**SIM900A GSM/GPRS Module**

****

**1.Detail:**

**The SIM900A GSM Module with an antenna is a compact and versatile device used for connecting microcontrollers (like Arduino) to a cellular network. It allows projects to perform mobile communication functions, including sending/receiving SMS, making/receiving voice calls, and accessing the internet via GPRS.**

**2. How GSM sensors work:**

**A GSM sensor system is a combination of a sensor, a microcontroller, and a GSM module that work together to transmit data over a mobile network. The sensor collects data, the microcontroller processes it, and the GSM module sends the information via SMS or GPRS to a remote user or server.**

**3.How GSM work:**

**VCC: Connect to a 5V, 2A external power supply.**

**GND: Connect to the ground of your external power supply and microcontroller.**

**TXD: Connect to the RX pin of your microcontroller (e.g., Arduino D10).**

**RXD: Connect to the TX pin of your microcontroller (e.g., Arduino D11).**

**PWRKEY: Connect to a digital pin on your microcontroller (e.g., Arduino D9) and pull low for one second to power on/off.**

**ANT: Connect to the external GSM antenna.**

**4.Test Code:**

#include <stdio.h>

#include <string.h>

#include <fcntl.h>

#include <errno.h>

#include <termios.h>

#include <unistd.include>

// Function to send AT commands and wait for a response

void send\_at\_command(int serial\_port, const char \*command, const char \*expected\_response) {

// Implementation details for sending commands and checking responses

}

int main() {

const char \*serial\_port\_path = "/dev/ttyUSB0"; // Adjust this if your device is different

const char \*phone\_number = "+911234567890"; // REPLACE with the recipient's number

const char \*sms\_text = "hello";

// Open and configure the serial port

int serial\_port = open(serial\_port\_path, O\_RDWR | O\_NOCTTY | O\_SYNC);

if (serial\_port < 0) { /\* Handle error \*/ return 1; }

struct termios tty;

if (tcgetattr(serial\_port, &tty) != 0) { /\* Handle error \*/ }

// Configure serial settings

cfsetispeed(&tty, B9600);

cfsetospeed(&tty, B9600);

// Other tty settings configuration...

if (tcsetattr(serial\_port, TCSANOW, &tty) != 0) { /\* Handle error \*/ }

tcflush(serial\_port, TCIOFLUSH);

sleep(10); // Wait for module to boot

// Send AT commands to initialize and send SMS

send\_at\_command(serial\_port, "AT\r", "OK");

send\_at\_command(serial\_port, "AT+CMGF=1\r", "OK");

// Send CMGS command with phone number

// Send message content

// Send Ctrl+Z to indicate end of message

// Read final response

close(serial\_port);

return 0;

}  
