

Контроллер периферийного оборудования

Версия 2.0

Оглавление

Функциональные возможности	3
Разъемы контактов.....	3
Установка в операционной системе	4
Коммуникационный протокол	5
Настройки коммуникационного порта	5
Форматы посылок	5
Общие команды.....	6
чтение версии программы.....	6
Команды и события для работы с кнопками	6
чтение состояний всех кнопок/ключей.	6
чтение состояния кнопки/ключа.	6
разрешить/запретить автоматическую посылку событий.....	6
чтение маски разрешения событий	7
Команды по управления лампами	7
включение/выключение ламп.	7
управление свечением лампы.	7
чтение состояний всех ламп.	7
команда чтения состояния лампы.	8
Команды устройства EEPROM	8
чтение памяти.....	8
запись в память	8
Примеры пакетов.....	9
Коды кнопок и ламп.....	9
Приложение. Внешний вид платы контроллера ввода-вывода.....	10

Функциональные возможности

Контроллер периферийного оборудования (далее – контроллер), предназначен для подключения устройств терминала к управляющей плате персонального компьютера.

Взаимодействие контроллера периферийного оборудования и компьютера осуществляется по шине USB. Контроллер реализует USB устройство класса CDC (Communication Device Class). Данный класс позволяет работать с USB, как с последовательным интерфейсом RS232.

Благодаря этому достигается простота и удобство в разработке пользовательского программного обеспечения по работе с контроллером.

В стандартной комплектации контроллера реализована возможность подключения:

12-ти кнопок управления;

3-х электронных ключей;

13-ти лампочек (12V).

Разъемы и переключки на плате:

J1 – разъем для программирования микроконтроллера по интерфейсу C2. Используется для загрузки firmware производителем плат.

J2, J4, J5, J6 – разъемы, для подключения внешнего оборудования (см. раздел “Разъемы контактов”).

J3 – разъем для подключения к блоку вычислителя стандартным кабелем USB A - USB B.

На плате имеется индикатор подачи питания.

желтый индикатор - наличие 12в

зеленый индикатор - наличие 5в

Разъемы контактов

Описание контактов разъема J2

1	не используется
2	лампа INFO
3	лампа CHANGE
4	лампа MENU
5	лампа LINE3
6	лампа LINE5
7	лампа LINE7
8	лампа BET
9	лампа AUTO
10	лампа START
11	12 вольт
12	кнопка CASH OUT
13	кнопка INFO
14	кнопка CHANGE
15	кнопка MENU
16	кнопка LINE3
17	кнопка LINE5
18	кнопка LINE7

19	кнопка BET
20	кнопка AUTO
21	кнопка START
22	общий (GND)

Описание контактов разъема J5.

1	12 вольт
2	лампа LINE1
3	лампа LINE9
4	общий (GND)
5	лампа LINE3
6	лампа LINE5

Описание контактов разъема J4.

1	ключ ADMIN
2	общий (GND)
3	лампа TOP (верхний фонарь)
4	лампа BOT (нижний фонарь)
5	общий (GND)
6	12 вольт
7	датчик MAIN_DOOR
8	датчик BILL_DOOR
9	общий (GND)
10	5 вольт
11	общий (GND)
12	12 вольт

Разъем J6 в этой версии не используется.

Установка в операционной системе

В операционных системах семейства Linux автоматически загружается драйвер `cdc_acm`, который создает устройство `/dev/ttyACM0`.

Рекомендуется создать правило `/etc/udev/rules.d/10-persistent-usb.rules`.

В правиле указать

```
KERNEL=="ttyACM*", ATTRS{idVendor}=="16c0",
ATTRS{idProduct}=="06ea", GROUP="uucp", MODE="0666",
SYMLINK+="ioboard", OPTIONS="last_rule"
```

тогда именно это устройство будет доступно в виде ссылки `/dev/ioboard`.

Для установки в системе Windows необходимо на запрос об установке драйвера указать путь к файлу `ioboard.inf`, в котором прописана ссылка на стандартный драйвер Windows, после этого устройство появится в системе как COM-порт.

Коммуникационный протокол

Настройки коммуникационного порта

Настройки порта можно оставить такими, какими их определила ОС по умолчанию. При работе с библиотеками, которые требуют при открытии порта указать параметры, можно указать 9600/8/n/1, устройство работает в любыми настройками порта.

Форматы посылок

<DLE><STX><LEN><PAYLOAD><CS><DLE><ETX>

или

<ACK>

или

<NAK>

управляющие символы

<DLE> 0x10

<STX> 0x02

<ETX> 0x03

<ACK> 0x06

<NAK> 0x15

информационные поля

LEN - размер PAYLOAD в байтах

PAYLOAD - тело кадра

CS - контрольная сумма, сумма байтов PAYLOAD по модулю 2

<ACK> - используется для подтверждения приема команды

<NAK> – возвращается если получена неверная команда, неверная контрольная сумма, истекло время ожидания для получения следующего байта, при ошибке обработки команды(например невозможность записать в EEPROM).

Интервал между байтами должен быть не более 150мс. Если в поле PAYLOAD встречается символ DLE, то он должен быть заменен на пару DLE, DLE.

Например, если в кадре

<DLE><STX><3><BYTE1><BYTE2><BYTE3><CS><DLE><ETX>

BYTE2 равняется 0x10 , то отправляемый кадр должен быть сформирован следующим образом:

<DLE><STX><3><BYTE1><DLE><BYTE2><BYTE3><CS><DLE><ETX>

Формат PAYLOAD

<CMD_ID>[<ATTR>]

CMD_ID - код команды

ATTR - атрибуты, формат зависит от типа команды.

Общие команды

чтение версии программы

запрос: <0x00>

ответ: <0x00><ID><VER>

ID - 2 байта 'I', 'O'

VER - 2 байта, 0x0200 — версия 2.00

Команды и события для работы с кнопками

чтение состояний всех кнопок/ключей.

запрос: <0x10>

ответ: <0x10><STATES>

STATES - 4 байта, 32 битный массив состояния кнопок/ключей. 0 – кнопка отжата (ключ не повернут), 1 – кнопка нажата (ключ повернут).

чтение состояния кнопки/ключа.

запрос: <0x11><ID>

ответ: <0x11><ID><STATE>

ID - 1 байт, индекс кнопки/ключа.

STATE - 1 байт, состояние кнопки/ключа. 0 – кнопка отжата (ключ не повернут), 0x80 – кнопка нажата (ключ повернут).

разрешить/запретить автоматическую посылку событий

запрос: <0x18><MASK>

ответа: <ACK>

MASK - 4 байта, маска кнопок/ключей. Каждый бит маскирует событие соответствующей кнопки/ключа. 0 – событие замаскировано (запрещено). 1 – событие размаскировано (разрешено). По умолчанию события от всех кнопок/ключей замаскированы (0x0000000ul).

чтение маски разрешения событий

запрос: <0x1C>

ответ: <0x1C><MASK>

MASK - 4 байта, маска кнопок/ключей. По умолчанию события от всех кнопок/ключей запрещены, маска равна 0x00000000.

событие нажатия кнопки (поворачивания ключа).

Посылка команды инициируется контроллером при нажатии/отпускании кнопки замыкания/размыкании ключа, если в течении 500мс хост не отправит АСК на посылку контроллера, контроллер повторит отправку события. Если хост инициирует другую команду, она будет обработана контроллером, по завершении обработки хост повторит отправку события. Если событие произошло во время обработки команды от хоста, хост отправит событие сразу после обработки.

запрос контроллера:<0x12><ID><STATE>

ответ хоста :<АСК>

ID - индекс кнопки/ключа

STATE - 1 байт состояния кнопки/ключа. 0 – кнопка отжата (ключ не повернут), 0x80 – кнопка нажата (ключ повернут).

Если приходит ID — 0xFF и STATE — 0xFF, это означает, что имело место переполнение внутренней очереди событий (очередь не вычитывалась слишком долго). После переполнения очереди необходимо считать всю очередь и перечитать состояние всех кнопок, поскольку часть событий о нажатии и отпускании была потеряна.

Команды по управления лампами

включение/выключение ламп.

запрос: <0x28><STATES>

ответ: <АСК>

STATES - 4 байта, 32-битный массив состояния ламп. 0 – выключить лампу, 1 –включить лампу.

управление свечением лампы.

запрос: <0x29><ID><MODE>

ответ: <АСК>

ID - 1 байт, индекс лампы.

MODE - 2 байта, режим работы лампы, где 16 бит от младшего к старшему соответствуют времени включения лампы в 1/8 долях секунды с повторением через 2 секунды.

пример значений для поля MODE

0xFF00 – моргать, секунду вкл, секунды выкл

0xF0F0 – моргать, 1/2 секунды вкл, 1/2 секунды выкл

для двух ламп 0xF0F0 0x0F0F – моргать в противофазе

для четырех ламп 0xF000 0x0F00 0x00F0 0x000F – свет движется вдоль четырех ламп

ЧТЕНИЕ СОСТОЯНИЙ ВСЕХ ЛАМП.

запрос: <0x20>

ответ: <0x20><STATES>

STATES - 4 байта, 32 – битный массив состояния ламп. 0 – лампа выключена, 1 – лампа включена.

КОМАНДА ЧТЕНИЯ СОСТОЯНИЯ ЛАМПЫ.

запрос: <0x21><ID>

ответ: <0x21><ID><MODE>

ID - 1 байт, индекс лампы.

MODE - 16 бит состояние лампы(ниже)

Команды устройства EEPROM

Устройство EEPROM условно адресуется идентификатором от 0 до 31. Команды чтения/записи позволяют устанавливать в соответствие идентификатору несколько байт данных(теоретически возможно сохранение 253байтной строки).

Внутренне память EEPROM организована из 64 блоков. Каждая запись разбивается на блоки, и занимает N блоков, где $N = (size_in_bytes+5)/7$, округленное в большую сторону. Это следует учитывать, чтобы не переполнить хранилище в EEPROM.

Следует учитывать, что в хранилище реализован механизм, который позволяет при сбое питания во время записи нового значения в EEPROM не удалять старое. Таким образом после перезагрузки гарантировано доступно последнее удачно сохраненное значение. Этот механизм не позволит повторно записать в хранилище данные размером более 31блоков, перезапись будет возможна только в том случае, если сначала записать для параметра промежуточное значение нулевой длины (удалить его из хранилища).

ЧТЕНИЕ ПАМЯТИ

запрос: <0x30><ID>

ответ: <0x30><ID><DATA>

id - 1 байт, идентификатор строки от 0 до 31

data - сохраненная строка

Когда устройство отправляет считанные данные в порт, межбайтовый таймаут не превышает 300мс.

ЗАПИСЬ В ПАМЯТЬ

id - 1 байт, идентификатор строки от 0 до 31

data - строка

запрос: <38><ID><DATA>

ответ: <ACK> или <NAK>

id - 1 байт, идентификатор строки от 8 до 31
data - строка

Идентификаторы в диапазоне 0-7 предназначены для записи информации производителем.
При нехватке места в EEPROM возвращается NAK.

Если при записи в микросхему памяти произошла ошибка возвращается NAK,
целесообразно при получении NAK повторять процедуру до трех раз, после можно считать, что
в памяти недостаточно место, или микросхема памяти вышла из строя.

Скорость записи не более 40мс на один байт, ответ выдается после физической записи.

Примеры пакетов

Записать в параметр 1, "00"(0x30 0x30)

запрос 10 02 04 38 01 30 30 39 10 03

ответ 06

читать параметр 1

запрос 10 02 02 30 01 31 10 03

ответ 10 02 04 30 01 30 30 31 10 03

Включить все лампы

запрос 10 02 05 28 FF FF FF FF 28 10 03

ответ 06

Коды кнопок и ламп

0	кнопка/лампа	Line 1
1	кнопка/лампа	Line 3
2	кнопка/лампа	Line 5
3	кнопка/лампа	Line 7
4	кнопка/лампа	Line 9
8	кнопка/лампа	Start
9	кнопка/лампа	Bet
10	кнопка/лампа	Auto
11	кнопка/лампа	Info
12	кнопка/лампа	Menu
13	кнопка	Cash Out
14	кнопка/лампа	Change
16	ключ	Main Door
17	ключ	Door
18	ключ	Admin
19	ключ	Short Book
20	лампа	TOP (верхняя секция маячка)
21	лампа	BOT (нижняя секция маячка)

Приложение. Внешний вид платы контроллера ввода-вывода.

