

Протокол обмена данными

Условные обозначения которые встретятся ниже:

< > – значит один байт данных.

[] – необязательные поля.

Сокращение LCM означает Liquid Crystal Module - Жидкокристаллический модуль.

DWIN-овцы любят называть свои дисплеи "DWIN smart LCM"

Числа в кадрах указаны в шестнадцатеричном формате.

То есть, если в примере кадра написано 64, имеется в виду 0x64.

В контексте протокола DWIN, термины frame-кадр и packet-пакет - имеют одно и то-же значение.

Хочу напомнить что

байт имеет сокращение только большой/заглавной буквой Б(рус)/В(англ):

маленькой/строчной буквой б(рус)/b(англ) сокращается **бит**.

Пример:

1 КБ/KB = 1 Kbyte = 1 килобайт = 8 Кб/кб/Kb/kb = 8 kbit = 8 килобит.

Справа от названий байтов могут указываться буквы H и L.

Это обозначения старшего-High и младшего-Low байта в едином слове(word) данных.

Слово-word - тип данных соответствующий 2 байтам = 16 битам.

Машинные слова по 2 байта часто используются в структуре кадров.

Структура кадра данных

Имеет следующий формат:

<Frame Header H> <Frame Header L> <Byte Count> <Command> [<Data>] [<CRC H> <CRC L>]

Сокращенная запись:

<FHH> <FHL> <BC> <CMD> [<DATA>] [<CRCH> <CRCL>]

- **Frame Header/FH - заголовок кадра:** определяет начало кадра. Может использоваться для уникальной идентификации нескольких LCM на шина связи.

Значение по умолчанию = 0x5AA5.

- **Byte Count/BC - количество байт:** количество байтов в кадре,! исключая заголовок кадра(2байта) и этот байт.

То есть количество байт, начинающиеся с командного байта.

Итого реальная длина кадра: ByteCount + 3 байта.

- **CMD - Команда:** определяет команду, которая должна быть выполнена.

- **Data - Данные:** полезная нагрузка(payload) кадра, состоит из: адреса, длины и значений.

- **CRC-16:** хеш из 2 байт для обнаружения повреждения целостности кадров. Активируется через CFG файл.

CMD - команда:

Протокол DWIN поддерживает 5 команд:

- **0x80:** Запись в регистры управления
- **0x81:** Чтение регистров управления
- **0x82:** Запись в RAM
- **0x83:** Чтение RAM
- **0x84:** Запись графиков/кривых-curve (ОЗУ)

Примечание:

Актуальная документация на команды регистров(0x80, 0x81) и графиков(х84) мной не найдена.

Предполагаю что в протоколе DGUSII идёт тенденция к уменьшению числа команд до двух:

0x82(запись) и 0x83(чтение)

Команда записи (0x82)

Команда записывает один или несколько байт.

Фактически мы указываем начальный адрес памяти и записываемые байты, а количество записываемых байт дисплей рассчитывает исходя из указанной длины кадра - ByteCount. Нивелируя CRC и прочие не data байты.

Это значит что допустимо записывать больше или меньше байт чем указано в даташите.

~~Но так как нам доподлинно не известен алгоритм/логика работы с фотой ОС - лучше так не делать.~~

Пример

Есть кадр функции изменения яркости экрана в ждущем/активном режимах и указанием времени до начала ждущего режима.

Выглядит так 5A A5 07 82 00 82 64 32 03 E8

Но если у нас в .CFG файле не активирован ждущий режим, зачем тогда записывать в дисплей параметры ждущего режима(яркость и время)?

Можно уменьшить длину кадра на 3 байта: 5A A5 05 82 00 82 64

Рационально ли утруждаться по этому поводу - решать вам.

А что будет если записать больше байт чем указано?

Странный вопрос. Дисплей запишет и не ругнется всё что отправим, но новые байты попадут на другую область памяти и что-то может сломаться(обязательно сломается).

Команда чтения (0x83)

Команда читает один или несколько слов по указанному адресу RAM.

То есть минимальный объём чтения - 1 слово - 2 байта.

Фактически мы указываем начальный адрес чтения и количество читаемых слов. А значит допустимо

читать больше или меньше чем указано в даташите.

Пример

Есть кадр функции чтения (Ver)версии софта дисплея: 5A A5 04 83 00 0F **01**

Где крайняя цифра **01** - количество читаемых слов.

Увеличим количество читаемых слов с 1 до 6: 5A A5 04 83 00 0F **06**

В таком случае произойдет дополнительное чтение областей памяти следующих за Ver: RTC и PIN_now.

Классно! Можно за один запрос прочесть кучу параметров! Однако ответ не удобно парсить(разбирать на составляющие)...

Кстати также можно делать в обратном порядке.

Например хотим мы получить только год из RTC дисплея.

И вместо запроса: 5A A5 04 83 00 10 **04**

Делаем запрос: 5A A5 04 83 00 10 **01**

Data - данные:

В примерах/даташите описываются так: ...D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0

С указанием назначения каждого байта.

Пример: D2 D1 D0

D2: 0x05 - параметр1 - байт;

D1:D0: 0x1234 - параметр2 - 2 байта(слово);

Итоговый кадр будет выглядеть так

5A A5 04 82 **05** 12 34

5A A5 - заголовок кадра,

Где:

04 - число байт кадра с командного байта,

82 - байт команды,

05 - байт данных D2,

12 - байт данных D1,

34 - байт данных D0.

CRC контроль целостности данных

Платформа DGUSII для проверки целостности данных использует CRC-16 Modbus.

Включается проверка CRC через .CFG файл.

Функция включения/отключения по serial отсутствует.

Ниже приведён пример кода вычисления CRC-16 Modbus на Си.

```
// расчёт MODBUS RTU CRC
UInt16 ModRTU_CRC(byte[] buf, int len )
{
    UInt16 crc = 0xffff;
    for (int pos = 0; pos < len; pos++)
    {
        // XOR byte into least sig. byte of crc
        crc ^= (UInt16)buf[pos];
        for (int i = 8; i != 0; i--) // Цикл по каждому биту
        {
            if ((crc & 0x0001) != 0) // Если LSB установлен
            {
                crc >>= 1; // сдвиг вправо и XOR 0xA001
                crc ^= 0xA001;
            }
            else // В противном случае LSB не установлен
                crc >>= 1; // Просто сдвиньте вправо
        }
    }
    // Поменять местами старшие и младшие байты
    crc = ((crc & 0xff00) >> 8) | ((crc & 0x00ff) << 8);
    return crc;
}
```

Запросы к RAM, flash, переменным VP/SP и др.

Команда записи в RAM (0x82)

Команда записывает один или несколько слов.

Можно записать сразу несколько слов, если они последовательные.

Формат данных:

<FHH> <FHL> <BC> **82** <VPH><VPL> <VL1><VL1> [<VL2><VL2> <VL3><VL3> ...]

<VPH><VPL> Указатель на адрес RAM память(5000, 5002...).

<VL#><VL#> Значение записи

Ответ от LCM:

<FHH> <FHL> <BC> 82 <4F> <4B>

Пример.

Запишем значение 1234 в переменную VP 0x**5000**

5A A5 05 82 **50 00** 04 D2

5000: адрес переменной VP

04D2: DEC значение 1234 в HEX представлении.

Запишем значение 0x1234 в переменную VP 0x**5000**

5A A5 05 82 **50 00** 12 34

5000: адрес переменной VP

1234 значение 0x1234 .

Ответ от LCM в обоих случаях:

5A A5 03 82 4F 4B

Команда чтения из RAM (0x83)

Команда читает один или несколько слов.

Можно читать сразу несколько слов, если они последовательные.

Формат данных:

<FHH> <FHL> <BC> 83 <VPH><VPL> <LEN>

<VPH><VPL> Указатель на адрес RAM память(5000, 5002...).

<LEN> Количество считываемых слов/word(2 byte) (HEX).

Ответ от LCM:

<FHH> <FHL> <BC> 83 <VPH><VPL> <LEN> <VL1><VL1> [<VL2><VL2> <VL3><VL3> ...]

Пример.

Прочитаем переменную VP **0x5000**

5A A5 04 83 **50 00** 01

5000: адрес переменной VP

01: количество слов/word(2 byte) (HEX).

Ответ от LCM:

5A A5 0683 **50 00** 01 00 02

0002: значение переменной по адресу 5000

Важно

Отдельно контролировать длину отсылаемого кадра(3ий байт).

Наиболее частые ошибки из-за неверно указанной длины пакета.

Функции/инструкции/операции OS DGUS

Формат изложения:

Заголовок

Адрес RAM/Определение по datasheet-у / Доступные действия R/W(читать/писать) /Длина(word~2 байта)

Описание

Перезагрузка дисплея

0x04 System_Reset W 2

Функция 55 AA 5A A5 сбрасывает питание ЦП T5L один раз.

Это эквивалентно перезагрузке с отключением питания.

Пример: 5A A5 07 82 00 04 55 AA 5A A5

Обновление прошивки ОС

0x06 OS_Update_CMD W 2

Адреса:

D3: 0x5A активировать однократное выполнение инструкции.

D2: Тип файла:

0x10: код ОС DWIN /T5*_OS, должен начинаться с 0x1000, каждый раз обновляет 28 КБ.

0xA5: код 8051 /T5*_UI, каждый раз обновляет 64 КБ.

D1:D0: Начальный адрес RAM по которому расположено обновление. Должен быть четным.

Копировать из RAM в flash и обратно

0x08 NOR_FLASH_RW_CMD W 4

Адреса:

D7: Режим: 0x5A=чтение flash, 0xA5=запись flash.

D6:4: Начальный адрес Nor Flash. Должен быть чётный.

0x000000-0x03:FFFE, 256KWords.

D3:2: Начальный адрес RAM(пользовательских переменных). Должен быть чётный.

D1:0: Количество копируемых слов. Должен быть чётный.

Примечание

Операция чтения/записи может занять некоторое время.

Если вы посылаете несколько uart команд подряд.

То перед отправкой очередной команды, убедитесь что HMI модуль закончил операцию записи/чтения.

Узнать это можно отправив команду чтения: 5A A5 04 83 00 08 04

Если в ответе D7 очищен(0x00) = предыдущая операция завершена.

Если адрес D7 не очищен(≠0x00), то вновь входящие операции не будут выполнены.

В противном случае произойдет логическая неразбериха.

Если последовательный порт используется - другие инструкции должны быть отправлены только после соответствующей

Эта инструкция используется для загрузки и обновления os.bin и ui.bin.

Используется в основном в ПО для ПК "DWIN OS Build" - имеет встроенную функцию загрузки файла os.bin через последовательный порт-2.

1. Сначала поместим данные в RAM память.

Например запишем данные(0x12345678) в пользовательскую переменную по адресу RAM **0x1002** выглядит так:

5A A5 07 82 **10 02** 12 34 56 78

2. Теперь можем из RAM переместить данные в flash.

Записываем из RAM в flash:

5A A5 0B 82 0008 **A5** 000002 1002 0002

A5 - режим записи flash

000002 - начальный адрес в flash

1002 - начальный адрес в RAM

0002 - количество копируемых слов.

3. Скопируем обратно из flash, но уже по новому адресу RAM 0x2000:

5A A5 0B 82 00 08 **5A** 00 00 02 20 00 00 02

5A - режим чтения flash

000002 - начальный адрес в flash

2000 - начальный адрес в RAM

0002 - количество копируемых слов.

4. Проверим: 5A A5 04 83 10 02 02

Узнать версию ПО в дисплее

0x0F Ver R 1

Адреса:

D1 означает версию ui/8051,

D0 означает версию ОС DWIN.

задержки.

Идеи применения:

1. Проверить стабильность работы дисплея и соединения.

Своеобразный ring пакет :))

Может помочь, например, чтобы понять, что дисплею не хватает мощности источника питания.

Пропинговать дисплей в течении 1 секунды после старта.

Если пакеты потерялись, то что-то тут не так.

Кстати, если мощности источника питания не хватает, то можно понизить яркость подсветки. Например мне помогло это при питании дисплея от слаботочного USB выхода ноута при отладке.

2. В процессе отладки может быть важна версия системы.

3. А что если экран разобьют и потом заменят на новый?

Ведь ПО контроллера(ардуино) разрабатывается при одной версии ОС. И вдруг поставят такой-же дисплей, но с устаревшей версией ОС. Тогда есть вероятность, что некоторые команды/инструкции могут быть не реализованы “и привет-пока”. В общем желательно версию ОС контролировать. Чтобы старая не попала. А в новых версиях думаю не станут вносить изменения в уже реализованные функции/инструкции.

4. Передача и ответ этой инструкции часто используются для проверки нормальности связи между последовательным портом компьютера и последовательным экраном.

Например:

Запрос: 5A A5 04 83 00 0F 01

Возвращает: 5A A5 06 83 000F 01 1110

0x11 — версия программного обеспечения GUI,

0x10 — версия программного обеспечения DWIN OS

Работа с RTC/часами дисплея

0x10 RTC R/W 4

Если у внешнего контроллера(ардуино) нет RTC, то можно использовать данные с RTC дисплея.

Адреса:

D7=год (0-0x63)

D6=месяц(0-0x0C),

D5=день(0-0x1F),

D4=неделя(0-0x6),

D3=час(0-0x17),

D2=минута(0-0x3B),

D1=секунда(0-0x3B),

D0 не определено.

Например:

Запись в RTC: 5A A5 0B 82 00 10 13 0A 01 00 0B 0C 0D 00

Выведет на Текстовый дисплей RTC: 2019-10-01 11:12:13 SUN,

Первые две цифры года определяются автоматически;

Неделя отображается на английском языке.

Чтение RTC: 5A A5 04 83 00 10 04

Ответ от предыдущей записи:

5A A5 0C 83 00 10 04 13 0A 01 00 0B 0C 0D 00

Ответ если нету RTC:

5A A5 0C 83 00 10 04 00 00 00 06 00 00 A5 00

Получить идентификатор текущей страницы/картинки

0x14 PIC_Now R 1

Обратная связь по рабочему состоянию осуществляется в миллисекундах, что обычно применяется в особых случаях. Пользователь может судить, занято ли ядро GUI программой DWIN-OS.

На практике:

Фактически случайно выдаёт 0/1 - не ясно от чего это зависит. Т.к. экран в это время может спокойно лежать.

Фиксирует крайнее событие взаимодействия с тачскрином. Позволяет определить тип и координаты касания тачскрина.

Идеи применения:

Возвращает ID/N текущей активной страницы.
Только чтение.

Пример:

Запрос: 5A A5 04 83 00 14 01

Ответ: 5A A5 06 83 00 14 01 0007 (0007 — страница 07)

Обратная связь о статусе GUI

0x15 GUI_Status R 1

Значения:

0x0000=свободно,

0x0001=идёт обработка 13.bin и 14.bin.

События тачскрина

0x16 TP_Status R/W 4

Адреса:

D7: При касании в D7 устанавливается значение 0x5A.

Если при чтении в D7 = 0x5A значит было касание.

Чтобы отловить новое касание:

нужно записать в D7 любое значение ≠0x5A, а затем считывать/отслеживать когда оно снова станет 0x5A.

D6: типы касаний.

0x00 = касания не зафиксированы,

0x01 = первичное касание пальца,

0x02 = поднять палец=было нажатие,

0x03 = держать палец на экране,

D5:D4=координата X

D3:D2=координата Y

D1:D0=0x0000.

Узнать текущей уровень яркости подсветки

1. Для отслеживания активности пользователя.
2. При нажатии в определённых областях можно активировать скрытый функционал - раньше так делали в платёжных терминалах.

Пример:

Запрос: 5A A5 04 83 00 16 03

Ответ: 5A A5 0A 83 00 16 03 5A **02** 02 55 00 E5

02 - короткое касание

0x0255 ~ 597 - координата X

0x00E5 ~ 229 - координата Y

Сбросить метку касаний D7, а заодно и тип касания - D6.

Запись: 5A A5 05 82 00 16 00 00

Получить текущее значение АЦП на AD0-AD7

0x32 AD0-AD7 Instantaneous value R 8

Позволяет получить значение АЦП с портов AD0-AD7

Горизонтальное разрешение экрана

0x7A LCD_HOR R 1

Исходя из значений указанных в .CFG файле при прошивке.

Пример:

Чтение: 5A A5 04 83 00 7B 01

0x31 LED_Now R 1

D1: 0x5A значение яркости подсветки. AD0-AD7 значение мгновенного обновления. Всегда возвращает 0x5A ?что это?

D0: Текущее значение яркости подсветки, 0x00-0x64.(0-100%)

Пример:

Запрос: 5A A5 04 83 003101

Ответ: 5A A5 06 83 00 31 01 5A **64**

0x64 ~ 100 %

Идея(плохая):

Можно косвенно отслеживать активен экран или нет. Правильно делать это через события тачскрина.

Мгновенное значение АЦП AD0-AD7, 2байта/AD. 1 слово на AD.
Напряжение = AD_значение * 3300/4095 мВ.

Запрос: 5A A5 04 83 00 32 08

Ответ:

5A A5 14 83 0032 08 **0FF9** 0FFC 0EF8 0EE8 00F0 00F5 012A 01DE
4089 4092 3832 3816 240 245 298 478 <ADx_value
3295 3298 3088 3075 193 197 240 385 <voltage_mV

Пример:

Чтение: 5A A5 04 83 00 7A 01

Ответ: 5A A5 06 83 00 7A 01 01 E0

0x01E0 ~ 480px — разрешение по горизонтали.

Вертикальное разрешение экрана

0x7B LCD_VER R 1

Исходя из значений указанных в .CFG файле при прошивке.

Параметры файла .CFG

Ответ: 5A A5 06 83 00 7B 01 01 10

0x0110 ~ 272px — разрешение по вертикали.

Позволяет узнать некоторые параметры .CFG файла.

А также изменить небольшую часть этих параметров.

Но изменения происходят только в RAM памяти.

То есть после перезагрузки дисплея изменения сбросятся на установленные файлом.CFG.

Большая часть байтов: D3:D2:D1 - только для чтения.

D0 соответствует адресу 0x05 файла .CFG и может записать некоторые биты: 0-4

Пример:

Следующие две инструкции вкл/откл зуммер экрана при касаниях.

Запись: 5A A5 07 82 00 80 5A 00 00 30 - выкл

Запись: 5A A5 07 82 00 80 5A 00 00 38 - вкл

0x80 System_Config R/W 2

Адреса:

D3: 0x5A = активировать однократное выполнение инструкции.

D2: значение конфигурации чувствительности сенсорной панели.

Только чтение.

D1: значение конфигурации режима сенсорной панели. Только чтение.

D0: настройка состояния системы.

7: Проверка CRC последовательного порта 0=выкл. 1=вкл.
Чтение.

6: Зарезервировано, напишите 0.

5: Включить загрузку файла 22, чтобы инициализировать пространство переменных. 1= вкл, 0= выкл. Чтение.

4: Параметр автоматического обновления/загрузки 1=вкл., 0=выкл., Чтение/запись.

3: Зуммер дисплея. 1=вкл. 0= выкл., Чтение/запись.

2: Режим ожидания. 1=вкл. 0=выкл, Чтение/запись.

1:0: Поворот отображения. 00 = 0°, 01 = 90°, 10 = 180°, 11 = 270°, Чтение/запись.

Установить яркость экрана

0x82 LED_Config R/W 2

Адреса:

D3 = уровень яркости в активном режиме, 0x00-0x64;

D2 = уровень яркости в режиме ожидания, 0x00-0x64;

D1:0=время бодрствования, ед.изм.10мс. N*10 мс. - время через которое включится режим ожидания. Если режим ожидания включен в файле .cfg.

Примечание:

В режиме ожидания. При касании экрана, происходит переход в активный режим.

При этом кнопки и другие элементы дизайна не реагируют на

Запрос чтения: 5A A5 04 83 00 80 02

Ответ: 5A A5 08 83 00 80 02 00 14 10 38

Ответ: 5A A5 08 83 00 80 02 00 14 70 3C

Яркость экрана в активном/ждущем режимах и длительность бодрствования(до перехода в ждущий режим).

Пример:

Запись: 5A A5 07 82 0082 64 32 03 E8, -

0x64 ~ 100% - яркость в активном режиме,

0x32 ~ 50% - яркость в ждущем режиме,

03E8 ~ 1000*10мс = через 10 секунд включится режим ожидания.

Запись: 5A A5 07 82 0082 2020 03 E8, - яркость в обоих режимах одинаковая 0x20 ~ 32%.

Если переход в режим ожидания отключён, то кадр изменения яркости можно укоротить, запись: 5A A5 05 82 00 82 0A

Что будет если установить ноль?

Подсветка совсем выключится. При этом сама матрица продолжит работать. Можно подсвечивать фонариком :))

Если установить уровень яркости 1, то чуть-чуть видны импульсы шима. Думаю такой эффект только на некоторых моделях.

касание. То есть происходит только одно событие - выход из режима ожидания(включение активного режима).

В случае если яркость обоих режимов одинаковая, то визуально будет казаться что режим ожидания не включается.

Однако когда он включится, не удивляйтесь что кнопки не реагируют на касание - сначала происходит переход в активный режим.

В общем чтобы такого “бага-фитчи” не было, надо изначально решить нужен ли переход экрана в ждущий режим. Если не нужен, то отключить его в .CFG.

Установить ШИМ на PWM0_Set

0x86 PWM0_Set R/W 2

Позволяет создать шим на выводе PWM0

Адреса:

D3=0x5A - активировать однократное выполнение инструкции.

D2= коэффициент частотного деления

D1:D0=PWM0 precision - коэффициент деления частоты

Несущая частота PWM0 = 825,7536 МГц / (коэффициент деления частоты PWM0 precision).

Пример:

Чтение: 5A A5 04 83 00 86 02

Ответ: 5A A5 04 83 00 86 02 00 00 00 00

1. Конфигурация на 100 кГц.

Коэффициент деления установлен на 1,

верхнее предельное значение = 825,7536 МГц/0,1 МГц = 8258 (0x2042).

Запишите данные 5A 01 20 42 в переменную 0x0086, выходное разрешение 13 бит, несущая ШИМ 100 кГц.

Запрос: 5A A5 07 82 00 86 5A 01 20 42

2. Запишите переменную 0x0092 для управления выходным коэффициентом заполнения PWM0.

Записываемое значение = коэффициент высокого уровня * значение верхнего предела

Например, 10%, записываемое значение равно $10\% * 8258 = 826$ (0x033A).

Запрос: 5A A5 07 82 00 86 5A 00 00 00

Установить время аппаратных RTC часов

0x9C RTC_Set W 4

Должна быть аппаратная поддержка RTC.

Адреса:

D7:D6= 0x5AA5 - активировать однократное выполнение инструкции.

D5:D0=год, месяц, день, час, минута, секунда, все в шестнадцатеричном формате.

009D: старший байт - год, младший байт - месяц,

009E: старший байт - день, младший байт - час,

009F: старший байт - минуты, младший байт - секунды.

Получить текущий PWM0_Out

0x92 PWM0_Out R/W 1

Позволяет получить текущее значение выхода PWM0

D1:D0=Ширина высокого уровня вывода PWM0, точность 0x0000-PWM0.

Пример:

Чтение: 5A A5 04 83 00 92 01

Ответ: 5A A5 04 83 00 92 01 00 00

Способы установить время RTC:

1: запись: 5A A5 0B 82 00 9C 5A A5 12 06 1B 15 15 15

12 - год и далее месяц, день, час, минута, секунда.

2: ввод через клавиатуру интерфейса:

1. Return key code: VP= 0x009C, key value: 0x5AA5.

2. Data Input: VP=009D, Variable type=High Byte - установка года.

3. Data Input: VP=009D, Variable type=Low Byte - установка месяца.

4. Data Input: VP=009E, Variable type=High Byte - установка день.

5. Data Input: VP=009E, Variable type=Low Byte - установка час.

6. Data Input: VP=009F, Variable type=High Byte - установка минут.

7. Data Input: VP=009F, Variable type=Low Byte - установка секунд.

Воспроизведение музыки из файла .WAE

0xA0 WAE Music Play_Set R/W 2

Адреса:

D3: ID/N мелодии из .wae, 0x00-0xFF.

D2: Количество сегментов, фиксированное 0x01. Сбрасывается после операции. В режиме зуммера это время зуммера, ед. измерения: 8 мс.

D1: 0x00-0x40. Громкость, ед. измер. 1/64. Начальное значение 0x40(~64) = 100%.

D0: Обратная связь. Текущий статус воспроизведения: 0x00=стоп,

0x01=пауза, 0x02=воспроизведение. Только чтение. На запись не реагирует.

Работа с внешней флеш памятью, загрузка/обновление файлов

0xAA External 16Mbytes FLASH write operation R/W 6

Адреса:

D11: 0x5A - активировать однократное выполнение инструкции.

D10: Режим работы:

Для D10=0x02, запись блока 32 КБ в флеш память.

Копирует из RAM в FLASH.

D9:D8: Адрес flash для записи блока 32 КБ. В диапазоне адресов 0x0000-0x01FF. 32KB*0x1FF ~ соответствует 16 МБ памяти.

D7:D6: Адрес RAM памяти с начала которого копируется блок.

Файл .WAE должен быть предварительно записан во встроенную флеш память, также как .ICL. Расположение файла .WAE определяется файлом конфигурации .CFG.

Настроим воспроизведения музыки из .WAE:

1. В файле .CFG(по адресу 0x07) укажите адрес памяти файла .WAE(первое число в названии файла).

Пример:

26.wae - записать 0x1A(~26) в 0x07 .CFG

40.wae - записать 0x28(~40) в 0x07 .CFG

2. В файле .CFG(по адресу 0x05) укажите источник воспроизведения:

В 6 бит: 0=зуммер, 1=динамик/музыка.

3. Начать воспроизведение музыки.

Пример запроса:

5A A5 07 82 00 A0 03 01 40 00

03 - ID/N мелодии в файле .wae.

01 - сегмент, default 1.

40 - громкость 100%.

00 - байт обратной связи - любое число, а можно и не записывать.

Инструкция позволяет читать/писать 16 МБ содержимого внешней SPI NOR FLASH памяти блоками по 32 КБ.

Память объемом 16 МБ можно рассматривать:

как пространство 512*32 КБ для обновления данных или

как пространство для 64-ёх шрифтов с фиксированной емкостью каждого 256 КБ.

Позволяет обновить через serial port-2: файл фоновых изображений.ICL, файл иконок, файл шрифтов и др.

Пример:

Инструкция 00B0 может рассматриваться как предоставление

Должно быть чётным.

D5:D4: Время ожидания следующей инструкции записи

ед.изм.1мс.

После завершения этой операции ОС DGUSII не будет читать flash память указанное количество времени. Это нужно чтобы предотвратить ошибки. Возникающие при неполной загрузке обновления.

20мс(0x0014)-100мс должно быть достаточно. Но зависит от задержек вашего внешнего контроллера.

При передаче последней инструкции записи это значение можно указать 0x00 - чтобы ОС мгновенно начала читать данные из памяти. Это не обязательно.

D3:D0: Не определено, запишите 0x00.

Для D10=0x10, чтение из FLASH в RAM.

D9: ID/N библиотеки шрифтов, 0x10-0x1F(~16-31), 256 КБ на библиотеку шрифтов, максимум 4 МБ.

D8:D6: Адрес начала данных в библиотеке шрифтов.

Определяется в соответствии с Word. 0x0000-0x01FFFF.

D5:D4: Адрес RAM в начало которого произойдет копирование. Четное.

D3:D2: Длина считываемых данных(word/2 байт). Четное.

D1:D0: не определено, записать 0x00.

Для D10=0x03, запись шрифтов из RAM в FLASH.

D9:D8: ID/N исходного шрифта, 0x0000-0x001F(~0-31), 256 КБ на библиотеку шрифтов, максимум 4 МБ.

D7:D6: ID/N целевого шрифта, 0x0000-0x003F(~0-63).

D5-D4: Номер/N копируемого шрифта, 0x0000-0x0010(~0-16).

D3:D0: Не определено, запишите 0x00.

Пример:

5AA5 0B 82 00B0 5AA5 0001 0905 0000

закрывать файл 09-го касания на странице 0001.

05 представляет собой код команды, возвращаемый значением

обновления для шрифта с ID 22 .

Запрос: 5A A5 0F 82 00 AA 5A 02 00 B0 80 00 00 14 00 00 00 00

Запрос1 чтения в RAM: 5A A5 0F 82 00 AA 5A 01 01 00 00 80 00 00 FF 00 00

Запрос2 чтения в RAM: 5A A5 0F 82 00 AA 5A 01 00 00 B0 80 00 FF FF 00 00

Запрос чтения RAM в COM-порт: 5A A5 04 83 80 00 FF

Доступ к инструкциям тачскрина

0xB0 Touch instruction access interface W 36

Адреса:

0xB0: 0x5AA5 - активировать однократное выполнение инструкции.

0xB1: ID страницы сенсорного управления(touch control).

0xB2: старший байт: идентификатор сенсорного управления (устанавливается в ПО для разработки DGUS II), 0x01-0xFF; Младший байт: код сенсорного управления, 0x00-0x7F.

0xB3: режим доступа

0xB4-0xD3: данные для изменения режима 0x02, 0x03.

Режим 0x0000: отключить сенсорное управление /сенсорные команды.

Режим 0x0001: включить это сенсорное управление(touch control).

Режим 0x0002: прочитать это сенсорное управление и записать его в SRAM по адресу 0xB4.

Режим 0x0003: обновить текущее сенсорное управление данными, на которые указывает 0xB4, формат и длина данных должны быть одинаковыми.

Эмуляция касаний сенсора/тачскрина

0xD4 TP operation simulation W 5

сенсорной клавиши (значение кода сенсорной команды см. в списке функций сенсорной переменной),
0000 представляет собой отключение сенсорного управления.

Примечание:

Основное управление тачскрином не может быть включено/выключено. Если он оценивается в ПО ПК, его не нужно считать эффективным сенсорным номером.

Если вам нужно изменить определенные свойства сенсорного экрана, вам нужно использовать режим 03, чтобы заменить предварительно записанный сенсорный файл 0xB4 в конфигурации свойств файла 13bin. Найдите ключевые слова 0xB0 на форуме DWIN с подробным описанием сообщения.

Адреса:

D4: 0x5AA5 - активировать однократное выполнение инструкции.

D5: режим - тип события нажатия.

0x0001=нажать/первичное касание пальца,

0x0002=поднять палец - освободить экран,

0x0003=удерживать нажатие,

0x0004=комплекс: нажать-отпустить - типичное короткое нажатие.

D6: координата X положения нажатия.

D7: координата Y положения нажатия.

Запускает операцию имитации нажатий на сенсорный экран.

При симуляции касаний звуки как обычно воспроизводятся(если опция звуковых сигналов при касаниях включена в .cfg)

Симуляция касаний выполнит переход дисплея из режима ожидания в активный режим(при активированном режиме ожидания в .cfg)

После имитации режима 0x0001/нажать и 0x0003/удержать необходимо имитировать режим 0x0002/поднять.

Пример:

5A A5 0B 82 00 D4 5A A5 00 04 00 EE 00 8F

0004 — нажать + поднять,

в координатах 00EE 008F (283 143),

Ещё пример: 5A A5 0B 82 00 D4 5A A5 00 04 02 F8 00 1E

?..далее не понятно...

Когда симуляция запущена(сборка функции касаний запущена),
x= 0xAA:KH y=0xA5:KL будут напрямую возвращать значения
ключей KH и KL в сборке касания.

Например, когда переменные вводятся в режим ввода/сбора,
координаты (0xAAF0, 0xA5F0) приведут к немедленному
завершению ввода/сбора.

Когда файл касаний 13, разработан с функцией keying, координата X = 0xFF: код клавиши. Y координата = 0x0001 вызовет соответствующую функцию keying.

Иконка должна быть загружена в файл .ICL.

Иконка фиксируется постоянно до перезагрузки. Если сменить экран и вернуться обратно, то иконка не исчезнет. Чтобы убрать иконку нужно отправить команду 00 в D7 или отобразить иконку в пустой позиции.

Иконки всегда отображаются в режиме фильтрации фона, а интенсивность фоновой фильтрации фиксирована на 0x08

ID/N иконки может быть выбран пользователем для достижения эффектов отображения “click” и “wait”.

Отобразить иконку на дисплее

0xD8 Pointer icon overlay display ? 4

Адреса:

D7: 0x5A - включить отображение иконки

D6: номер/ID файла .ICL в котором находится иконка

D5:D4: ID/N иконки в файле .ICL

D3:D2: координата X

D1:D0: координата Y

Пример:

5A A5 0B 82 00D8 5A **2D** 00 01 00 64 00 64

Отобразит иконку 1 из файла **45**.icl, в координатах XY: 100x100.

CRC проверка данных в flash памяти

0xE0 Memory CRC check R/W 2

Адреса:

D3: 0x5A - активировать однократное выполнение инструкции.

D2: выбор типа памяти

0x00 = пространство шрифтов (16 МБ)

0x02 = код ОС DWIN

0x03 = пространство Nor Flash (файлы библиотек/LIB).

D1:D0: интерфейс данных.

Интерфейсы данных:

для проверки шрифтов:

D1 = стартовый ID шрифта (256 КБ для каждого шрифта);

D0 = количество проверяемых блоков по 4 КБ, 0x00-0xFF.

для проверки кода ОС:

D1:D0: начиная с 0x1000,

длина проверяемого кода ОС составляет 0x0001-0x7000 в байтах.

для проверки данных в флэш-памяти:

D1:D0: ID базы данных флэш-памяти, фиксированная проверка данных 4 КБ каждый раз.

Используется для обнаружения ошибок/поврежденных данных в flash памяти.

После проверки возвращает значение CRC.

Данная инструкция позволяет воспроизводить музыкальные файлы онлайн(из RAM памяти).

Данные представляют собой 16-битный целочисленный формат.

Пример:

Воспроизведение музыки из буфера по адресу 0x**8000**, размер буфера 1K.

5A A5 0B 82 00F0 5A 02 00 00 **8000** 0400

0x**8000** - адрес буфера.

0x**0400** ~1024 - размер буфера 1K.

Воспроизведения музыкального потока онлайн

0xF0 Interface of playing music flow data W 4

Адреса:

D7: 0x5A - активировать однократное выполнение инструкции.

D6: режим-стадия,

0x00 = стоп(очистить буфер),

0x01 = пауза/приостановить (зарезервировать буфер),

0x02 = воспроизвести/плей.

D5:D4: не определено, записать 0x00.

D3:D2: адрес RAM с массивом данных музыкального потока(0x1000 - 0xFFFF). Четный.

D1:D0: размер/длина массива данных (слов). Максимум 8KWords. Чётный;

Рисование касаниями сенсора

0xF4 Painting interface W 8

Адреса:

D15: 0x5A - включить сенсорное окно рисования.

D14: режим работы,

0x00 - нормальный режим,

0x01 - инициализация (чистая инициализация).

D13: конфигурация параметров.

7-2 зарезервировано, напишите 0.

1-0 режим обработки буфера рисования после смены страницы.

0x00 - закрыть,

0x01 - повторная инициализация,

остальные = без изменений.

D12: зарезервировано, запишите 0x00.

D11: ширина линии пера, от 0x01 до 0x0F.

D10:D8: цвет окраски, D6=КРАСНЫЙ D5=ЗЕЛЕНый D4=СИНИЙ

D7:D4: координата положения верхнего левого угла окна (x,y).

D3:D2: ширина окна в пикселях, должна быть кратна 4.

D1:D0: высота окна в пикселях, должна быть кратна 4.

Включить высокоскоростную загрузку UART1

0xFE UART1 high speed download ? 2

0x5AA5 - включить высокоскоростную загрузку через UART1

Рисует линию при касаниях.

Можно использовать, например, для ввода подписи человека.

RAM занимаемая рисованием сенсорного окна, составляет 128 КБ (диапазон адресов двойного слова 0x00:8000-0x00: FFFF), что соответствует 208*208 пикселям.

Пример:

Эта команда используется после прикосновения к экрану.

Отображение дорожки координат в обозначенной области экрана, которая обычно используется в «функции рукописного рисования».

0x00 режим=нормальный режим

1. Отправить команду окна рисования на страницу:

5A A5 13 82 00 F4 5A 00 00 00 02 FF 00 00 00 2C 00 BC 00 D0 00 D0

2. Ручная сенсорная панель для рисования графики

3. Переключить страницу

4. Вернитесь на предыдущую страницу, снова отправьте команду окна рисования, и символы снова отобразятся. В верхнем левом углу окна (44 188) точка координат соответствует самому большому окну рисования размером 208*208 пикселей.

Рисование кривых/графиков

0x300-0x37F Dynamic curve interface R/W 128

Адреса:

0x300-0x30F: обратная связь по состоянию для 8-канальных буферов кривых (рекомендуется только чтение),

2 слова на канал,

старшее слово — это место указателя памяти (0x0000-0x07FF), где хранятся данные кривой,

а младшее слово — эффективная длина данных кривой буфер (0x0000-0x0800).

Запись 0x0000 в эффективную длину данных буфера кривой приведет к невозможности отображения кривой.

0x310-0x311: начать запись данных буфера кривой

D3:D2: 0x5AA5 включает операцию записи данных буфера кривой один раз и очищает после операции.

D1: количество блоков данных, 0x01-0x08.

D0: не определено, запишите 0x00.

0x312-0x37F: блок данных, записанный в буфер кривой, который составляет 16 бит без знака.

Отдельный блок данных: идентификатор канала данных (0x00-0x07) + длина слова данных (0x01-0x6E) + данные.

При включенном отображении динамической кривой начните с 0x1000 и создайте буфер данных для каждой кривой в соответствии с 2Kwords на канал.

Буфер CH0 — 0x1000-0x17FF,

буфер CH1 — 0x1800-0x1FFF и т. д.

Неиспользуемые зоны буфера кривой могут использоваться как пользовательские переменные. Также пользователи могут напрямую перезаписывать данные буфера кривой, а затем изменять 0x300-0x30F соответствующую позицию указателя памяти и длину данных, чтобы обеспечить правильное отображение кривой.

Позволяет получить адрес RAM(переменной) и количество слов данных. Которые были отправлены из дисплея в COM/serial порт крайний раз.

То есть выходные данные этой инструкции меняются при отправке переменных из дисплея.(например по событию Data auto-uploading, synchrodata, etc.)

Пример:

Запрос: 5A A5 04 83 0F 00 02

Ответ: 5A A5 08 83 0F 00 02 00 1100 01 означает, что адрес

Индикатор отправленных данных/ переменной

0x0F00 Variable change indication R 2

Адреса:

D3:5A: означает изменение переменной,

D2:D1: указатель на ячейку RAM(адрес переменной),

D0: размер переменной(word).

переменной 0x1100 загрузил 1 слово данных.

Запрос: 5A A5 04 83 0F 00 02

Ответ: 5A A5 08 83 0F 00 02 00 1000 02 означает, что адрес переменной 0x1000 загрузил 2 слова данных.

Wi-Fi и другой интерфейс управления приложениями для интернет-коммуникационного оборудования.

Интерфейс мультимедийных приложений,
цифровой мультимедийный интерфейс 0x500-0x57F,
аналоговый мультимедийный интерфейс 0x580-0x5BF

Интерфейс внешней памяти (например, диск U) для чтения или записи

Wi-Fi интерфейс управления

0x400-0x4FF [Network communication interface](#) R/W 256

Интерфейс мультимедийных приложений

0x500-0x5BF [Multi-Media interface](#) R/W 192

Интерфейс внешней памяти /USB

0x5C0-0x5FF [External memory interface](#) R/W 64

Wi-Fi и интернет в дисплее DWIN

Компания DWIN разработала Wi-Fi модуль «**WiFi-10**» и облачную платформу «**DWIN Cloud**» для приложений **AIoT**.

Модуль WiFi-10 специально разработан для дисплеев серии DGUS.

В настоящее время платформа **DWIN Cloud** имеет поддержку только в Китае.

Дисплеи **DWIN DGUSII** (включая серию CPU T5 и серию CPU T5L) имеют сетевой интерфейс работающий через Wi-Fi модуль. Их можно подключить к **DWIN Cloud** с помощью простой среды разработки DGUS. На основе облачной **DWIN Cloud** клиенты могут реализовать удаленное управление приложениями, анализ данных, работу оборудования и другие функции.

Краткое техническое описание:

Модуль **WiFi-10** построен на базе **ESP8266**.

WiFi-10 активно считывает 0x490 и другие регистры из **DWIN OS** каждые 0,1-0,2 секунды.

WiFi-10 каждые 3-5 секунд записывает статус WiFi-10 в регистр DGUS (статус модуля WiFi: 0x480, 0x4A0 и т.д.).

Модуль можно настроить через сенсорный экран или мобильное приложение.

Основные настройки это указание адреса вашего wi-fi роутера и пароль.

После завершения настройки, WiFi-10 автоматически подключится к **DWIN Cloud** через интернет вашего роутера и начнет выполнять синхронизацию данных.