



Версия драйвера: 0.9
Версия документации: 0.9
Дата последнего изменения: 18 марта 2008

«Штрих-М: Драйвер кросс - платы 17'»

Руководство программиста

НТЦ «Штрих-М» 2008

Содержание

Содержание	2
Комплект поставки.....	4
Подключение	4
Структура драйвера	4
Методы	4
FindDevice	4
Initialize	4
Reset	5
GetDeviceType	5
GetDriverVersion	5
GetPortSettings	5
SetPortSettings	5
UploadFirmware	5
GetDeviceMode	6
GetState.....	6
ButtonsControl	6
SetDeviceMode	6
GetTableStruct.....	6
GetFieldStruct.....	7
TableInit.....	7
GetFieldValue	7
ReadMemory	7
GetProgMemorySize	7
GetDataMemorySize	7
GetEEPROMMemorySize.....	7
SendCommonMessage	8
SetFieldValue	8
Uninitialize	8
ShowPropertyDlg.....	8
ClearQueue	8
ResetWatchDog.....	8
GetDeviceErrorDescription	9
GetDriverErrorDescription.....	9
PISetRelayState	9
PIImpulseRelayState	9
PIGetRelayState	10
События	10
OnError	10
OnResponseReceived	10
OnGetDeviceMode	11
OnCommandSent	11
OnGetFieldValue	11
OnGetMemorySize	11
OnGetTableStruct.....	11
OnGetFieldStruct.....	12
OnTableNotDefined	12
OnGetRelayState.....	12

Комплект поставки

CrossShtrih17Drv.dll – драйвер для операционной системы Windows

CrossShtrih17Test.exe – тест драйвера

Примеры – исходные коды драйвера и теста

Сервисная инструкция Cross_Shtrih_17_Rev2.4_v0.3(070609).pdf – описание протокола Cross_Shtrih_17

Сервисная инструкция Power_Interrupter_Rev1.1_v0.3(070302).pdf - описание протокола Power_Interrupter

ДрайверCross17.pdf - эта документация

Подключение

Имя объекта драйвера: *CrossShtrih17Drv.CrossShtrih17*

Структура драйвера

Все вызовы методов драйвера помещают подготовленные команды в очередь сообщений (LIFO).

После записи последовательности байт команды в последовательный порт возбуждается событие *OnCommandSent*

Если по каким-либо причинам не удастся послать команду устройству, или устройство не отвечает, тогда инкрементируется счетчик команды, и повторяется попытка отправки команды. Максимальное число таких попыток – 10, при превышении этого числа возбуждается событие *OnError*.

При получении ответа от устройства, происходит возбуждение соответствующего специализированного события, после чего возбуждается событие *OnResponseReceived*. Если специализированного события для данной команды не найдено, возбуждается только событие *OnResponseReceived*.

Методы

FindDevice

Описание:

Отображает диалог поиска устройства. Поиск ведется по выбранным пользователем последовательным портам.

Параметры:

portNum – выходной параметр, если устройство найдено и нажата кнопка «Выбрать», содержит номер последовательного порта.

baudRate - выходной параметр, если устройство найдено и нажата кнопка «Выбрать», содержит текущую скорость обмена данными поддерживаемую устройством.

Initialize

Описание:

Инициализирует соединение по заданному порту и скорости, устанавливает таймауты приема байта и команды.

Параметры:

portNumber – номер порта
baudRate - скорость обмена данными
timeoutByte – таймаут приема байта
timeoutCmd – таймаут приема команды

Reset

Описание:

Выполняет операцию сброса.

GetDeviceType

Описание:

Выполняет операцию получения типа устройства.

GetDriverVersion

Описание:

Возвращает строку содержащую версию драйвера.

Параметры:

version – выходной параметр, содержит строку – версию драйвера

GetPortSettings

Описание:

Возвращает Параметры: соединения, Параметры: возвращаются через событие *OnResponseReceived*

SetPortSettings

Описание:

Устанавливает Параметры: соединения.

Параметры:

baudRate – новая скорость обмена сообщениями
timeout – таймаут приема команды

UploadFirmware

Описание:

Закачивает новую прошивку.

Параметры:

filename – путь к файлу с прошивкой
changeMode – автоматически меняет режим работы (с рабочего режима на режим программирования и обратно по завершению операции). Если

задано false необходимо вручную менять режим до и после операции, вызывая метод *SetDeviceMode*

GetDeviceMode

Описание:

Запрос текущего режима работы. Информация о текущем режиме возвращается через событие *OnGetDeviceMode*

GetState

Описание:

Запрос состояния кросс платы. Информация о текущем режиме возвращается через событие *OnResponseReceived*

ButtonsControl

Описание:

Управление кнопками на материнской плате. Устанавливает состояние кнопок.

Параметры:

btnState – состояние кнопок:

- 0 бит: Резерв
- 1 бит: нажатие кнопки RESET
- 2 бит: нажатие кнопки POWER
- 3 бит: нажатие кнопки POWER длинное
- 4 бит: Отключение питания с задержкой
- 5 бит: Отмена отключения питания с задержкой
- 6 бит: Резерв
- 7 бит: Резерв

SetDeviceMode

Описание:

Устанавливает режим работы устройства.

Параметры:

mode – режим работы:

- 12H – нормальный режим работы
- 10H – режим программирования

GetTableStruct

Описание:

Запрос структуры таблицы. Информация о структуре таблицы возвращается через событие *OnGetTableStruct*

Параметры:

num – номер запрашиваемой таблицы.

GetFieldStruct

Описание:

Запрос структуры поля. Информация о структуре поля возвращается через событие *OnGetFieldStruct*

Параметры:

tableIndex – номер запрашиваемой таблицы.
fieldIndex – номер запрашиваемого поля.

TableInit

Описание:

Инициализация таблиц. Установление исходных значений таблиц

GetFieldValue

Описание:

Запрос текущего значения поля. Текущее значение поля возвращается через событие *GetFieldValue*

Параметры:

tableIndex – номер запрашиваемой таблицы
fieldIndex – номер запрашиваемого поля
rowIndex - номер запрашиваемой строки

ReadMemory

Описание:

Чтение памяти кросс платы и запись в файл

Параметры:

memType – тип считываемой памяти
filename – путь с именем файла

GetProgMemorySize

Описание:

Запрос размера дампа памяти программ. Запрашиваемая информация возвращается через событие *OnGetMemorySize*

GetDataMemorySize

Описание:

Запрос размера дампа памяти данных. Запрашиваемая информация возвращается через событие *OnGetMemorySize*

GetEEPROMMemorySize

Описание:

Запрос размера дампа памяти EEPROM. Запрашиваемая информация возвращается через событие *OnGetMemorySize*

SendCommonMessage

Описание:

Помещения в очередь заданной команды.

Параметры:

mes – массив байт команды

completeMessage – Если установлено в true, сообщение будет дополнено байтами контрольной суммы

SetFieldValue

Описание:

Устанавливает значение поля

Параметры:

tableIndex – номер таблицы

fieldIndex – номер поля

rowIndex – номер строки

value – значение

Uninitialize

Описание:

Закрывает соединение с кросс платой

ShowPropertyDlg

Описание:

Отображает диалог параметров соединения.

Параметры:

port – выходной параметр, выбранный пользователем номер порта

baudRate – выходной параметр, скорость передачи данных

timeoutByte – выходной параметр, таймаут приема байта

timeoutCmd – выходной параметр, таймаут приема команды

ClearQueue

Описание:

Очищает очередь сообщений

ResetWatchDog

Описание:

Осуществляет сброс датчика WatchDog. Кросс плата сбрасывает датчик всякий раз, когда просиходит получение команды, поэтому сброс датчика реализован с помощью отправки команды запроса состояния *GetState*

GetDeviceErrorDescription

Описание:

Возвращает словесное описание ошибки специфичной для данного устройства

Параметры:

errorCode – код ошибки
description – выходной параметр, строка с описанием ошибки

GetDriverErrorDescription

Описание:

Возвращает словесное описание ошибки специфичной для данного драйвера

Параметры:

errorCode – код ошибки
description – выходной параметр, строка с описанием ошибки

PISetRelayState

Описание:

Устанавливает состояние реле.

Параметры:

stateFlags – флаги состояния:

- 0 бит: Реле 1
- 1 бит: Реле 2
- 2 бит: Реле 3
- 3 бит: Реле 4
- 4 бит: Не используется (всегда 0)
- 5 бит: Не используется (всегда 0)
- 6 бит: Не используется (всегда 0)
- 7 бит: Не используется (всегда 0)

Примечание:

Данный метод доступен только в платах типа *Power_Interrupter*

PIImpulseRelayState

Описание:

Импульс состояния реле

Параметры:

stateFlags – флаги состояния:

- 0 бит: Реле 1
- 1 бит: Реле 2
- 2 бит: Реле 3
- 3 бит: Реле 4

- 4 бит: Не используется (всегда 0)
 - 5 бит: Не используется (всегда 0)
 - 6 бит: Не используется (всегда 0)
 - 7 бит: Не используется (всегда 0)
- Time – время импульса

Примечание:

Данный метод доступен только в платах типа *Power_Interrupter*

PIGetRelayState

Описание:

Запрос состояния реле. Информация о состоянии реле возвращается через событие *OnGetRelayState*

Примечание:

Данный метод доступен только в платах типа *Power_Interrupter*

СОБЫТИЯ

OnError

Описание:

Произошла ошибка.

Параметры:

resultCode – код ошибки драйвера

win32ErrorCode – код ошибки Win32

errorType – тип ошибки:

- 0 – нет ошибки
- 1 – ошибка драйвера (resultCode – код ошибки, для получения словесного описания ошибки используйте метод *GetDriverErrorDescription*)
- 2 – ошибка Win32 (win32ErrorCode – код ошибки, используйте функции Win32 API для получения словесного описания ошибки)
- 4 – ошибка устройства (resultCode – код ошибки, для получения словесного описания ошибки используйте метод *GetDeviceErrorDescription*)

OnResponseReceived

Описание:

Получен ответ от устройства

Параметры:

cmd - код команды, на которую был получен ответ

ary – массив байт ответа

OnGetDeviceMode

Описание:

Получен режим работы устройства

Параметры:

error - код ошибки устройства (0 если операция завершена успешно)

mode – режим работы

purpose – причина изменения режима работы

OnCommandSent

Описание:

Команда отправлена. Возбуждение этого сообщения не означает что команда получена устройством

Параметры:

arg – массив байт отправленной команды

OnGetFieldValue

Описание:

Получено значение поля таблицы

Параметры:

tableIndex – номер таблицы

fieldIndex – номер поля

rowIndex – номер строки таблицы

value - значение

OnGetMemorySize

Описание:

Получен размер дампа памяти

Параметры:

size – размер дампа памяти

memType – тип памяти:

- 30h – память данных
- 31h – память программ
- 32h - память EEPROM

OnGetTableStruct

Описание:

Получена структура таблицы

Параметры:

tableIndex – номер таблицы

name – название таблицы

fieldsNum – количество полей

rowsNum – количество строк

OnGetFieldStruct

Описание:

Получена структура поля таблицы

Параметры:

tableIndex – номер таблицы
fieldIndex – номер поля
name – название поля
type – тип поля
len – длина поля
min – минимальное значение
max – максимальное значение

OnTableNotDefined

Описание:

Ошибка. Произошло обращение к таблице, которая не определена

Параметры:

tableIndex – номер таблицы

OnGetRelayState

Описание:

Получено состояние реле

Параметры:

cmd – код команды
relay1 – состояние реле1
relay2 – состояние реле2
relay3 – состояние реле3
relay4 – состояние реле4