copy from the Arduino IDE window and paste here)

```
#define NOTE_C4  262
#define NOTE_G3  196
#define NOTE_A3  220
#define NOTE_B3  247
#define NOTE_C4  262
//dont forget to set
#define NOTESNUM 225

#define D1 277.18
#define R1 293.67
#define M1 329.63
#define F1 369.99
#define S1 415.3
#define L1 440
#define C1 493.88
#define D2 554.37
#define R2 587.33
#define M2 659.26
```

```
#define F2 739.99
#define S2 830.61
#define L2 880
#define C2 987.77


// notes in the melody:
float melody[] =
{
  C1,L1,S1,M1,

  F1,0,F1,D2,C1,0,L1,0,
  S1,0,S1,S1,C1,0,L1,S1,
  F1,0,F1,L2,S2,L2,S2,L2,
  F1,0,F1,L2,S2,L2,S2,L2,

  F1,0,F1,D2,C1,0,L1,0,
  S1,0,S1,S1,C1,0,L1,S1,
  F1,0,F1,L2,S2,L2,S2,L2,
  F1,0,F1,L2,S2,L2,S2,L2,

  L1,L1,L1,L1,D2,D2,D2,D2,
  C1,C1,C1,C1,M2,M2,M2,M2,
  F2,F2,F2,F2,F2,F2,F2,F2,
  F2,F2,F2,F2,

  C1,L1,S1,M1,

  F1,0,F1,D2,C1,0,L1,0,
  S1,0,S1,S1,C1,0,L1,S1,
  F1,0,F1,L2,S2,L2,S2,L2,
  F1,0,F1,L2,S2,L2,S2,L2,

  F1,0,F1,D2,C1,0,L1,0,
  S1,0,S1,S1,C1,0,L1,S1,
  F1,0,F1,L2,S2,L2,S2,L2,
  F1,0,F1,L2,S2,L2,S2,L2,

  L1,L1,L1,L1,D2,D2,D2,D2,
  C1,C1,C1,C1,M2,M2,M2,M2,
  F2,F2,F2,F2,F2,F2,F2,F2,
```

```
  F2,F2,F2,F2,F2,F2,F2,F2,

  L1,L1,L1,L1,D2,D2,D2,D2,
  C1,C1,C1,C1,M2,M2,M2,M2,
  F2,F2,F2,F2,F2,F2,F2,F2,
  F2,F2,F2,F2,

  F2

};

// note durations: 2 = quarter note, 1 = eighth note, etc.:
int noteDurations[] = {
  1,1,1,1,

  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,

  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,

  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,

  1,1,1,1,

  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,

  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
```

```
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,

  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,
  1,1,1,1,

  4

};

void setup() {
  delay(3000);
  // iterate over the notes of the melody:
  for (int thisNote = 0; thisNote < NOTESNUM ; thisNote++) {

    // eighth note=8,quarter notes=4
    float noteDuration = 238.0952381;
    tone(8, melody[thisNote], noteDuration);

    float pauseBetweenNotes = noteDuration * noteDurations[thisNote];
    delay(pauseBetweenNotes*0.9);
    noTone(8);
    delay(pauseBetweenNotes*0.1);
    // stop the tone playing:
    noTone(8);
  }
}

void loop() {
  // no need to repeat the melody.
}
```

Your demo video (play the complete melody) link: https://youtu.be/OQ0YUcwRh7Y