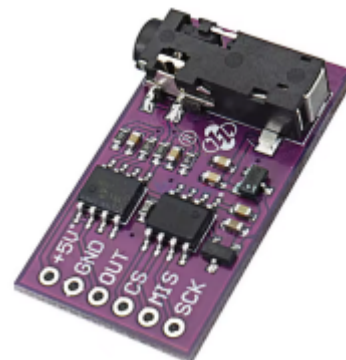


# LamaPLC: CJMCU-6701: Biosensor for measuring Galvanic Skin Response (GSR) with SPI communication

The **CJMCU-6701** is a specialized biosensor module used for measuring *Galvanic Skin Response (GSR)*, also known as *Electrodermal Activity (EDA)*. Unlike the **ACS758 current sensors**, this module measures skin conductance to infer emotional or physiological arousal, such as stress or excitement.



## Key Features

- **Dual Interface:** Supports both Analog (direct voltage output) and [SPI communication](#) for versatile integration.
- **Built-in ADC:** Features a precise 12-bit Analog-to-Digital Converter for digital processing.
- **Voltage Support:** Compatible with both 3.3V and 5V power supplies.
- **Form Factor:** Compact design often including a 3.5mm headphone jack or pads for connecting skin electrodes.

## Skin electrodes for modul

The CJMCU-6701 GSR module typically pairs with finger-strap electrodes that connect via a 3.5mm jack on the backside of the board. These electrodes measure the ion flow from sweat gland activity on the skin surface.



## Types of Compatible Electrodes

- **Velcro Finger Straps (Most Common):** Reusable straps with metal or conductive fabric contact pads. They are typically worn on the index and middle fingers.
- **Ag/AgCl (Silver-Silver Chloride) Electrodes:** Used for high-precision or research-grade measurements. These can be disposable snap-on pads or reusable cup electrodes that provide a very stable electrical interface.
- **Conductive Dry Pads:** Simple metal plates often found in DIY kits. These are durable but more sensitive to movement noise than gelled or velcro options.

## Electrode Placement & Maintenance

- **Optimal Sites:** The fingertips or palms are the most responsive areas because they have the

highest density of eccrine sweat glands.

- **Skin Prep:** For best results, the skin should be clean but not scrubbed with alcohol, as overly dry skin can lower the baseline conductance.
- **Contact Pressure:** Velcro straps should be snug but not tight. Too much pressure can restrict blood flow, while too little causes erratic “spikes” in your data.



If you'd like to support the development of the site with the price of a coffee — or a few — [please do so here](#).

Here's a handy tip: you can quickly save this page as a PDF by clicking “*export to PDF*” in the menu on the right side of the screen.

2026/02/14 23:38

## Pinout



Pin	Function	Connection Type
<b>+5V</b>	Power Supply	( <b>3.3V/5V</b> ) Power Input
<b>GND</b>	Ground	Common Ground
<b>OUT</b>	Analog Output	Analog Pin (e.g., A0)
<b>CS</b>	Chip Select	SPI Interface
<b>MIS</b>	Master In Slave Out	SPI Interface
<b>SCK</b>	Serial Clock	SPI Interface

## Backside Port Functions

- **3.5mm Audio Jack (Standard):** The primary port for connecting the GSR electrode cable. It allows for a secure, plug-and-play connection to the finger electrodes.
- **Solder Pads (Alternative):** Some versions of the board feature labeled pads on the back (**GSR**) for users who want to bypass the headphone jack and solder electrode wires directly to the PCB for a more permanent or compact installation.
- **Potentiometer Access:** While the adjustment screw is on the front, the solder points for the Sensitivity Calibration Potentiometer are visible on the back. This component adjusts the baseline output voltage to accommodate different skin types.

## Option 1: Arduino Analog Wiring (Recommended for simplicity)

This is the easiest way to get started. You only need three wires for data/power and the electrode cable plugged into the back.

CJMCU-6701 Pin	Arduino Pin	Note
VCC	5V	Power supply
GND	GND	Common ground
OUT	A0	Analog Input 0
Back Jack	Electrodes	Plug in electrode cable

This code reads the raw sensor value and performs basic averaging to filter out noise common in biosignals.

```

const int gsrPin = A0; // Sensor OUT connected to A0
int sensorValue = 0;
int gsrAverage = 0;

void setup() {
  Serial.begin(9600); // Initialize Serial Monitor
}

void loop() {
  long sum = 0;

  // Take 10 samples and average them to reduce noise
  for(int i = 0; i < 10; i++) {
    sensorValue = analogRead(gsrPin);
    sum += sensorValue;
    delay(5);
  }

  gsrAverage = sum / 10;

  // Print the raw average value (0-1023)
  Serial.print("GSR Raw Value: ");
  Serial.println(gsrAverage);

  // Optional: Convert to Voltage
  float voltage = gsrAverage * (5.0 / 1023.0);
  Serial.print("Voltage: ");
  Serial.println(voltage, 2);

  delay(100);
}

```

## Option 2: Arduino SPI Wiring (For better noise immunity/precision)

Using SPI uses four data lines but provides a direct digital reading, bypassing the Arduino's built-in ADC for potentially cleaner data.

CJMCU-6701 Pin	Arduino UNO Pin	Arduino Mega Pin
VCC	5V	5V
GND	GND	GND

CJMCU-6701 Pin	Arduino UNO Pin	Arduino Mega Pin
CS	D10	D53
SCK	D13	D52
MIS	D12	D50
<b>Back Jack</b>	Electrodes	Plug in electrode cable

In this CJMCU-6701, there is no **MOSI** pin because the onboard ADC (often a Microchip MCP3201) is read-only. It doesn't need instructions from the Arduino; it simply "spits out" data when clocked.

This sketch uses the standard Arduino SPI Library to read the 12-bit digital value.

```
#include <SPI.h>

const int csPin = 10; // Chip Select pin

void setup() {
  Serial.begin(9600);
  pinMode(csPin, OUTPUT);
  digitalWrite(csPin, HIGH); // Ensure sensor is disabled initially

  // Initialize SPI: Speed 1MHz, MSB first, SPI Mode 0
  SPI.begin();
}

void loop() {
  // Start SPI transaction
  SPI.beginTransaction(SPISettings(1000000, MSBFIRST, SPI_MODE0));
  digitalWrite(csPin, LOW); // Select the sensor

  // Read 2 bytes (16 bits total) from the sensor
  // Since it's read-only, we send 0x00 to provide clock pulses
  byte highByte = SPI.transfer(0x00);
  byte lowByte = SPI.transfer(0x00);

  digitalWrite(csPin, HIGH); // Deselect the sensor
  SPI.endTransaction();

  // Combine bytes and extract 12-bit value
  // MCP3201 format: 2 leading zero bits, 1 null bit, then 12 data bits, 1
  // extra bit
  int rawValue = ((highByte & 0x1F) << 7) | (lowByte >> 1);

  Serial.print("GSR Digital Value: ");
  Serial.println(rawValue);

  delay(200); // Sample rate adjustment
}
```

## SPI topics on lamaPLC

Page	Date	Tags
• <a href="#">lamaPLC Communication: SPI</a>	2026/04/23 21:51	bus, communication, spi, basic, arduino, ssi, sdi, miso, sdo
• <a href="#">lamaPLC: AI-Thinker LoRA products</a>	2026/04/23 21:51	ai-thinker, lora manufacturer, communication, lora, modul, ra-01, ra-02, spi, arduino
• <a href="#">lamaPLC: Bi-Directional Logic Level Converter 3.3V ↔ 5V</a>	2026/04/12 00:34	bi-directional, logic level converter, i2c, uart, spi
• <a href="#">LamaPLC: CJMCU-3901/PMW-3901 compact optical flow sensor module/IC by PixArt with SPI communication</a>	2026/04/23 21:52	cjmcu-3901, cjmcu, pmw3901, pmw-3901, optical flow, sensor, pixart, spi, communication, arduino, code, pmw3901mb-txqt
• <a href="#">LamaPLC: CJMCU-6701: Biosensor for measuring Galvanic Skin Response (GSR) with SPI communication</a>	2026/04/23 21:52	cjmcu, cjmcu-6701, acs758, acs-758, galvanic skin response, gsr, electrodermal activity, eda, spi, communication, arduino, code, sensor, healthcare
• <a href="#">lamaPLC: ESP32 / ESP8266</a>	2025/11/22 00:07	esp8266, esp32, esp32-c2, esp32-c3, esp32-c5, esp32-c6, esp32-c61, esp32-h2, esp32-s2, esp32-s3, esp32-p4, espressif systems, communication, ethernet, ip, wi-fi, thread, zigbee, matter, homekit, bluetooth, mqtt, adc, spi, uart, i2c, i2s, rmt, pwm, usb, usb otg, twai
• <a href="#">LamaPLC: GY-9250 MPU-9250/6500 9-axis Attitude Sensor Board</a>	2026/04/23 21:52	ak8963, gy-9250, mpu-9250, 9-axis, motion detection, magnetometer, communication, i c, i2c, spi
• <a href="#">lamaPLC: Max31865 RTD to Digital Converter - PT100/PT1000 Platine</a>	2026/04/23 21:52	max31865, rtd, pt 100, pt 1000, temperature, spi, platinum, arduino, code, sensor, adafruit
• <a href="#">lamaPLC: MCP23017 / MCP23S17 16-Bit I/O Expander with Serial Interface I<sup>2</sup>C / SPI</a>	2026/04/23 21:52	communication, i2c, mcp23017, mcp23s17, spi, i o expander, serial, cjmcu-2317, cjmcu
• <a href="#">LamaPLC: SC16IS750 / SC16IS752: One or two serial (UART) ports from microcontroller via I<sup>2</sup>C or SPI communication</a>	2026/04/23 21:52	cjmcu-750, cjmcu-752, cjmcu, nxp, sc16is750, sc16is752, uart, serial, i2c, spi, modul, converter, arduino, code
• <a href="#">lamaPLC: st756x display drivers</a>	2026/05/20 16:17	display, driver, i2c, spi, lcd, cog, oled, st7565, st7567, gm12864, gm12864-59n, gm12864-03a, gm12864-01a, gme12864-41
• <a href="#">LamaPLC: ST7789 driver / TFT-LCD screen</a>	2026/05/19 21:37	communication, spi, st7789, tft, lcd, screen, ips
• <a href="#">LamaPLC: Texas Instruments ADCs: Delta-sigma multi-channel Analog Converters with SPI communication</a>	2026/04/23 21:52	ads111x, ads12xx, delta-sigma, converter, texas instruments, adc, spi, communication, sensor, arduino, code, ads1110, ads1112, ads1113, ads1114, ads1115, ads1118, ads1119, ads1220, ads1232, ads1234, ads1256, ads1261, ads1263, multi channel

- [lamaPLC: Waveshare LoRA products](#) 2026/03/07 01:46 [waveshare, lora manufacturer, communication, lora, modul, usb-to-lora-xf02, core 1262, 1262, spi, arduino, rp2040-lora, rp2040](#)
- [Magnetic angle sensors](#) 2026/03/05 21:19 [magnetic angle sensor, magnetic flux, sensor, spi, i2c, pwm, communication, modul, as5047p, as5600, mt6701, mt6816, mt6835, tle5012b, amr, gmr, tmr, anisotropic magnetoresistive](#)

[CJMCU](#), [CJMCU-6701](#), [ACS758](#), [ACS-758](#), [Galvanic Skin Response](#), [GSR](#), [Electrodermal Activity](#), [EDA](#), [SPI](#), [communication](#), [Arduino](#), [code](#), [sensor](#), [healthcare](#)

This page has been accessed for: Today: 2, Until now: 77

From:  
<https://www.lamaplc.com/> - **lamaPLC**

Permanent link:  
[https://www.lamaplc.com/doku.php?id=sensor:cjmcu\\_6701](https://www.lamaplc.com/doku.php?id=sensor:cjmcu_6701)

Last update: **2026/04/21 20:47**

