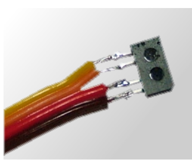
The following information is for the EMANT300

Optical Heart Rate Monitor

Pulse Oximetry is a non invasive method of measuring a person's oxygenation level. It monitors the percentage of haemoglobin (Hb) which is saturated with oxygen. A probe is attached to the patient's finger or ear lobe comprising LEDs and photodetector. As the light from the two LEDs (red - 660 nm and infrared - 950 nm) pass through the body tissues to a photodetector, it is absorbed by blood and soft tissue.

The light absorption rate at the two wavelengths by the hemoglobin is different and depends on the degree of oxygenation. The light level changes as the blood is pumped by the heart. As a consequence, the oximeter also measures the heart rate in beats per minute (BPM).

In this application note, we create a heart rate monitor by using one IR LED & phototransistor pair and observing the waveform at the phototransistor output. This is intended for illustrating a typical light sensor application and not intended for actual medical use. The accompanying video also shows the typical low level signals involved in this type of circuit and how the dynamic range from a 24 bit ADC (EMANT300 USB DAQ) allows such signals to be observed without further signal conditioning.



The TCRT1010 have a compact construction where the emitting-light source and the detector are arranged in the same direction to sense the presence of an object by using the reflective IR-beam from the object. The operating wavelength is 950 nm. The detector is a phototransistor. Together with the EMANT300 USB DAQ, they form a computerbased optical heart monitor

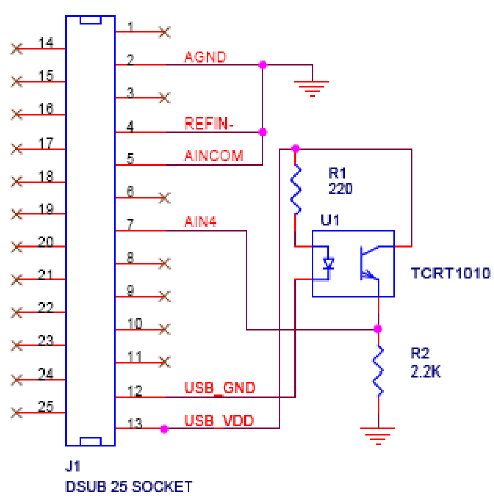


Fig 1 : Optical heart rate monitor schematic - Connections to the EMANT300 USB DAQ

IR emitter and detector encased in velcro

The following C# .NET program uses the EmantHRM component. This component takes the voltage read at the emitter of the phototransistor and converts it to heart rate in BPM (beats per

View Cart

Products

DAQ Training Kit

EMANT300 USB DAQ module, Light Application Adaptor, USB Cable, Instructional Guides

Bluetooth DAQ Starter Kit

EMANT380 Bluetooth DAQ module, Light Application Adaptor, 3.3V regulator, Battery Holder, Instructional Guides

EMANT300 USB DAQ Module

24 bit ADC, 8 bit Current Output, 8DIO, PWM or Counter

EMANT380 Bluetooth DAQ Module

24 bit ADC, 8 bit Current Output, 8DIO, PWM or Counter

Light Application Adaptor

3 LEDs, 1 switch, 1 Light Sensor, 1 Thermistor, Screw Terminals for external circuits

Bridge Sensor Application Adaptor

3.3V excitation, Up to 3 bridge inputs

Pressure Application Adaptor

200 kPa Freescale MPX2202, +/- 0.25% Linearity, Differential Configurations

Prototype Board Application Adaptor

Prototype board with DB25 socket. Soldering required

Breadboard Application Adaptor

Breadboard with DB25 socket. For quick experiments

Screw Terminals Application Adaptor

Screw Terminal with Prototype board. For quick wiring and custom signal conditioning circuits

3.3V Regulator

Regulates input voltage to 3.3V output

Battery Holder

3 AA Battery Holder

Discontinued Products

Strain Application Adaptor

2.5V excitation, Up to 6 inputs, Quarter, half or full bridge configurations

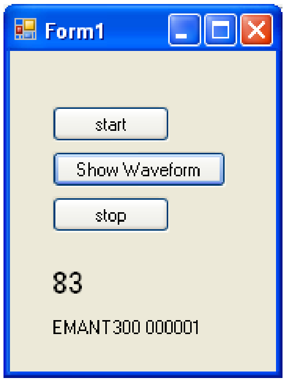
Tutorials

Visual Basic .NET
LabVIEW
Symbian S60 Python

Application Notes

Measure Strain
Strain Gauge
Center of Pressure
Hand Dynamometer
Measure Light
Photodiode
LDR
Heart Rate Monitor
Measure Pressure
Silicon Piezoresistive
Depth and Hydrostatic
Pressure
Measure Temperature
Thermocouple
Thermistor
Multiple Thermistors
Diodes
Measure Frequency
Counter
Digital IO
3A 250Vac Switch
DC Fan Speed Control
Stepper Motor Control
Measure Distance
Ultrasonic sensor
Other Applications
PID Temperature Control
Analog Input > 2.5V
10V Analog Output
Use multiple EMANT300
Text Msg by SMS
Voice Recognition
Skype & EMANT300
Matlab & EMANT300
C# Strain Logger to Excel

minute)

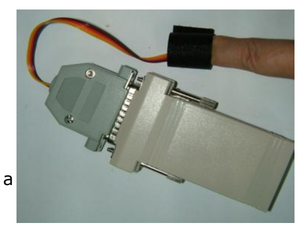



```
private void button1_Click(object sender, EventArgs e)
{
hrm1.Open();
label2.Text = hrm1.HwId;
}

private void button2_Click(object sender, EventArgs e)
{
hrm1.Close();
}

private void hrm1_onValidHR(object sender, EventArgs e)
{
label1.Text = hrm1.Value.ToString();
}

private void button3_Click(object sender, EventArgs e)
{
hrm1.Show();
}
```



When the console program is first run, the user to place his/her finger on the sensor.

When a finger is placed on the phototransistor, the voltage read rises above 0.3V. If a good contact is made, a sinusoidal type signal is observed. As the signal rides on a fluctuating DC signal, a simple differential signal is created. From maths, we know that when a max or min is reached, its differential value is zero. After this an autocorrelation function is applied and the peaks extracted to obtain the heart rate. When the signal is noisy, a wrong BPM will be calculated. Any count that is obviously wrong is ignored. This is all taken care of by the HRM class. The HRM class is written in C#. You can improve on it using the Visual C# 2005 Express.

Clicking on the Show Waveform button shows the filtered waveform seen at the phototransistor. You will achieve a good heart rate if your waveform is as seen below.