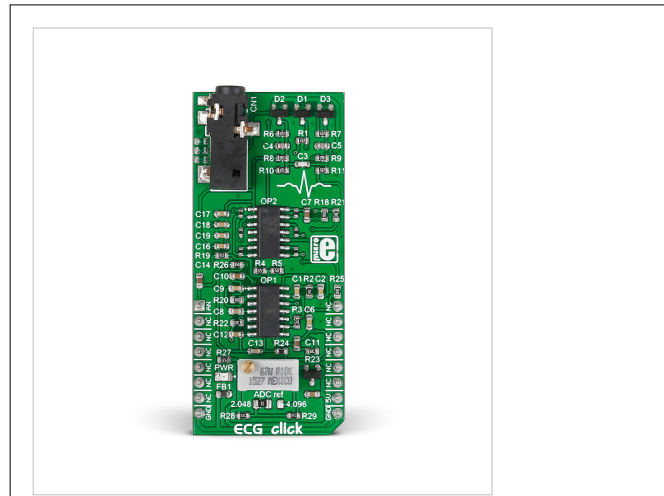


# ECG click

## ECG click



### ECG click

<b>IC/Module</b>	MCP609 – OpAmp <sup>[1]</sup> MAX6106 – Voltage Reference <sup>[2]</sup>
<b>Interface</b>	AN
<b>Power supply</b>	5V
<b>Website</b>	<a href="http://www.mikroe.com/click/ecg">www.mikroe.com/click/ecg</a> <sup>[3]</sup>
<b>Schematic</b>	ECG click schematic <sup>[4]</sup>

ECG click measures the electrical activity of a beating heart through electrodes taped to the skin. It requires little setup, and the final measurement results can be displayed as an Electrocardiogram using a free app.

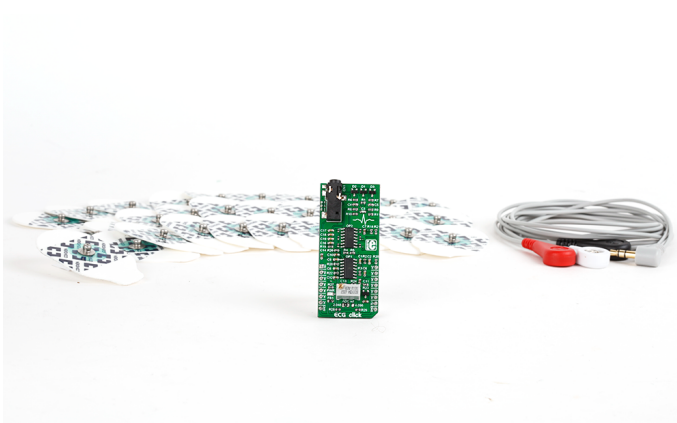
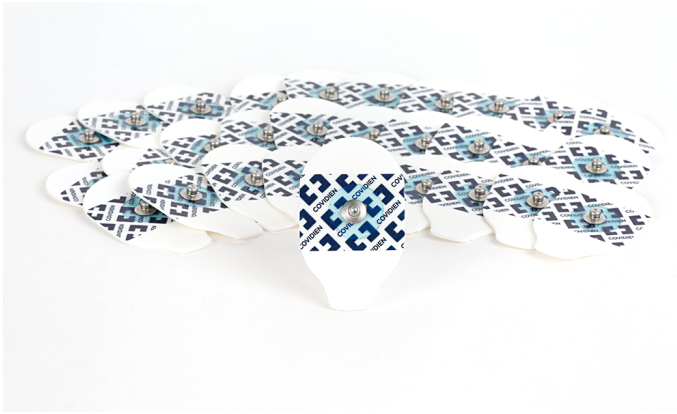
**Note:** ECG click is a prototyping tool, not a medical-grade device. Do not use it to diagnose patients.

## Necessary components

In order to record ECG, you will need the following:

- ECG click
- Cable
- Disposable electrodes

All of the above items are sold in our store. You can buy them separately, or as a ECG kit bundle. The disposable electrodes come in a pack of 30.



## Circuit design and operating principles

There is onboard 3.5mm phone jack that is used to connect cables/electrodes to the click board. Electrode should collect voltage form the skin (few millivolts). Signal from the jack should be amplified and filtered. Therefore, ECG click can be divided into seven blocks.

1. Protection - Provides ESD protection (protects click), Overvoltage and Overcurrent protection (protects the human). In addition to protection, the input block filters and prevents radio waves fro “enter” the preamplifier.
2. Preamplifier – is implemented through three operational amplifiers configured as an instrumentation amplifier (IA – amplifies the voltage difference between “+” and “-“ electrode) which at its output provides a single-end signal.
3. High-Pass filter – should eliminate the DC component of the signal ( $f_c = 1.6\text{Hz}$ ). It is a passive RC filter (first order).
4. Amplifier – provides additional amplification that can be adjusted using the trimmer potentiometer VR1. This is to that the analog output could accommodate to the input voltage range of the ADC. The amplifier is implemented using an operational amplifier in a non-inverting configuration.

5. High-Pass filter – Should eliminate the DC component of the signal ( $f_c = 0.16$  Hz) this time after the amplifier. It is also a passive RC filter (first order).
6. Low-Pass filter – Should limit the frequency range to 60Hz. It is a third order active filter with a gain of 15 (second order Sallen-Key filter topology + passive RC filter first order = third order filter).
7. DRL circuit (Driven Right Leg) – this type of circuit is often added to biological signal amplifiers to reduce Common-mode interference. Amplifiers for bio signals, such as ECG, measure very small electrical signals emitted by the body (measured in milliVolts or even microVolts). Unfortunately, the human body can also act as an antenna which picks up electromagnetic interference, (particularly 50/60 Hz noise from electrical power lines. This interference can obscure the biological signals, making them hard to measure. Right Leg Driver circuitry is used to eliminate interference noise by actively canceling it. It is a selective amplifier stage that shifts the phase of signal for  $180^\circ$  (inverting) and returns it to respondents in order to cancel.

## Hardware configuration jumpers and settings

ECG click has **one jumper and a trimmer potentiometer** for setting the output voltage to match the input voltage level of the ADC which will be used.

The SMD (0805) jumper determines its output voltage range. When you connect all three electrodes to each other, the output should be a constant voltage (either 1.024V or 2.048V depending on the jumper position). This value is the zero-voltage line on the graph. The positive part of the ECG waveform will go above zero and the negative part will go below zero.

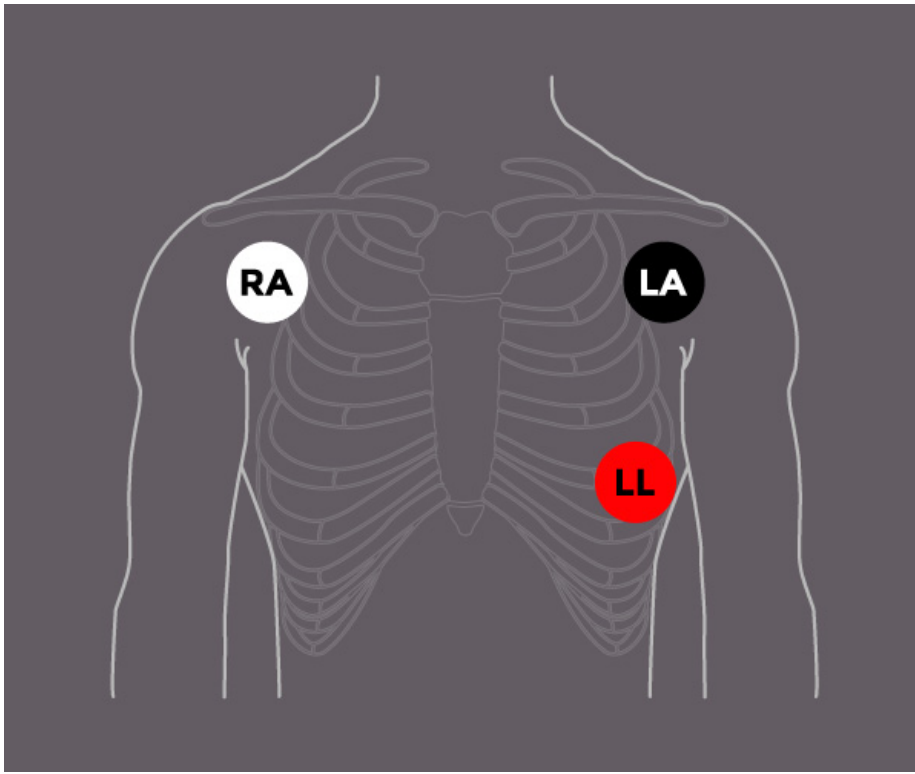
The trimmer potentiometer is for adjusting the gain. So if we set the jumper to 2.048 position (zero is now 1.024V) that means that the gain should be set so that the ECG waveform is in the range of 0-2.048V. If we set the jumper to the 4.096 position (zero is now 2.048V), the gain should be set so that the ECG waveform is in the range of 0-4.096V.

For optimal results, the target board MCU should have at least a 10-bit ADC (12-bit recommended). Sampling rate should be at least 256Hz.

We also recommended that the target development board is **powered from the battery** via a connector for external power supply.

## Hooking up the electrodes

Electrocardiography is performed by measuring the heart's electrical activity through electrodes taped to the surface of the skin. The three electrodes are placed on the left arm, right arm and the left side of the abdomen (below the heart), on the left leg.



Here's how the electrodes are marked and color coded:

- Left leg (LL) – red
- Left arm (LA) – black
- Right arm (RA) – white

Tips for better results:

- Try not to move much while measuring, because the electrodes will pick up noise from muscle activation
- Experiment with the positioning of the disposable electrode pads
- Use new pads for each measurement
- Clean the skin area where you want to apply the pads

If you want visual instructions, see our ECG click setup guide <sup>[5]</sup>.

## Programming

The code shown here initializes the ADC for measuring the ECG, and UART for talking to the PC. When the button is pressed, the timer interrupt will be initialized. An interrupt is generated every 3.9 ms, upon which the MCU gathers ADC and time data, and sends it to the PC over UART, until the timer reaches 15 seconds.

```
#include <stdint.h>
#include <stdbool.h>

void InitTimer1() {
    T1CON                = 0x8010;
    T1IP0_bit            = 1;
    T1IP1_bit            = 1;
```

```
T1IP2_bit      = 1;
T1IF_bit       = 0;
T1IE_bit       = 1;
PR1            = 39000;
TMR1           = 0;
}

static bool read_flag = false;
static volatile uint32_t interrupt_ctr = 0;
uint32_t adc_reads_ctr = 0;
uint32_t timer_reads_ctr = 0;
static volatile uint32_t seconds_counter = 0;
double adc_reads;
double timer_reads;

void main()
{
    uint32_t i = 0;
    char timer_read_string[10];
    char final_string [20];

    TRISB8_bit = 1;           // Set RA1 pin as input
    TRISA15_bit = 1;
    adlpcfg = 0x0100;
    ADC1_Init();

    UART5_Init(57600);
    delay_ms(500);

    while(1)
    {

        if (Button(&PORTA, 15, 10, 1))
        {
            Uart5_Write_Text("START\r\n");
            delay_ms(500);
            InitTimer1();
            EnableInterrupts();
        }

        if (read_flag == true)
        {
            T1IE_bit      = 0;
            adc_reads     = ADC1_Get_Sample(8);
            timer_reads   = ((double)interrupt_ctr)*3.9;
            inttostr(adc_reads, final_string);
            floattostr(timer_reads, timer_read_string);
        }
    }
}
```

```
        strcat(final_string, ",");
        strcat(final_string, timer_read_string);
        ltrim(final_string);
        Uart_Write_Text(final_string);
        Uart_Write_Text("\r\n");
        read_flag = false;
        T1IE_bit = 1;
    }

    if (seconds_counter == 15)
    {
        T1IE_bit = 0; // disable global interrupts
        Uart_Write_Text("END\r\n");
        while(1);
    }

}

}

void Timer1Interrupt() iv IVT_TIMER_1 ilevel 7 ics ICS_SRS
{
    read_flag = true;

    interrupt_ctr++;
    if (interrupt_ctr % 256 == 0)
        seconds_counter++;

    T1IF_bit = 0;
}
```

Code examples for ECG click, written for MikroElektronika hardware and compilers are available on Libstock <sup>[6]</sup>.

## mikroPlot

MikroPlot is a graph generator. It's a simple tool to help you visualize sensor data recorded over time. It's suitable for bio signals (ECG, EEG, EMG) as well as environmental data logging (temperature, humidity etc). The app can receive data sets directly from a microcontroller through a USB UART connection.

A tutorial for generating an EKG in mikroPlot is available on learn.mikroe.com <sup>[7]</sup>

## Resources

- MCP609 Vendor's Data Sheet <sup>[1]</sup>
- MAX6106 Vendor's Data Sheet <sup>[2]</sup>
- ECG click and mikroPlot learn article <sup>[7]</sup>
- EKG click Libstock example <sup>[6]</sup>
- mikroBUS™ standard specifications <sup>[8]</sup>

## References

- [1] <http://ww1.microchip.com/downloads/en/DeviceDoc/11177f.pdf>
  - [2] <https://datasheets.maximintegrated.com/en/ds/MAX6100-MAX6107.pdf>
  - [3] <http://www.mikroe.com/click/ecg>
  - [4] [http://cdn-docs.mikroe.com/images/1/10/ECG\\_click\\_schematic\\_v100.pdf](http://cdn-docs.mikroe.com/images/1/10/ECG_click_schematic_v100.pdf)
  - [5] <https://youtu.be/115Yw2S4Veo>
  - [6] <http://libstock.mikroe.com/projects/view/1936/ecg-click>
  - [7] <http://learn.mikroe.com/ecg-click-mikroplot-complete-solution-human-heart-data-analysis/>
  - [8] <http://download.mikroe.com/documents/standards/mikrobus/mikrobus-standard-specification-v200.pdf>
-

# Article Sources and Contributors

**ECG click** *Source:* <http://docs.mikroe.com/index.php?oldid=1715> *Contributors:* Srdjan.misic

## Image Sources, Licenses and Contributors

**File:ECG\_click.jpg** *Source:* [http://docs.mikroe.com/index.php?title=File:ECG\\_click.jpg](http://docs.mikroe.com/index.php?title=File:ECG_click.jpg) *License:* unknown *Contributors:* Srdjan.misic

**file:Ecg-electrode-v.png** *Source:* <http://docs.mikroe.com/index.php?title=File:Ecg-electrode-v.png> *License:* unknown *Contributors:* Srdjan.misic

**file:Ek1-cable-v.png** *Source:* <http://docs.mikroe.com/index.php?title=File:Ek1-cable-v.png> *License:* unknown *Contributors:* Srdjan.misic

**file:Ecg-bundle-v.png** *Source:* <http://docs.mikroe.com/index.php?title=File:Ecg-bundle-v.png> *License:* unknown *Contributors:* Srdjan.misic

**file:Electrode-placement.jpg** *Source:* <http://docs.mikroe.com/index.php?title=File:Electrode-placement.jpg> *License:* unknown *Contributors:* Srdjan.misic

## License

---

Creative Commons Attribution  
<https://creativecommons.org/licenses/by/4.0/>