

# Беспроводной джойстик PS2

Указания к использованию



## Оглавление

1 Описание джойстика PS2.....	3
2 Указания к использованию и подключению джойстика .....	5
3 Подключение к компьютеру.....	6
4 Составление скриптов.....	6
5 Загрузка и проверка .....	12



## 1 Описание джойстика PS2

Джойстик PS2 состоит из управляющего и приемного устройств, главная задача самого джойстика – отправка сигналов по нажатию клавиш. Приемник и микроконтроллер (далее – хост, например, сама игровая консоль PS2) сопряжены друг с другом. Информация с джойстика поступает на хост, который также с помощью приемника передает команды на джойстик и устанавливает режим отправки данных с джойстика.

Особое указание: внешний вид джойстиков и приемников может отличаться в зависимости от партии или завода-изготовителя (на всех приемниках имеются индикаторы, но на некоторых есть дополнительные индикаторы питания). Назначение контактов у всех приемников и тип дешифрования одинаковые.

Выходные контакты приемника:

1	2	3	4	5	6	7	8	9
DI/DAT	DO/CMD	NC	GND	VDD	CS/SEL	CLK	NC	ACK

Приемник показан на фото ниже (обратите внимание на порядок портов).

DI/DAT: сигнал передается от джойстика к хосту, последовательная передача 8 бит данных, синхронная передача по заднему фронту тактового сигнала. Сигнал считывается при перепадах сверху вниз тактового сигнала.

DO/CMD: сигнал передается от хоста к джойстику, в обратном направлении сигналу порта DI, последовательная передача 8 бит данных, синхронная передача по заднему фронту тактового сигнала.

NC: неиспользуемый контакт.

GND: заземление.

VDD: питание приемника, диапазон напряжения 3-5 В.

CS/SEL: для подачи триггерного сигнала джойстика, во время передачи данных сигнал передается на низком уровне.

NC: неиспользуемый контакт.

ACK: ответный сигнал с джойстика на хост, понижается по окончании цикла передачи 8-битных данных, в то время как CS сигнал всегда на низком уровне. Если сигнал CS не понижается, то через 60 мс хост PS попытается подключиться к другому периферийному устройству. При составлении скрипта ACK не задействуется.



Рисунок 1 – Обозначение интерфейсов приемника

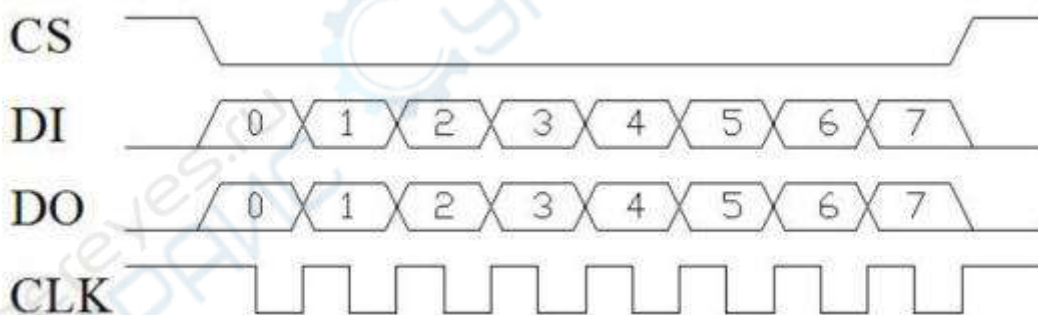


Рисунок 2 Временная диаграмма интерфейса

**Тактовая частота 250 кГц (4 мкс). Если прием данных нестабильный, допускается увеличить тактовую частоту.**

После передачи одной серии данных уровень сигнала CS переключается с низкого на высокий, при передаче 1 байта переключения не происходит. В процессе обмена данными уровень сигнала CS всегда низкий.

Отправка и прием данных (1 бит) выполняются синхронно по заднему фронту тактового сигнала.

Когда хост захочет считать данные с джойстика или отправить команду на джойстик, он понизил уровень сигнала CS и отправит команду «0x01». В ответ джойстик отправит свое ID «0x41=режим зеленого индикатора, 0x73=режим красного индикатора». Одновременно при отправке ID джойстика хост отправит запрос данных, команду «0x42». Следом джойстик отправит в ответ хосту «0x5A», означающую, что данные приняты.

idle: канал данных пуст, передача отсутствует.  
За один цикл побитно передается 9 байтов (8 знаков).

Таблица 1 – Сопоставительная таблица данных с интерфейсами

№	DO	DI	Bit0, Bit1, Bit2, Bit3, Bit4, Bit5, Bit6, Bit7
0	0X01	idle	
1	0x42	ID	
2	idle	0x5A	
3	WW	data	SELECT, L3, R3, START, UP, RIGHT, DOWN, LEFT
4	YY	data	L2, R2, L1, R1, $\Delta$ , $\circ$ , $\times$ , $\square$
5	idle	data	PSS_RX (0x00=left, 0xFF=right)
6	idle	data	PSS_RY (0x00=up, 0xFF=down)
7	idle	data	PSS_LX (0x00=left, 0xFF=right)
8	idle	data	PSS_LY (0x00=up, 0xFF=down)

При нажатии клавиш передаются массивы данных из «0» и «1». Например, при нажатии кнопки SELECT передается массив Data[3]=1111110B.

После установки режима вибрации мы можем отправить WW, YY, чтобы управлять вибромотором. WW используется для управления меньшим вибромотором с правой стороны, при «0x00» вибромотор отключен, при других значениях включен. YY используется для управления мощным вибромотором слева, «0x40-0xFF» включает вибромотор, чем большее значение, тем сильнее вибрация. Конкретные настройки представлены ниже, в главе о составлении скриптов.

**Режим красного индикатора:** при движении рычажков вправо-влево команды меняются в диапазоне «0x00-0xFF», также при нажатии рычажков активны значения кнопок L3, R3 соответственно.

**Режим зеленого индикатора:** аналоговые значения правого и левого рычажков неактивны. Когда рычажки находятся в крайних положениях, они посылают команды, соответствующие UP, RIGHT, DOWN, LEFT,  $\Delta$ ,  $\circ$ ,  $\times$ ,  $\square$ . Кнопки L3, R3 не активны.

## 2 Указания к использованию и подключению джойстика

Необходимо вставить в джойстик две батарейки номиналом 1.5 В. Приемник и микроконтроллер питаются от одного источника, диапазон напряжения 3-5 В. Неправильное подключение, работа при повышенных и пониженных напряжениях могут привести к поломке приемника.

На джойстике есть кнопка включения: ON – включено, OFF – выключено. Если после включения джойстик не обнаружил приемник, будет мерцать красный индикатор. Если в установленное время джойстик не свяжется с приемником, он переключится в режим ожидания, красный индикатор погаснет. Чтобы выйти из режима ожидания, нажмите кнопку START.

Если на приемник поступает питание, но устройства не сопрягаются, будет мигать зеленый индикатор.

Включите джойстик и приемник, джойстик и приемник должны связаться. После успешного сопряжения загорится индикатор. Нажмите кнопку MODE (на некоторых джойстиках эта кнопка называется ANALOG), выберите «режим красного индикатора» или «режим зеленого индикатора».

Некоторые пользователи пишут, что джойстик и приемник не сопрягаются должным образом. Обычно это связано с неправильным подключением приемника или ошибками в скрипте.

Решение проблемы: подключите приемник напрямую к источнику питания (питающий кабель должен быть без дефектов и подключен верно), не подсоединяйте никаких кабелей передачи данных и синхронизирующего сигнала. При корректном подключении джойстик сразу же свяжется с приемником и загорится индикатор. В противном случае еще раз проверьте подключение и скрипт на наличие ошибок.

### 3 Подключение к компьютеру

Способы соединения приемника и stm32:

DI-> PB12;  
DO->PB13;  
CS->PB14;  
CLK->PB15.

### 4 Составление скриптов

Полные скрипты программ находятся в файле pstwo.c, ознакомьтесь с ним подробнее.

```
void PS2_Init(void)
{
    //вход DI->PB12
    RCC->APB2ENR|=1<<3; //запуск тактового сигнала с PORTB
    GPIOB->CRH&=0xFFFF0FFF; // настройки входа PB12, ставим по умолчанию
    GPIOB->CRH|=0X00080000;
    // DO->PB13      CS->PB14      CLK->PB15
    RCC->APB2ENR|=1<<3; // запуск тактового сигнала с PORTB
    GPIOB->CRH&=0X000FFFFF;
    GPIOB->CRH|=0X33300000; //выставляем выходы PB13, PB14, PB15
}
```

Инициализация портов: PB12 – вход; PB13, PB14, PB15 – выходы.

```
//отправляем команды на джойстик
void PS2_Cmd(u8 CMD)
{
    volatile u16 ref=0x01;
    Data[1] = 0;
    for(ref=0x01; ref<0x0100; ref<<=1)
    {
        if(ref&CMD)
        {
            DO_H; //выводим контрольный бит
        }
        else DO_L;
        CLK_H; //повышение тактового сигнала
        delay_us(10);
    }
}
```

```

        CLK_L;
        delay_us(10);
        CLK_H;
        If(DI)
            Data[1] = ref[Data[1];
    }
    delay_us(16);
}
//оцениваем, включен ли режим красного индикатора, 0x41=имитация зеленого
индикатора, 0x73=имитация красного индикатора
//ответное значение: 0, режим красного индикатора
//                другое, другие режимы

u8 PS2_RedLight(void)
{
    CS_L;
    PS2_Cmd(Comd[0]); //начало команды
    PS2_Cmd(Comd[1]); //запрос данных
    CS_H;
    if( Data[1] == 0X73)        return 0;
    else return 1;
}
//считывание данных с джойстика
void PS2_ReadData(void)
{
    volatile u8 byte=0;
    volatile u16 ref=0x01;
    CS_L;
    PS2_Cmd(Comd[0]);        //начало команды
    PS2_Cmd(Comd[1]);        //запрос данных
    for(byte=2;byte<9;byte++) //начать прием данных
    {
        for(ref=0x01; ref<0x100; ref<<=1)
        {
            CLK_H;
            delay_us(10);
            CLK_L;
            delay_us(10);
            CLK_H;
            if(DI)
                Data[byte] = ref[Data[byte];
        }
        delay_us(16);
    }
    CS_H;
}

```

Две функции, описанные выше, предназначены для отправки данных с хоста на джойстик и с джойстика на хост. Данные, отправляемые с джойстика на хост, сохраняются в виде массива Data[], состоящего из 9-и элементов, значения элементов показаны в

таблице 1. Еще одна функция определяет режим отправки данных с джойстика, а также устанавливает ID, то есть значение Data[1].

```
//обработка данных, считанных с PS2, только для управления кнопками
// кнопка нажата – 0, не нажата – 1
u8 PS2_DataKey()
{
    u8 index;
    PS2_ClearData();
    PS2_ReadData();
    Handkey=(Data[4]<<8)|Data[3];    //всего 16 кнопок, нажата – 0, не нажата – 1
    for(index=0;index<16;index++)
    {
        if(Handkey&(1<<(MASK[index]-1))==0)
            return index+1;
    }
    return 0;    //нет нажатых кнопок
}
//получаем аналоговые значения для рычажков, диапазон 0-256
u8 PS2_AnalogData(u8 button)
{
    return Data[button];
}
//очищаем область данных
void PS2_ClearData()
{
    u8 a;
    for(a=0; a<9; a++)
        Data[a]=0x00;
}
```

Восьмибитные массивы Data[3] и Data[4] соответствуют положениям 16-и кнопок, при нажатии передается 0, при отсутствии нажатия – 1. Обработывая значения массивов, получаем информацию о положении кнопок и формируем ответное значение кнопки.

Другая опция этой функции – формирование ответного аналогового значения, что возможно только в «режиме красного индикатора». Значения изменяются только при движении рычажков, и по отдельности хранятся в Data[5], Data[6], Data[7], Data[8].

Инициализация конфигурации джойстика:

```
//short poll
void PS2_ShortPoll(void)
{
    CS_L;
    delay_us(16);
    PS2_Cmd(0x01);
    PS2_Cmd(0x42);
    PS2_Cmd(0x00);
    PS2_Cmd(0x00);
    PS2_Cmd(0x00);
    CS_H;
    delay_us(16);
}
```



```

//входим в настройки конфигурации
void PS2_EnterConfinf(void)
{
    CS_L;
    delay_us(16);
    PS2_Cmd(0x01);
    PS2_Cmd(0x43);
    PS2_Cmd(0X00);
    PS2_Cmd(0x01);
    PS2_Cmd(0x00);
    PS2_Cmd(0X00);
    PS2_Cmd(0X00);
    PS2_Cmd(0X00);
    PS2_Cmd(0X00);
    CS_H;
    delay_us(16);
}
//настройки режима отправки
void PS2_TurnOnAnalogMode(void)
{
    CS_L;
    PS2_Cmd(0x01);
    PS2_Cmd(0x44);
    PS2_Cmd(0X00);
    PS2_Cmd(0x01); //analog=0x01; digital=0x00 ПО настраивает режим отправки
    PS2_Cmd(0xEE); //0x03 блокирует настройки, нельзя задать настройки при
                    //помощи кнопки MODE
                    //0xEE не блокирует сохраненные настройки ПО, можно задать
                    //настройки при помощи кнопки MODE
    PS2_Cmd(0X00);
    PS2_Cmd(0X00);
    PS2_Cmd(0X00);
    PS2_Cmd(0X00);
    CS_H;
    delay_us(16);
}
//Настройки вибрации
void PS2_VibrationMode(void)
{
    CS_L;
    delay_us(16);
    PS2_Cmd(0x01);
    PS2_Cmd(0x4D);
    PS2_Cmd(0X00);
    PS2_Cmd(0x00);
    PS2_Cmd(0X01);
    CS_H;
    delay_us(16);
}
//завершаем и сохраняем настройки конфигурации
void PS2_ExitConfig(void)

```

```

{
    CS_L;
    delay_us(16);
    PS2_Cmd(0x01);
    PS2_Cmd(0x43);
    PS2_Cmd(0X00);
    PS2_Cmd(0x00);
    PS2_Cmd(0X5A);
    PS2_Cmd(0X5A);
    PS2_Cmd(0X5A);
    PS2_Cmd(0X5A);
    PS2_Cmd(0X5A);
    CS_H;
    delay_us(16);
}
//Инициализация конфигурации джойстика
void PS2_SetInit(void)
{
    PS2_ShortPoll();
    PS2_ShortPoll();
    PS2_ShortPoll();
    PS2_EnterConfig(); // вход в настройки конфигурации
    PS2_TurnAnalogModeOn; // конфигурация режимов «красного» или «зеленого
                          индикатора», сохранить или нет
    PS2_VibrationMode(); //включение режима вибрации
    PS2_ExitConfig(); // завершение и сохранение конфигурации
}

```

Как видно из настроек функции, она отвечает за отправку команд, после чего джойстик автоматически выберет действия.

При инициализации конфигурации джойстика PS2\_ShortPoll() выполняется 3 раза для установки и восстановления соединения. Также в скрипте предоставлены комментарии к конкретным настройкам.

```

/*****
Function:    void PS2_Vibration(u8 motor1, u8 motor2)
Description: функция вибрации джойстика
Calls:      void PS2_Cmd(u8 CMD);
Input:      motor1: вибрации слабого вибромотора справа, «0x00» включен, другие - отключен
            motor 1: вибрации мощного вибромотора слева, «0x40-0xFF» включен, чем выше
            значение, чем сильнее вибрация
*****/
void PS2_Vibration(u8 motor1, u8 motor2)

```

```

{
    CS_L;
    delay_us(16);
    PS2_Cmd(0x01); // начало команды
    PS2_Cmd(0x42); //запрос данных
    PS2_Cmd(0X00);
    PS2_Cmd(motor1);
    PS2_Cmd(motor2);
}

```

```

PS2_Cmd(0X00);
PS2_Cmd(0X00);
PS2_Cmd(0X00);
PS2_Cmd(0X00);
CS_H;
delay_us(16);
}

```

Инициализация вибромоторов и выполнение команды (PS2\_VibrationMode();// запуск режима вибрации) производится только во время инициализации функции void PS2\_SetInit(void).

Пишем главную функцию:

```

int main(void)
{
    u8 key;
    Stm32_Clock_Init(9);//настройки тактового сигнала системы
    delay_init(72); // инициализация задержки
    uart_init(72, 9600); //инициализация последовательного порта 1
    LED_Init();
    PS2_Init(); //запуск инициализации порта
    PS2_SetInit(); // инициализация конфигурации, настройки «режима красного
                    // индикатора», можно выбрать отредактировать или нет
                    // запуск режима вибрации
    while(1)
    {
        LED=!LED;
        key=PS2_DataKey();
        if(key!=0) //кнопка действительна
        {
            print(" \r\n  %d  id    pressed    \r\n", key);
            if(key == 11)
            {
                PS2_Vibration(0xFF,0x00); // задержка после вибрации
                delay_ms(1000);
                delay_ms(500);
            }
            else if(key == 12)
            {
                PS2_Vibration(0x00,0x41); // задержка после вибрации
                delay_ms(1000);
                delay_ms(500);
            }
            else
                PS2_Vibration(0x00, 0x00);
        }
        printf("% 5d% 5d% 5d% 5d\r\n",PS2_AnalogData(PSS_LX),PS2_AnalogData(PSS_LY),
        PS2_AnalogData(PSS_RX), PS2_AnalogData(PSS_RY));
        delay_ms(500);
    }
}

```

Когда кнопка нажата, с нее передается значение.

## 5 Загрузка и проверка

Скомпилируйте скрипт и загрузите его. Если рычажки установлены в режим красного индикатора, индикатор загорится красным, аналоговое значение на выходе последовательных портов будут 127 или 128. При шатании рычажков соответствующие аналоговые значения изменятся, в этот момент можно нажать кнопки джойстика и вывести с них значения, показанные на рисунке 4.



Рисунок 3



Рисунок 4

Нажмите кнопку  $\Delta$ , выведется соответствующее значение «13».



Рисунок 5

Нажмите по отдельности L1 и R1, на последовательных портах появятся значения кнопок и джойстик завибрирует в соответствии с настройками.



Рисунок 6

Нажмите MODE, переключитесь в режим зеленого индикатора, цвет индикатора на джойстике станет зеленым. Аналоговое значение последовательных портов будет 255, легонько наклоните рычажки – значения не изменятся.



Рисунок 7

Наклоните правый рычажок в крайнее положение. На выходе последовательных портов появится «Mispressed», что соответствует значению кнопки  $\Delta$ , но аналоговые значения не изменятся.



Рисунок 8

Главное отличие «режима красного индикатора» и «режима зеленого индикатора» в изменении аналоговых значений при движении рычажков.

Попробуйте изменить настройки при инициализации: будет ли заблокирован выбор режима отправки? Проверьте, появились ли изменения в работе джойстика. Установите значения PS2\_Vibration(0x00, 0x41) и почувствуйте, как джойстик вибрирует с разной частотой.

## 6 Декодирование портов PS2: FAQ

### Вопрос 1. Какое напряжение питания приемника?

Ответ: подойдет напряжение 3.3-5 В, однако если к источнику питания подключена индуктивная нагрузка (например, в рейсовом самолете), из-за скачков напряжения приемник может сгореть. Сейчас мы предоставляем переходную плату PS2, использование которой снижает вероятность выхода из строя приемника при подключении к источнику питания индуктивной нагрузки.

### Вопрос 2. Как понять, что джойстик и приемник в порядке и сопряжены правильно?

Ответ: чтобы понять, может ли джойстик сопрягаться с приемником, подключите приемник к портам VCC и GND без подключения других дата-кабелей. При включении индикатор приемника будет постоянно мигать — это означает, что сопряжение не выполнено. Если индикатор не мигает, значит джойстик и приемник сопряжены и функционируют правильно.

### Вопрос 3: При использовании UNO и PS2 появилась подсказка «Ошибка последовательных портов», что это значит?

Это значит, что контроллер не найден или что-то другое, скорее всего проблемы с подключением проводов. Сперва проверьте, сопряжен ли джойстик с приемником (см.



вопрос 2), в большинстве случаев проблема устраняется при корректном сопряжении. Если ошибка все еще не устранена, внимательно проверьте все подключения, сверьтесь с главой 1 данной инструкции, где указаны порядок и назначение контактов приемника, затем соберите и проверьте скрипт заново.

#### Вопрос 4: При использовании UNO и PS2 как заставить джойстик вибрировать?

Ответ: в нашей сборке вибрации не предусмотрены по умолчанию. Соответствующая подпрограмма Arduino состоит примерно из 20 строк, показанных на рисунке 9.

```
/*
*****
* select modes of PS2 controller:
* - pressures = analog reading of push-buttons
* - rumble    = motor rumbling
* uncomment 1 of the lines for each mode selection
*****
//#define pressures true
#define pressures false
//#define rumble true
#define rumble false
*/
```

Рисунок 9

Закомментированные части нужно поменять местами:

```
#define pressures true
//#define pressures false
#define rumble true
//#define rumble false
```

Таким образом можно посредством нажатия кнопки X вызывать вибрации различной амплитуды, менять их интенсивность можно кнопка «вверх», «вниз», «вправо» и «влево» с левой стороны джойстика, на выходе будут различные аналоговые значения.

#### Вопрос 5: При использовании UNO и PS2 как вывести аналоговые значения рычажков?

Ответ: При движении рычажков одновременно нажмите кнопку L1 или R1.

#### Вопрос 6: После портирования программ под STM32 при нажатии кнопки MODE индикатор MODE не реагирует. Почему?

Причина в настройках программы, изучите первые 200 строк скрипта в файле pstwo.c, функцию void PS2\_TurnOnAnalogMode(void). При редактировании параметров этой функции можно использовать кнопку MODE.

#### Вопрос 7: В чем назначение индикатора MODE?

Ответ: поскольку у джойстика различные поставщики, индикаторы режимов тоже могут отличаться. Индикатор MODE данного джойстика имеет следующее назначение:

1. Индикатор режима: «режим зеленого индикатора» или «режим погасшего индикатора».
2. При неизменном значении ID, «0x41= режим погасшего индикатора, 0x73=режим зеленого индикатора».

3. В режиме зеленого индикатора: правый и левый рычажки отправляют аналоговые значения в диапазоне 0x00-0xFF, также при нажатии рычажков активны кнопки L3, R3.
4. В режиме погасшего индикатора: аналоговые значения правого и левого рычажков неактивны. Когда рычажки находятся в крайних положениях, они посылают команды, соответствующие UP, RIGHT, DOWN, LEFT,  $\Delta$ ,  $\circ$ ,  $\times$ ,  $\square$ . Кнопки L3, R3 не активны.

