

УДК 512.6, 517.9, 519.6

УДАЛЕННЫЕ МОДУЛИ ИНДИКАЦИИ НА БАЗЕ ЖИДКОКРИСТАЛЛИЧЕСКИХ ИНДИКАТОРОВ

Богословский Дмитрий Николаевич¹, Буряков Михаил Григорьевич²

¹С.н.с. ЛФВЭ ОИЯИ, старший преподаватель кафедры персональной электроники;
ГБОУ ВПО Международный Университет природы, общества и человека «Дубна»;
Институт системного анализа и управления;
141980, Московская обл., г. Дубна, ул. Университетская, 19;
Объединенный институт ядерных исследований;
141980, Московская обл., г. Дубна, ул. Жолио-Кюри, 6;
e-mail: bogoslovski@mail.ru.

²Студент;
ГБОУ ВПО Международный университет природы, общества и человека «Дубна»;
Институт системного анализа и управления;
141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: mikhail466@gmail.com.

В результате выполнения работы разработана библиотека функций для работы с LCD-модулями на базе контроллеров HD44780/KS0066 со специализированным параллельным интерфейсом, а также спроектирован модуль конвертора интерфейсов «UART (RS – 232/RS – 422/RS – 485) – параллельный интерфейс LCD-модуля» на базе микроконтроллера C8051F410 от фирмы Silicon Laboratories. Библиотека функций позволяет использовать все типы LCD-индикаторов с вышеупомянутым параллельным интерфейсом. Модуль конвертора интерфейсов в свою очередь позволяет использовать LCD-индикатор со стандартными последовательными интерфейсами RS-232/RS-422/RS-485. Таким образом, LCD-индикаторы с модулем конвертора интерфейсов можно применять как самостоятельный мобильный блок индикации. Применение: различное промышленное оборудование; транспорт; информационные модули в офисных, торговых помещениях и т.д. Создана подробная документация временных диаграмм различных режимов работы параллельных интерфейсов ЖК-модуля, блок-схемы с подробными комментариями и программное обеспечение.

Ключевые слова: ЖКИ, удаленные, модули.

REMOTE INDICATION MODULE LCD DISPLAY BASED

Bogoslovskiy Dmitriy¹, Buryakov Mikhail²

¹S.R. LPHE JINR, Senior teacher at the department of Personal Electronics;
Dubna International University of Nature, Society and Man,
Institute of system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
Join institute for nuclear research;
141980, Moscow reg., Dubna, Joliot-Curie, 6;
e-mail: bogoslovski@mail.ru.

²Student;
Dubna International University of Nature, Society and Man;
Institute of system analysis and management;
141980, Dubna, Moscow reg., Universitetskaya str., 19;
e-mail: mikhail466@gmail.com.

The main result of this work is development of a library of function for LCD-modules based on HD44780/KS0066 controllers with specialized parallel interface as well as design of interfaces converter module «UART (RS-232/RS-422/RS-485) – LCD-module parallel interface» based on the C8051F410 microcontroller, which was created by The Silicon Laboratories company. The library of functions allows to use all types of LCD-indicators with the above-mentioned parallel interface. Interfaces converter module, in turn, allows to use LCD – indicators with standard serial interfaces RS – 232/RS – 422/RS – 48. Thus,

LCD – indicators combined with interfaces converter module can be used as a standalone mobile display unit. Possible applications include: industrial equipment, vehicles, information modules in offices and stores, etc. Detailed documentation including timing charts of different modes of LCD – module’s parallel interfaces and flowchart with comprehensive descriptions as well as software was created.

Keywords: Remote, LCD, modules.

Введение

Жидкокристаллические (ЖК) модули являются одним из основных средств вывода информации для современных цифровых систем. Представляют собой недорогое и удобное решение, позволяющие сэкономить время и ресурсы при работе новых изделий. Обеспечивают отображение большого объема информации при хорошей различимости и низком электропотреблении, благодаря чему широко используются в измерительных приборах, медицинском оборудовании, промышленном оборудовании, информационных системах, аппаратуре с автономным питанием.

1. Исследование предметной области

Опишем принцип работы UART, интерфейсов RS – 232/RS – 422/RS – 485 и последовательный периферийный интерфейс.

1.1. Универсальный асинхронный приемопередатчик (UART)

Универсальный асинхронный приёмопередатчик (УАПП, англ. UART, Universal Asynchronous Receiver/Transmitter) – вид приёмопередатчика, устройство, которое переводит данные из последовательной в параллельную форму и обратно. Обычно используется вместе со стандартами связи, такими как EIA RS-232. В асинхронном режиме работы UART отличается от RS-232 только логическими уровнями (у первого они обычно соответствуют TTL уровням, у второго – равны +12В для логического нуля и – 12В для логической единицы).

UART представляет собой отдельное устройство или является частью интегральной схемы, используется для передачи данных через последовательный порт компьютера или периферийного устройства.

UART часто встраивают в микроконтроллеры. UART можно представить в виде двух функциональных блоков – приемник (Receiver) и передатчик (Transmitter). В состав UART входят:

- тактовый генератор связи (бодрейт-генератор) – задает тактовую частоту приемопередатчика для данной скорости связи;
- управляющие регистры – задают режим работы последовательного порта и его прерываний;
- статусные регистры – в них устанавливаются флаги по различным событиям;
- буферы – в буфер приемника попадает принятый символ, в буфер передатчика помещают передаваемый;
- сдвиговые регистры – приемника по биту накапливают принимаемые из порта биты, сдвиговый регистр передатчика являются обоймой, из которой в последовательный порт выстреливаются биты передаваемого символа (кадра).

По различным событиям устанавливаются флаги и генерируются прерывания (завершение приема/отправки кадра, освобождение буфера, различные ошибки).

UART – полнодуплексный интерфейс, то есть приемник и передатчик могут работать одновременно, независимо друг от друга. Порт приемника обозначают RX, передатчика – TX. Последовательной установкой уровней на этих портах относительно общего провода («земли») и передается информация. По умолчанию передатчик устанавливает на линии единичный уровень. Передача начинается посылкой бита с нулевым уровнем (старт-бита), затем идут биты данных младшим битом вперед (низкий уровень – «0», высокий уровень – «1»), завершается посылка передачей одного или двух битов с единичным уровнем (стоп-битов).

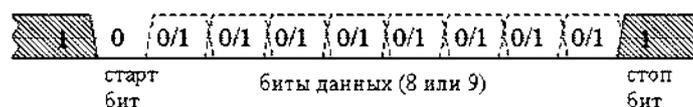


Рис. 1.1. Структура кадра посылки интерфейса UART

Перед началом связи между двумя устройствами необходимо настроить их приемопередатчики на одинаковую скорость связи и формат кадра.

Скорость связи или baudрейт (baudrate) измеряется в бодах – число передаваемых бит в секунду (включая старт и стоп-биты). Задается эта скорость в baudрейт-генераторе делением системной частоты на задаваемый коэффициент. Типичный диапазон скоростей: 2400 ... 115200 бод.

Формат кадра определяет число стоп-битов (1 или 2), число бит данных (8 или 9), а также назначение девятого бита данных. Все это зависит от типа контроллера.

Приемник и передатчик тактируются, как правило, с 16-кратной частотой относительно baudрейта. Это нужно для считывания сигнала. Приемник, поймав падающий фронт старт-бита, отсчитывает несколько тактов и следующие три такта считывает (семплирует) порт RX. Это как раз середина старт-бита. Если большинство значений семплов – «0», старт-бит считается состоявшимся, иначе приемник принимает его за шум и ждет следующего падающего фронта. После удачного определения старт-бита, приемник точно также семплирует серединки битов данных и по большинству семплов считает бит «0» или «1», записывая их в сдвиговый регистр. Стоп-биты тоже семплируются, и если уровень стоп-бита не «1» – UART определяет ошибку кадра и устанавливает соответствующий флаг в управляющем регистре.

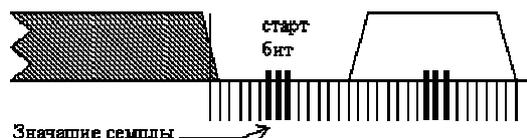


Рис. 1.2. Формирование старт бита в UART

Поскольку baudрейт устанавливается делением системной частоты, при переносе программы на устройство с другим кварцевым резонатором, необходимо изменить соответствующие настройки UART.

1.2. Последовательный периферийный интерфейс (шина SPI)

Serial Peripheral Interface, SPI bus – последовательный синхронный стандарт передачи данных в режиме полного дуплекса, разработанный для обеспечения сопряжения микроконтроллеров и периферии. SPI также иногда называют четырехпроводным (four-wire) интерфейсом.

Главным составным блоком интерфейса SPI является обычный сдвиговый регистр, сигналы синхронизации и ввода/вывода битового потока которого и образуют интерфейсные сигналы. Таким образом, протокол SPI является протоколом обмена данными между двумя сдвиговыми регистрами, каждый из которых одновременно выполняет и функцию приемника, и функцию передатчика. Непременным условием передачи данных по шине SPI является генерация сигнала синхронизации шины. Этот сигнал имеет право генерировать только ведущий шины и от этого сигнала полностью зависит работа подчиненной шины.

В отличие от стандартного последовательного порта, SPI является синхронным интерфейсом, в котором любая передача синхронизирована с общим тактовым сигналом, генерируемым ведущим устройством (процессором). Принимающая периферия (ведомая) синхронизирует получение битовой последовательности с тактовым сигналом. К одному последовательному периферийному интерфейсу ведущего устройства-микросхемы может присоединяться несколько микросхем. Ведущее устройство выбирает ведомое для передачи, активируя сигнал «выбор кристалла» (chip select) на ведомой микросхеме. Периферия, не выбранная процессором, не принимает участие в передаче по SPI.

Существует три типа подключения к шине SPI, в каждом из которых участвуют четыре сигнала:

- MOSI или SI – выход ведущего, вход ведомого (Master Out Slave In). Служит для передачи данных от ведущего устройства ведомому.
- MISO или SO – вход ведущего, выход ведомого (Master In Slave Out). Служит для передачи данных от ведомого устройства ведущему.
- SCLK или SCK – последовательный тактовый сигнал (Serial Clock). Служит для передачи тактового сигнала для ведомых устройств.
- CS или SS – выбор микросхемы, выбор ведомого (Chip Select, Slave Select).

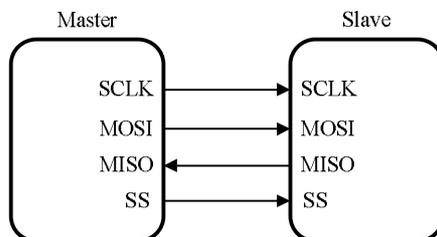


Рис. 1.3. Простейшее подключение к шине SPI

Подключение, в котором участвуют только две микросхемы, показано на рисунке (рис. 1.3). Здесь, ведущий шины передает данные по линии MOSI синхронно со сгенерированным им же сигналом SCLK, а подчиненный захватывает переданные биты данных по определенным фронтам принятого сигнала синхронизации. Одновременно с этим подчиненный отправляет свою посылку данных. Представленную схему можно упростить исключением линии MISO, если используемая подчиненная информационная система (ИС) не предусматривает ответную передачу данных или в ней нет потребности. Одностороннюю передачу данных можно встретить у блоков ЦАП, цифровых потенциометров, программируемых усилителей. Таким образом, рассматриваемый вариант подключения подчиненной ИС требует 3 или 4 линии связи, чтобы подчиненная ИС принимала и передавала данные, помимо наличия сигнала синхронизации, необходимо также, чтобы линия SS была переведена в низкое состояние. В противном случае, подчиненная ИС будет неактивна.

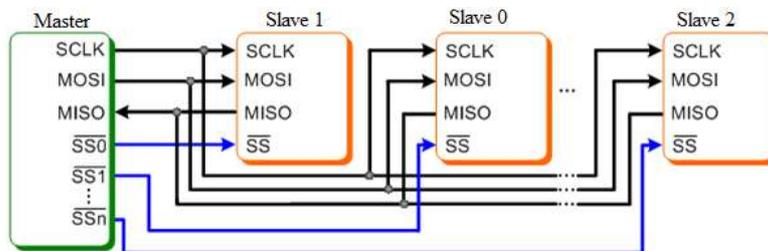


Рис. 1.4. Независимое подключение к шине SPI

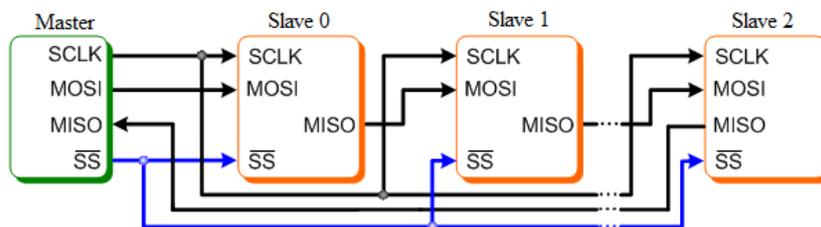


Рис. 1.5. Каскадное подключение к шине SPI

При необходимости подключения к шине SPI нескольких микросхем используется либо независимое (параллельное) подключение (рис. 1.4), либо каскадное (последовательное) (рис. 1.5). Независимое подключение более распространенное, т.к. достигается при использовании любых SPI-совместимых микросхем. Здесь, все сигналы, кроме выбора микросхем, соединены параллельно, а ведущий шины, переводом того или иного сигнала SS в низкое состояние, задает, с какой подчиненной ИС он будет обмениваться данными. Главным недостатком такого подключения является необходимость в дополнительных линиях для адресации подчиненных микросхем (общее число линий связи равно $3 + n$, где n – количество подчиненных микросхем). Каскадное включение избавлено от

этого недостатка, т.к. здесь из нескольких микросхем образуется один большой сдвиговый регистр. Для этого выход передачи данных одной ИС соединяется со входом приема данных другой, как показано на рисунке (рис. 1.3). Входы выбора микросхем здесь соединены параллельно и, таким образом, общее число линий связи сохранено равным 4. Однако использование каскадного подключения возможно только в том случае, если его поддержка указана в документации на используемые микросхемы. Чтобы выяснить это, важно знать, что такое подключение по-английски называется «daisy-chaining».

Возможно четыре варианта логики работы интерфейса SPI. Эти варианты получили название режимов SPI и описываются двумя параметрами:

- CPOL – исходный уровень сигнала синхронизации (если CPOL = 0, то линия синхронизации до начала цикла передачи и после его окончания имеет низкий уровень (т.е. первый фронт нарастающий, а последний – падающий), иначе, если CPOL = 1, – высокий (т.е. первый фронт падающий, а последний – нарастающий));
- CPHA – фаза синхронизации; от этого параметра зависит, в какой последовательности выполняется установка и выборка данных (если CPHA = 0, то по переднему фронту в цикле синхронизации будет выполняться выборка данных, а затем, по заднему фронту, – установка данных; если же CPHA = 1, то установка данных будет выполняться по переднему фронту в цикле синхронизации, а выборка – по заднему).

Таблица 1.1. Режимы SPI

Режим SPI	0	1	2	3
CPOL	0	1	0	1
CPHA	0	0	1	1
Временная диаграмма первого цикла синхронизации				

1.3. Интерфейс RS-232C

Последовательный интерфейс синхронной и асинхронной передачи данных, определяемый стандартом EIA RS-232-C. Интерфейс RS-232-C соединяет два устройства. Линия передачи первого устройства соединяется с линией приема второго и наоборот (полный дуплекс). Для управления соединенными устройствами используется программное подтверждение (введение в поток передаваемых данных соответствующих управляющих символов). Интерфейс RS-232C предназначен для подключения к компьютеру стандартных внешних устройств (принтера, сканера, модема, мыши и др.), а также для связи компьютеров между собой. Данные в RS-232C передаются в последовательном коде побайтно. Каждый байт обрамляется стартовым и стоповыми битами. Данные могут передаваться как в одну, так и в другую сторону (дуплексный режим).

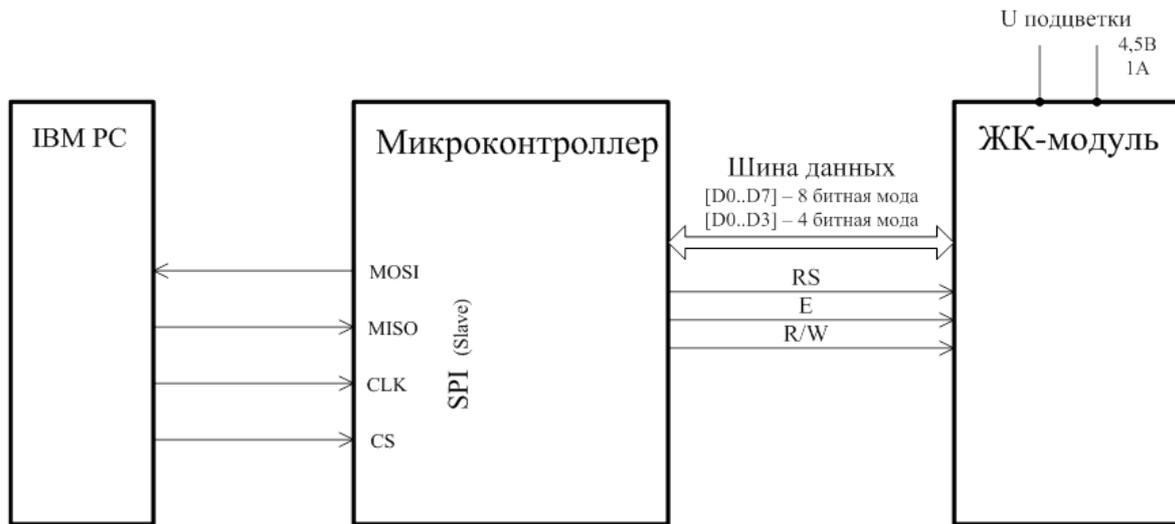


Рис. 1.6. Структурная схема подключения ЖК-модуля через SPI – параллельный интерфейс ЖК-модуля

Компьютер имеет 25-контактный (DB25P) или 9-контактный (DB9P) разъем для подключения RS-232C.

RS - 232 Connector

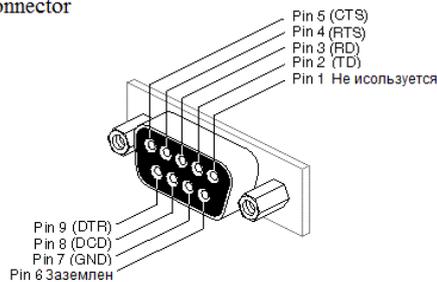


Рис. 1.7. RS-232 Connector

Таблица 1.2. Описание сигналов RS-232

№ контакта (Pin)	Обозначение	Назначение
Pin 1	Не используется	Сигнальное заземление, нулевой провод.
Pin 2	TD	Данные, передаваемые компьютером в последовательном коде (логика отрицательная).
Pin 3	RD	Данные, принимаемые компьютером в последовательном коде (логика отрицательная).
Pin 4	RTS	Сигнал запроса передачи. Активен во все время передачи.
Pin 5	CTS	Сигнал сброса (очистки) для передачи. Активен во все время передачи. Говорит о готовности приемника.
Pin 6	Заземлен	Сигнальное заземление, нулевой провод.

Pin 7	GND	Защитное заземление (экран).
Pin 8	DCD	Сигнал сброса (очистки) для передачи. Активен во все время передачи. Говорит о готовности приемника.
Pin 9	DTR	Готовность выходных данных.

Наиболее часто используются трех- или четырехпроводная связь (для двунаправленной передачи). Для двухпроводной линии связи в случае только передачи из компьютера во внешнее устройство используются сигнальное заземление, нулевой провод и TD. Все 10 сигналов интерфейса задействуются только при соединении компьютера с модемом.

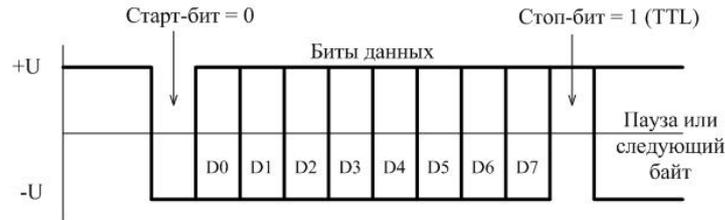


Рис. 1.8. Формат данных RS-232C

Формат передаваемых данных показан на рисунке 1.8. Собственно данные (5, 6, 7 или 8 бит) сопровождаются стартовым битом, битом четности и одним или двумя стоповыми битами. Получив стартовый бит, приемник выбирает из линии биты данных через определенные интервалы времени. Очень важно, чтобы тактовые частоты приемника и передатчика были одинаковыми, допустимое расхождение – не более 10%. Скорость передачи по RS-232C может выбираться из ряда: 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 бит/с.

Обмен по RS-232C осуществляется с помощью обращений по специально выделенным для этого портам COM1, COM2, COM3. Форматы обращений по этим адресам можно найти в многочисленных описаниях микросхем контроллеров последовательного обмена UART (Universal Asynchronous Receiver/Transmitter).

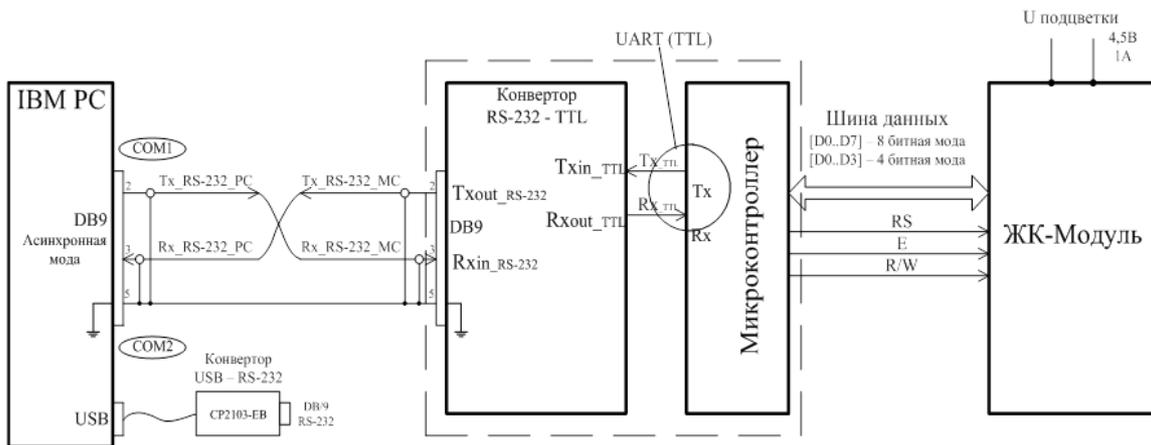


Рис. 1.9. Структурная схема подключения ЖК-модуля через RS-232 – параллельный интерфейс ЖК-модуля

1.4. Интерфейсы RS-485

Интерфейс RS-485 (другое название – EIA/TIA-485) – один из наиболее распространенных стандартов физического уровня связи. Физический уровень – это канал связи и способ передачи сигнала. В основе интерфейса RS-485 лежит принцип дифференциальной (балансной) передачи данных. Суть его заключается в передаче одного сигнала по двум проводам. Причем по одному проводу (условно

А) идет оригинальный сигнал, а по другому (условно В) – его инверсная копия. Другими словами, если на одном проводе «1», то на другом «0» и наоборот. RS-485 – полудуплексный интерфейс. В сети может быть много передатчиков, так как они могут отключаются в режиме приема. Аппаратная реализация интерфейса – микросхемы приемопередатчиков с дифференциальными входами/выходами (к линии) и цифровыми портами (к портам UART контроллера).

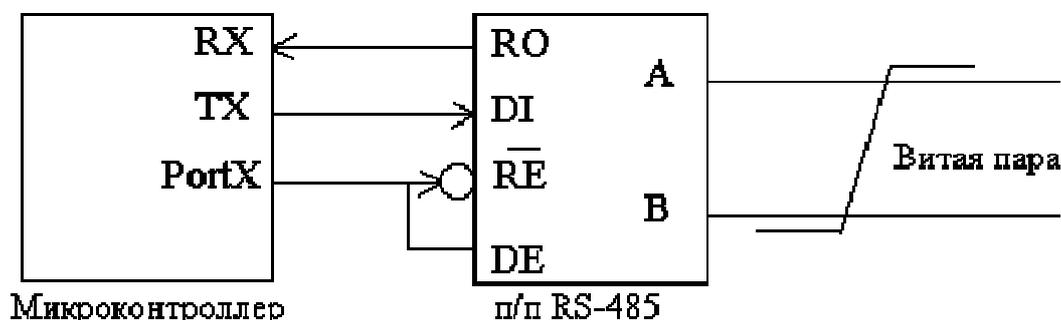


Рис. 1.10. Схема аппаратной реализации

Цифровой выход приемника (RO) подключается к порту приемника UART (RX). Цифровой вход передатчика (DI) к порту передатчика UART (TX). Поскольку на дифференциальной стороне приемник и передатчик соединены, то во время приема нужно отключать передатчик, а во время передачи – приемник. Для этого служат управляющие входы – разрешение приемника (RE) и разрешения передатчика (DE). Так как вход RE инверсный, то его можно соединить с DE и переключать приемник и передатчик одним сигналом с любого порта контроллера. При уровне «0» – работа на прием, при «1» – на передачу.

Приемник, получая на дифференциальных входах (AB) разность потенциалов (UAB) переводит их в цифровой сигнал на выходе RO. Чувствительность приемника может быть разной, но гарантированный пороговый диапазон распознавания сигнала производители микросхем приемопередатчиков пишут в документации. Обычно эти пороги составляют ± 200 мВ. То есть, когда $UAB > +200$ мВ – приемник определяет «1», когда $UAB < -200$ мВ – приемник определяет «0». Если разность потенциалов в линии настолько мала, что не выходит за пороговые значения – правильное распознавание сигнала не гарантируется.

Максимальная скорость связи по спецификации RS-485 может достигать 10 Мбод/сек. Максимальное расстояние – 1200 м. Если необходимо организовать связь на расстоянии большем 1200 м или подключить больше устройств, чем допускает нагрузочная способность передатчика – применяют специальные повторители (репитеры).

RS-422/RS - 485 Connector

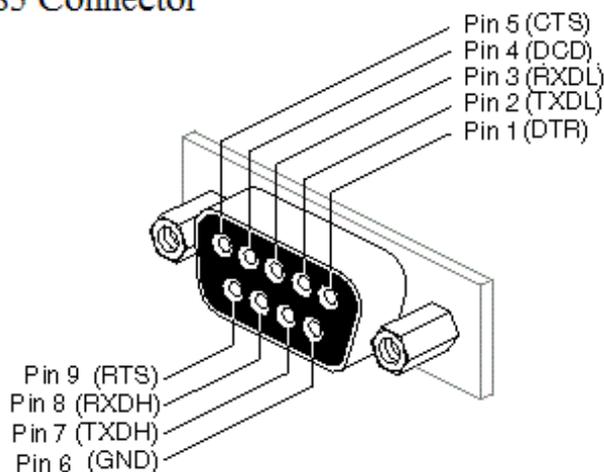


Рис. 1.11. RS-422/RS-485 Connector

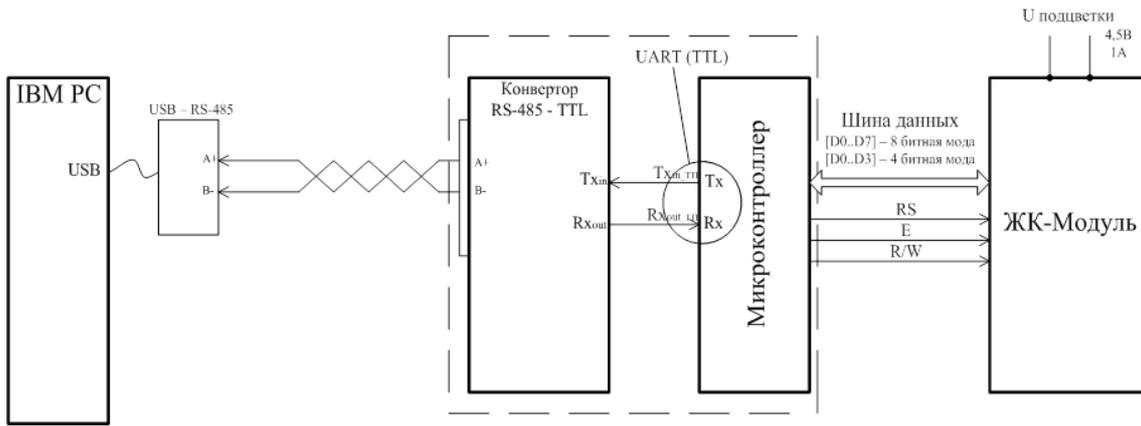


Рис. 1.12. Структурная схема подключения ЖК-модуля через RS-485 (полудуплекс) – параллельный интерфейс ЖК-модуля. RS-485 – опция в промышленных PC

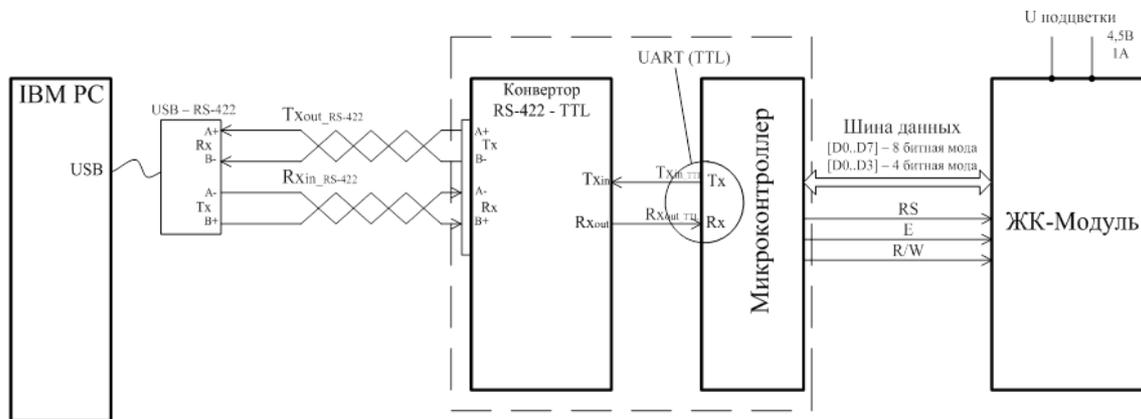


Рис. 1.13. Структурная схема подключения ЖК-модуля через RS-422 (дуплекс) – параллельный интерфейс ЖК-модуля. RS-422 – опция в промышленных PC

Таблица 1.3. Описание сигналов RS-422/RS485

№ контакта (Pin)	Обозначение	Назначение
Pin 1	DTR	Готовность выходных данных.
Pin 2	TXDL	Данные, передаваемые компьютером в последовательном коде (логика отрицательная).
Pin 3	RXDL	Данные, принимаемые компьютером в последовательном коде (логика отрицательная).
Pin 4	DCD	Сигнал сброса (очистки) для передачи. Активен во все время передачи. Говорит о готовности приемника.
Pin 5	CTS	Сигнал сброса (очистки) для передачи. Активен во все время передачи. Говорит о готовности приемника.
Pin 6	GND	Защитное заземление (экран).

Pin 7	TXDH	Данные, передаваемые компьютером в последовательном коде (логика положительная).
Pin 8	RXDH	Данные, принимаемые компьютером в последовательном коде (логика положительная).
Pin 9	RTS	Сигнал запроса передачи. Активен во все время передачи.

Таблица 1.4. Характеристика скорость обмена/длина линии связи (зависимость экспоненциальная)

62,5 кбит/с	1200 м
375 кбит/с	300 м
2400 кбит/с	100 м
10000 кбит/с	10 м

2. Микроконтроллер

2.1. Микроконтроллеры серии C8051Fxxx фирмы Silicon Laboratories

Микроконтроллеры фирмы Silicon Laboratories C8051Fxxx предназначены для построения устройств, требующих высокой производительности, большой степени интеграции, точности измерений и малого энергопотребления. Они программно совместимы с 8051-м стандартом, но одновременно имеют рекордно высокую производительность, до 100 MIPS и ультранизкое энергопотребление. Эти микроконтроллеры также уникальны своей интеграцией с аналоговыми узлами. Многие модели имеют несколько независимых АЦП (например, у модели C8051F060 имеется 16 разрядный АЦП с 2 входами и 10 разрядный АЦП с 8 входами), встроенный ЦАП, компараторы напряжения, программируемые усилители напряжения и др.

Микроконтроллеры семейства C8051Fxxx программируются в системе, на плате, с помощью встроенного в микроконтроллер модуля отладки JTAG. Интерфейс JTAG (соответствует спецификации IEEE 1149.1) позволяет производить программирование FLASH памяти, а также выполнять тестирование и отладку ваших программ. Программирование микроконтроллера, а также отладка программ в режиме реального времени (функция эмулятора) производится с помощью программатора по USB интерфейсу (рис. 2.1). Микроконтроллеры имеют развитую систему портов ввода/вывода, а также, конструктивно встроенные интерфейсы.



Рис. 2.1. Программатор USB DEBUG ADAPTER

Отладочная плата была разработана в Объединённом Институте Ядерных Исследований (ОИЯИ) специально для университета «Дубна». Имея широкий спектр интерфейсов, она оптимальна для применения в учебном процессе. Плата имеет внешние интерфейсы RS-232/RS-422/RS-485, может

запитываться от нестабилизированного источника питания до 15В или через USB-конвертор. Дополнительно для питания дополнительных устройств есть вывод питания +5В.



Рис. 2.2. Отладочная плата на базе контроллера C8051F410 фирмы Silicon Laboratories

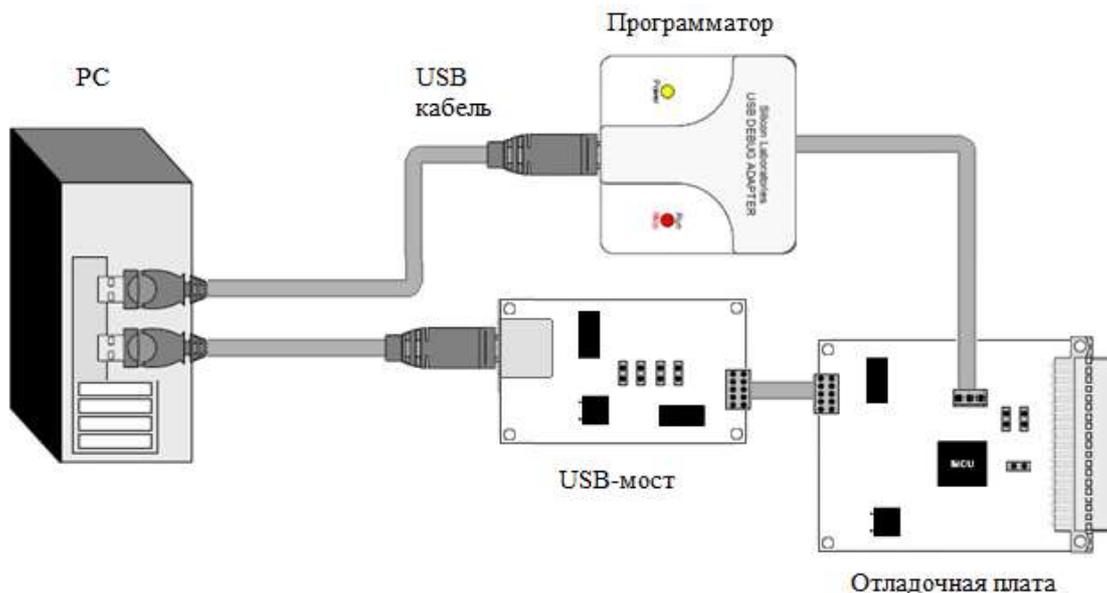


Рис. 2.3. Схема сборки отладочного комплекта

2.3. Контроллер HD44780 фирмы Hitachi

Контроллер HD44780 (а также совместимый с ним KS0066) – стандарт на контроллеры черно-белых жидкокристаллических знаковинтезирующих дисплеев с параллельным 4- или 8-битным интерфейсом. Эти контроллеры были фактически монополистами на рынке в 90-е годы. В настоящее время все активнее вытесняются контроллерами с SPI интерфейсами, а также контроллерами цветных дисплеев.

Контроллер HD44780 потенциально может управлять 2-мя строками по 40 символов в каждой (для модулей с 4-мя строками по 40 символов используются два однотипных контроллера), при матрице символа 5 x 7 точек. Контроллеры также поддерживает символы с матрицей 5 x 10 точек, но

в последние годы ЖКИ-модули с такой матрицей практически не встречаются, поэтому можно считать, что фактически бывают только символы 5 x 7 точек. Существует множество различных более-менее стандартных форматов ЖКИ-модулей. Принципиальных ограничений на комбинации и количество отображаемых символов контроллер не накладывает – модуль может иметь любое количество символов от 1 до 80, хотя в некоторых комбинациях программная адресация символов может оказаться не очень удобной. В рамках одного формата могут производиться ЖКИ-модули нескольких конструктивов, отличающихся как габаритами ЖКИ (и, как следствие, размерами символов), так и размерами платы и посадки. В рамках одного конструктива ЖКИ-модуль может иметь еще ряд модификаций. В частности, могут применяться несколько типов ЖКИ, отличающихся:

- цветом фона;
- цветом символов;
- по применяемым ЖК-материалам;
- структуре: TN, STN и FSTN типа.

ЖКИ STN и FSTN типа имеют более высокую стоимость, но одновременно обладают повышенной контрастностью и вдвое большим максимальным углом обзора, причем ЖКИ FSTN типа имеют лучшие характеристики, чем STN.

ЖКИ-модули могут оснащаться задней подсветкой, размещаемой между ЖКИ и печатной платой, для чего ЖКИ производятся с полупрозрачным или прозрачным задним слоем (в последнем случае считывание информации возможно только при наличии подсветки).

Собственно подсветка может быть реализована несколькими способами:

- электролюминисцентная панель, представляющей собой тонкую пленку, излучающую свет при прикладывании переменного тока повышенного напряжения порядка 100...150 В;
- люминисцентной лампой с холодным катодом (также работающей при повышенном напряжении), излучение которой равномерно распределяется по всей площади ЖКИ с помощью отражателя или плоского световода;
- подсветка на основе светодиодной матрицы.

Первые два способа подсветки обеспечивают высокую яркость и могут иметь белый тон свечения при относительно низком потреблении, но требуют наличия источника повышенного напряжения, что создает некоторые трудности при создании аппаратуры с автономным питанием. Напротив, светодиодная подсветка не требует высоковольтного источника (прямое падение напряжения составляет 4,2 В) и при использовании несложного источника тока позволит производить питание от источника с напряжением 5 В. Кроме того, светодиодная подсветка имеет значительно большее (в десятки раз) время наработки, а также только она допустима к эксплуатации в расширенном диапазоне температур (-20°C...+70°C).

Для соединения ЖКИ-модуля с управляющей системой используется параллельная синхронная шина, насчитывающая 8 или 4 (выбирается программно) линий данных DB0...DB7, линию выбора операции R/W, линию выбора регистра RS и линию стробирования/синхронизации E. Кроме линий управляющей шины имеются две линии для подачи напряжения питания 5 В – GND и VCC, и линия для подачи напряжения питания драйвера ЖКИ – V0.

Указанные выше названия линий шины являются стандартными, но существует множество различных вариантов расположения контактов у каждого конкретного конструктива ЖКИ-модуля. На самом деле, единственным реально стандартным вариантом расположения контактов является двухрядное 14-ти контактное поле, расположенное вертикально в левой части модуля, а также совпадающее с ним двухрядное 16-ти контактное поле, содержащее дополнительную пару контактов с подключенными к ней выводами питания подсветки. В любом случае, для получения достоверной информации необходимо воспользоваться соответствующей справочной литературой изготовителя модуля.

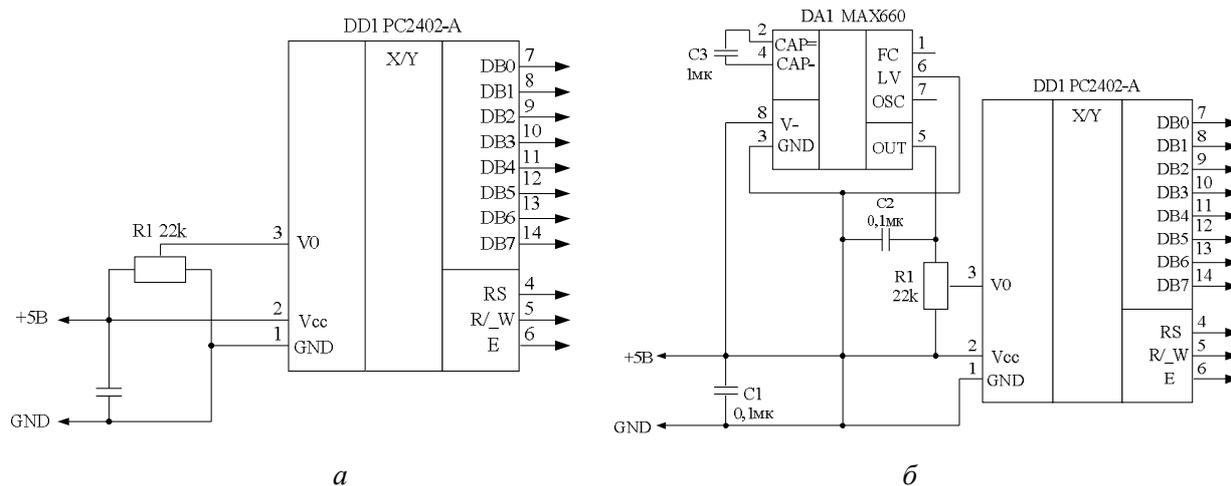


Рис. 2.4. Подключение питания ЖКИ-модуля

На начальном этапе необходимо подать питание на ЖКИ-модуль и добиться от него признаков работоспособности. Схема включения модуля, рассчитанного на стандартный диапазон температур, показана на рисунке (рис. 2.4, а). Подстроечный резистор R1 позволяет плавно менять напряжение питания драйвера ЖКИ, что приводит к изменению угла поворота жидких кристаллов. Этим резистором можно отрегулировать фактическую контрастность при некотором преимущественном угле наблюдения (снизу-вверх или сверху-вниз). Включение в данную схему ЖКИ-модуля, рассчитанного на расширенный диапазон температур, не приведет к успеху, так как из-за особенностей применяемых в них ЖК-материалов, эти ЖКИ требуют повышенного напряжения питания и при питании напряжением 5 В изображение либо будет отсутствовать совсем, либо будет слабоконтрастным. Для преодоления ситуации необходимо подать на вывод V0 отрицательное напряжение (напряжение на ЖКИ определяется разностью VCC и V0), составляющее в предельном случае – 5 В. Если в схеме отсутствует источник отрицательного напряжения, то не составляет труда собрать простейший преобразователь, например, по схеме на рисунке (рис. 2.4б).

Для активизации ЖКИ-модуля подается напряжение питания и вращается движок резистора R1 (рис. 2.4). После окончания цикла внутренней инициализации модуль включается в режим развертки одной верхней строки. При изменении напряжения на выводе V0 сегменты этой строки должны менять свое состояние от прозрачного до непрозрачного, что является свидетельством правильного подключения питания модуля и работоспособности контроллера и драйверов ЖКИ. Далее устанавливается движок в такое положение, при котором изображение сегментов в верхней строке едва проступает на основном фоне ЖКИ. Теперь ЖКИ-модуль готов к приему и отображению информации. После того, как на индикаторе отобразился любой текст, можно более точно отрегулировать изображение в соответствии с необходимой контрастностью и требуемым углом наблюдения.

Для соединения модуля с управляющей системой можно выбрать один из двух вариантов: по 8-ми или 4-х разрядной шине. В первом случае потребуется 11 сигнальных линий, во втором – только 7. Сам процесс обмена также может быть организован двояко. Можно подключить ЖКИ-модуль к системной шине (если таковая имеется) и выполнить обмен в синхронном режиме с максимальной скоростью. Этому широко распространенному в прошлые годы способу присущ ряд недостатков:

- большинство современных устройств выполняется с применением однокристалльных микро-ЭВМ без использования дополнительной внешней памяти и, как следствие, системная шина у этих устройств просто отсутствует;
- в современных системах повышенной сложности и производительности, у которых присутствует дополнительная память и, естественно, системная шина, скорость операций на шине находится за пределами возможностей контроллера HD44780 (2 МГц при питании 5 В и 1 МГц при 3 В). Это может потребовать введения дополнительных схем для замедления скорости работы шины при выполнении операций обмена с ЖКИ-модулем;
- подключение к системной шине в большинстве случаев потребует вводить схемы дешифрации и формирования сигналов E и R/W, что опять приведет к дополнительным затратам.

Все сказанное выше не означает, что вариант с подключением к шине принципиально неэффективен. В какой-то конкретной системе этот способ, наоборот, может быть самым оптимальным. Кроме того, некоторые современные процессоры, например, семейство MC68HC12, имеет встроенные средства для формирования сигналов выборки (CS), с возможностью программно определить скорость обмена с каждым конкретным устройством.

Другой вариант – обмен с ЖКИ-модулем выполняется чисто программными средствами, через порты ввода-вывода управляющей микро-ЭВМ. Остановимся подробнее на этом варианте, так как он позволяет рассмотреть общий случай, абстрагируясь от конкретной системы. Вариант соединения с системной шиной, напротив, требует рассмотрения конкретных устройств. Поэтому в случае необходимости можно рекомендовать изучить временные диаграммы операций чтения и записи, а также значения временных параметров и сконструировать соответствующие управляющие схемы.

На рисунке (рис. 2.5, а) приведена схема подключения ЖКИ-модуля с 8-ми разрядной шиной к некоторой абстрактной микро-ЭВМ XYZ. Эта микро-ЭВМ содержит два порта: 8-ми разрядный двунаправленный PA0...PA7, к которому подключена шина DB0...DB7 ЖКИ-модуля, и 3-х разрядный PB0...PB2, к которому подключены линии управляющих сигналов: E, RS, R/W. На рисунке (рис. 2.5б) можно видеть схему подключения ЖКИ-модуля к этой же микро-ЭВМ в 4-х разрядном режиме. Обратите внимание, что для обмена в 4-х разрядном режиме используется старшая тетрада шины данных – DB4...DB7.

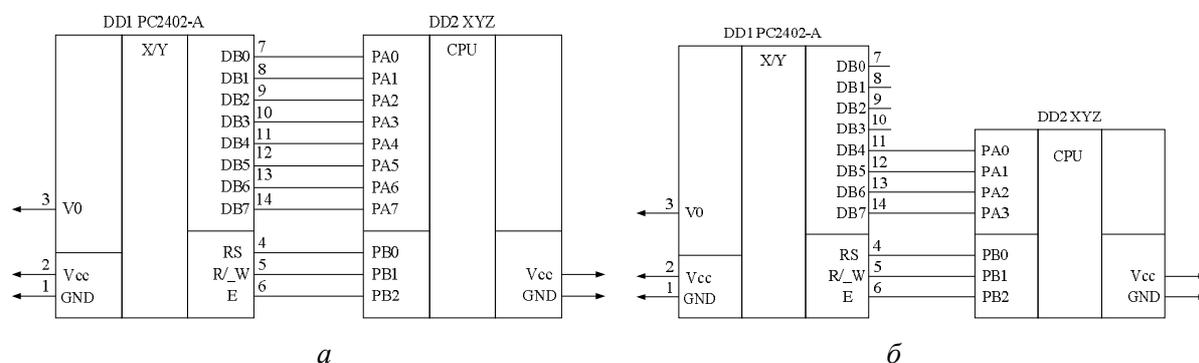


Рис. 2.5. Подключение к управляющей системе

В соответствии с временной диаграммой в исходном состоянии сигнал E = 0, сигнал R/W = 0, значение сигнала RS – произвольное, шина данных DB0...DB7 в состоянии высокого импеданса (HI). Такое состояние управляющих сигналов (E и R/W) должно поддерживаться все время в промежутках между операциями обмена с ЖКИ-модулем. Шина данных в эти моменты в принципе свободна, и может использоваться в мультиплексном режиме для каких-либо других целей, например, для сканирования матрицы клавиатуры. Естественно, необходимо позаботиться об исключении конфликтов на шине данных в момент совершения операций обмена с ЖКИ-модулем.

Последовательность действий, которые необходимо выполнять управляющей системе при совершении операций записи и чтения для 8-ми и 4-х разрядной шины приведены рис. 2.6 – 2.9.



Рис. 2.6. Алгоритм записи для 8-ми разрядной шины



Рис. 2.7. Алгоритм чтения для 8-ми разрядной шины

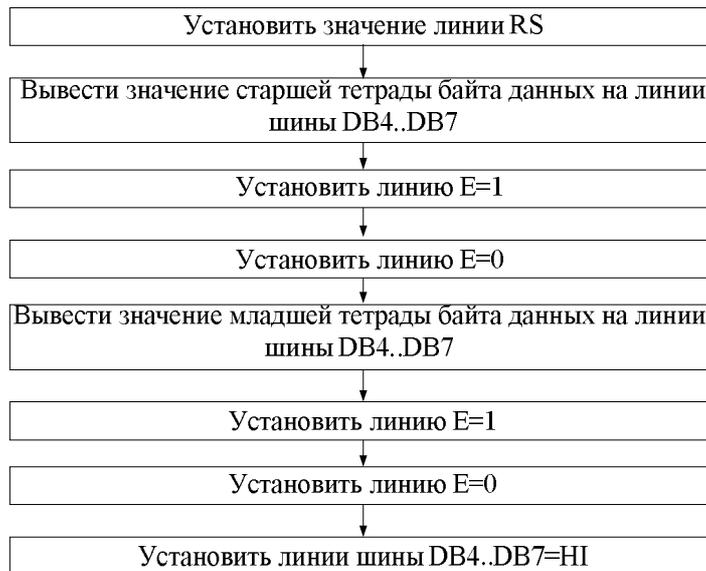


Рис. 2.8. Алгоритм записи для 4-х разрядной шины

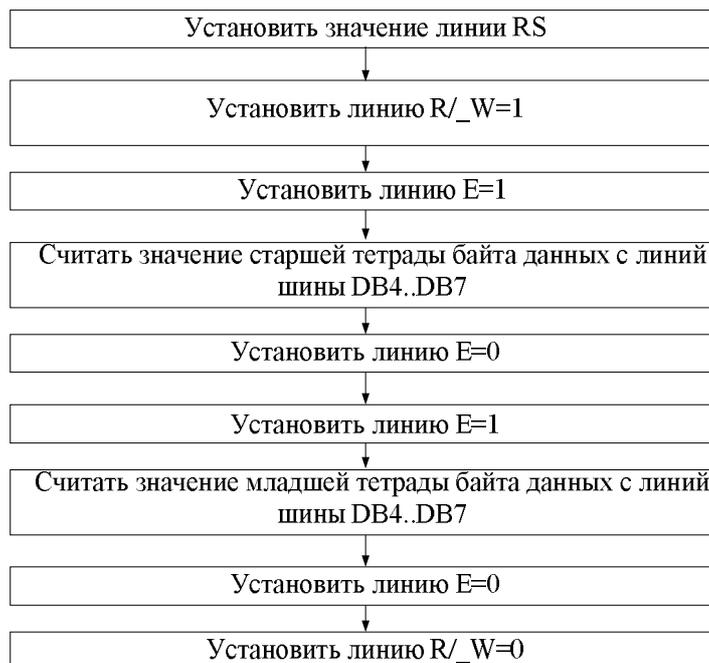


Рис. 2.9. Алгоритм чтения для 4-х разрядной шины

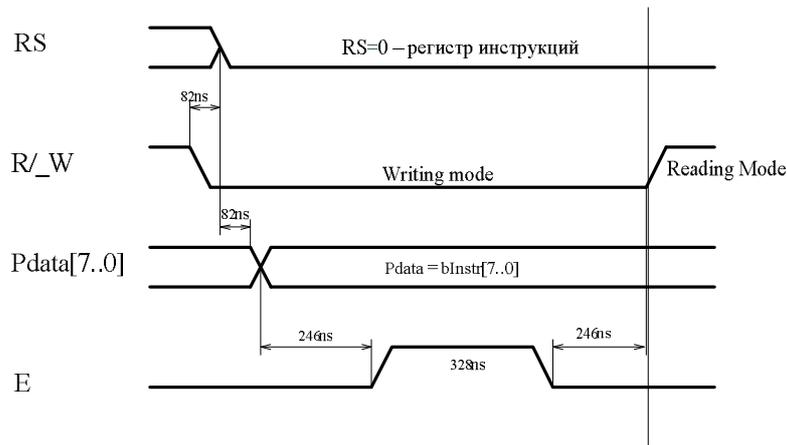


Рис. 2.10. Временная диаграмма записи в регистр инструкций параллельного интерфейса контроллера ЖК-модуля (8-bit Mode)

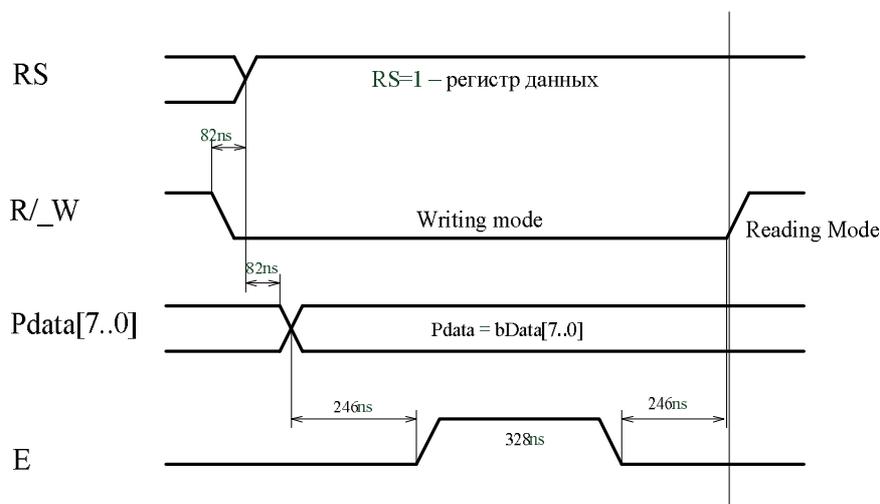


Рис. 2.11. Временная диаграмма записи в регистр данных параллельного интерфейса контроллера ЖК-модуля (8-bit Mode)

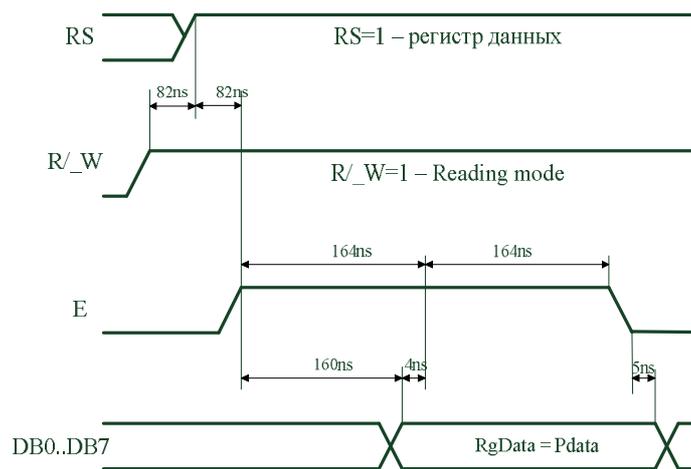


Рис. 2.12. Временная диаграмма чтения из регистра данных параллельного интерфейса контроллера ЖК-модуля (8-bit Mode)

Приведенные операции подразумевают, что время выполнения каждого шага составляет не менее 250 нс. При использовании современных быстродействующих микро-ЭВМ это условие может быть легко нарушено, поэтому необходимо тщательно контролировать минимальные значения вре-

менных интервалов, чтобы они всегда находились в области допустимых значений, указанных в таблице 2.1, 2.2 и при необходимости вводить задержки.

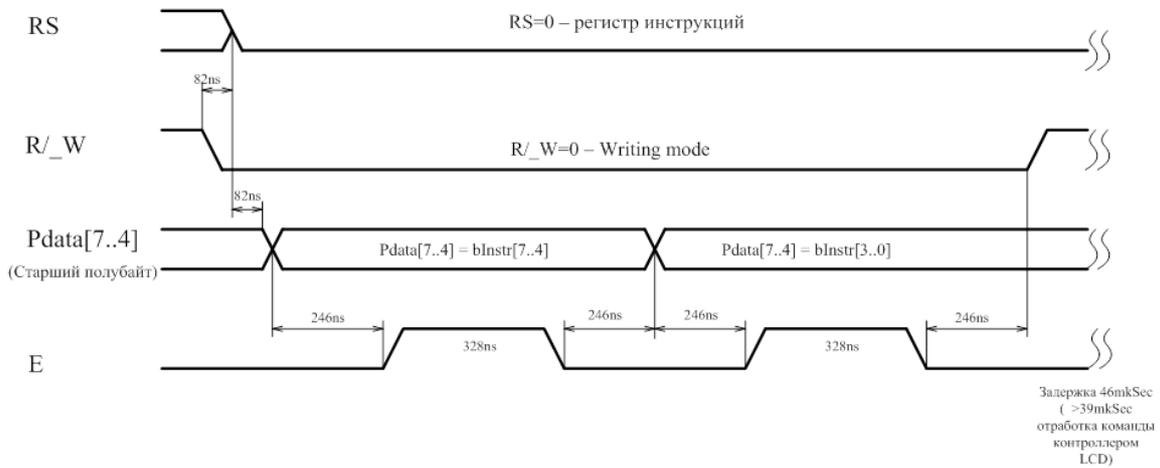


Рис. 2.13. Временная диаграмма записи в регистр инструкций параллельного интерфейса контроллера ЖК-модуля (4-bit Mode)

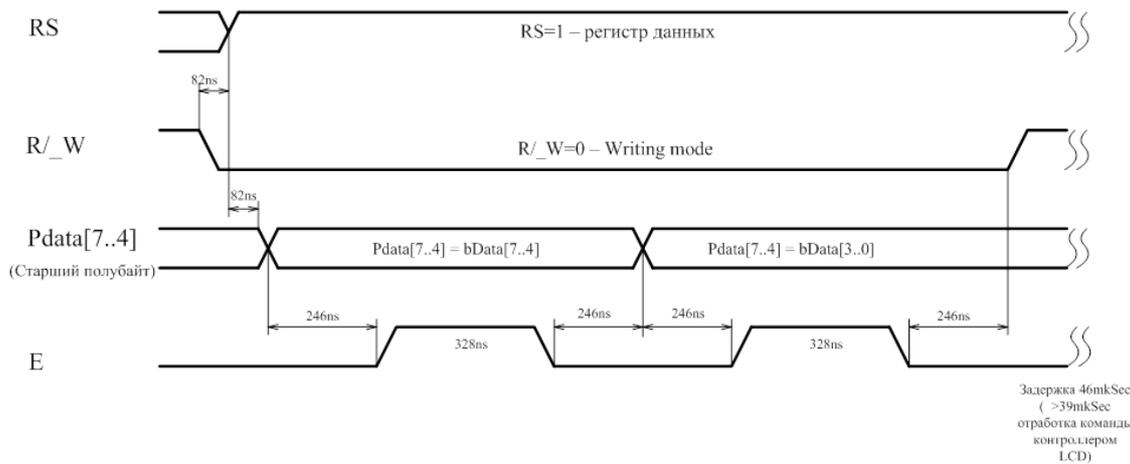


Рис. 2.14. Временная диаграмма записи в регистр данных параллельного интерфейса контроллера ЖК-модуля (4-bit Mode)

Таблица 2.1. Значения временных характеристик. Операция записи

Параметр	Обозначение	Мин.	Макс.	Единица
Период сигнала E	tcycE	500	-	нс
Положительный полупериод сигнала E	PWEN	230	-	нс
Фронт/спад сигнала E	tEr, tEf	-	20	нс
Установление адреса	tAS	40	-	нс
Удержание адреса	tAH	10	-	нс

Установление данных	tDSW	80	-	нс
Удержание данных	tDSW	10	-	нс

Таблица 2.2. Значения временных характеристик. Операция чтения

Параметр	Обозначение	Мин.	Макс.	Единица
Период сигнала E	tcycE	500	-	нс
Положительный полупериод сигнала E	PWEN	230	-	нс
Фронт/спад сигнала E	tEr, tEf	-	20	нс
Установление адреса	tAS	40	-	нс
Удержание адреса	tAH	10	-	нс
Установление данных	tDSW	-	160	нс
Удержание данных	tDSW	5	-	нс

Описанные выше операции записи/чтения байта являются базовыми для осуществления обмена с ЖКИ-модулем. Реализация этих двух операций – единственное, что отличает процесс обмена по 8-ми разрядной шине от обмена по 4-х разрядной шине. На основе этих двух операций, реализованных программно (когда модуль подключен к портам микро-ЭВМ), или аппаратно (когда модуль подключен к системной шине), строятся все виды операций программирования и управления.

Несколько слов о подключении ЖКИ-модулей формата 40 x 4. Эти модели содержат два контроллера HD44780 и фактически представляют собой два модуля 40 x 2. Для сокращения числа соединений все сигналы обоих контроллеров соединены параллельно, исключение составляет только сигнал E, одновременно выполняющий роль тактового сигнала и сигнала выборки микросхемы CS, поэтому обмен с каждым из контроллеров ведется попеременно.

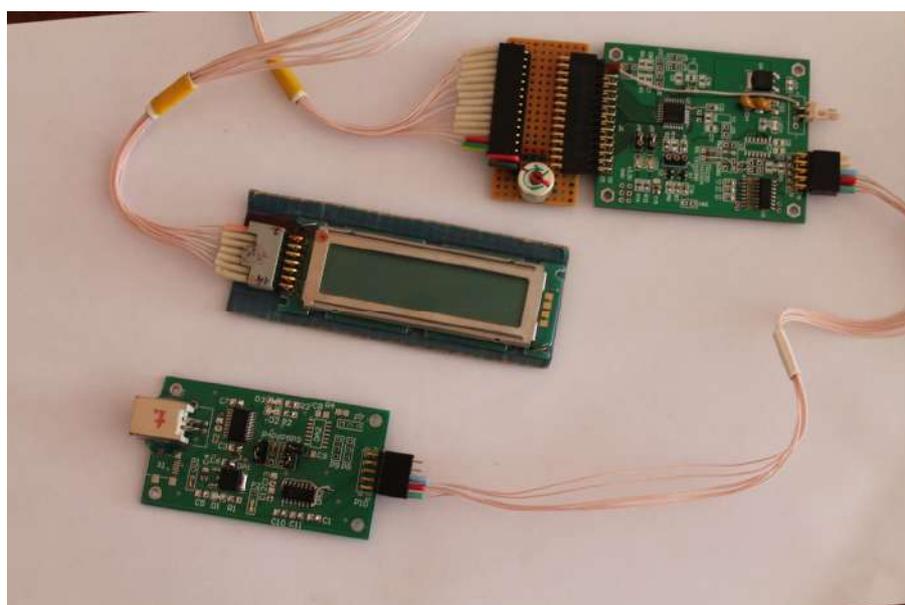


Рис. 2.15. Вид собранного отладочного комплекта

2.5. Программирование и управление

Структурная схема контроллера приведена на рисунке (рис. 2.16). Основные элементы, с которыми приходится взаимодействовать при программном управлении: регистр данных (DR), регистр команд (IR), видеопамять (DDRAM), ОЗУ знакогенератора (CGRAM), счетчик адреса памяти (AC), флаг занятости контроллера.

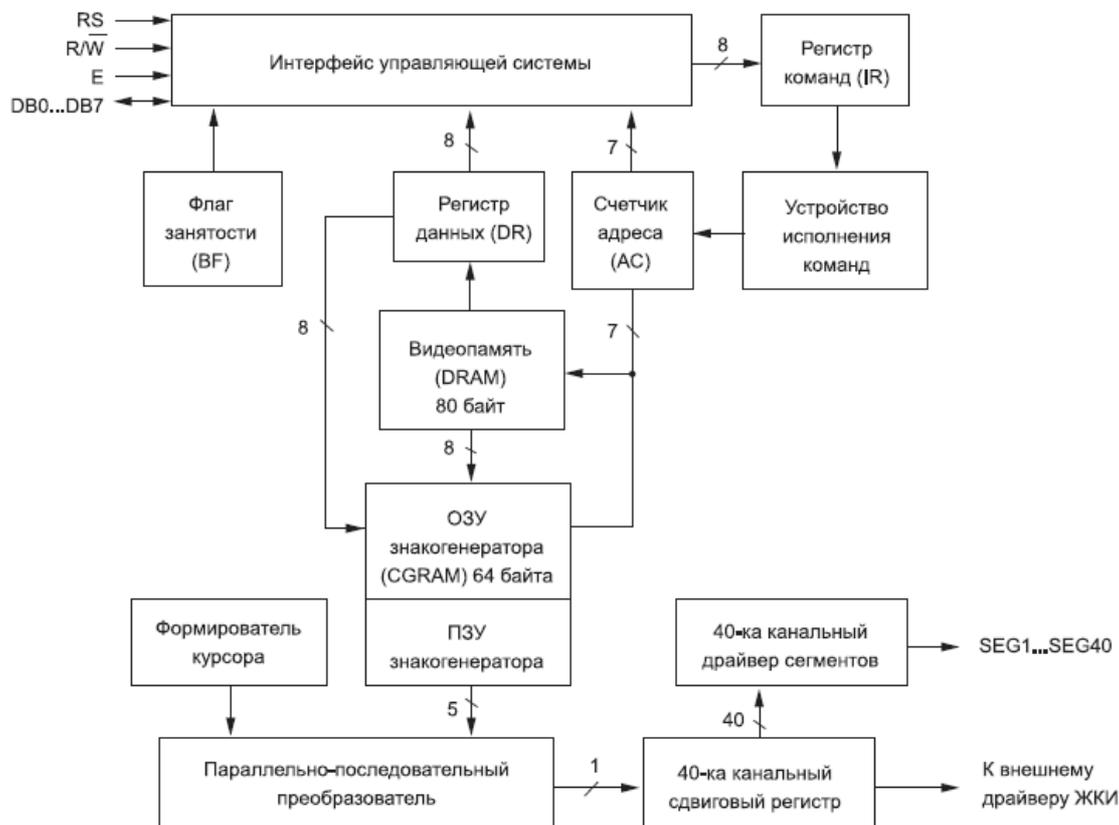


Рис. 2.16. Структурная схема контроллера HD44780

Другие элементы не являются объектом прямого взаимодействия с управляющей программой они участвуют в процессе регенерации изображения на ЖКИ: знакогенератор, формирователь курсора, сдвиговые регистры и драйверы (напоминаем, что приведенная схема – упрощенная, и многие не важные для получения общей картины промежуточные элементы на ней опущены).

Управление контроллером ведется посредством интерфейса управляющей системы. Основными объектами взаимодействия являются регистры DR и IR. Выбор адресуемого регистра производится линией RS, если RS = 0 – адресуется регистр команд (IR), если RS = 1 – регистр данных (DR).

Данные через регистр DR, в зависимости от текущего режима, могут помещаться (или прочитываться) в видеопамять (DDRAM) или в ОЗУ знакогенератора (CGRAM) по текущему адресу, указываемому счетчиком адреса (AC). Информация, попадающая в регистр IR, интерпретируется устройством выполнения команд как управляющая последовательность. Прочтение регистра IR возвращает в 7-ми младших разрядах текущее значение счетчика AC, а в старшем разряде флаг занятости (BF).

Видеопамять, имеющая общий объем 80 байтов, предназначена для хранения кодов символов, отображаемых на ЖКИ. Видеопамять организована в две строки по 40 символов в каждой. Эта привязка является жесткой и не подлежит изменению. Другими словами, независимо от того, сколько реальных строк будет иметь каждый конкретный ЖКИ-модуль, скажем, 80 x 1 или 20 x 4, адресация видеопамяти всегда производится как к двум строкам по 40 символов.

Будучи устройством с динамической индикацией, контроллер циклически производит обновление информации на ЖКИ. Сам ЖКИ организован как матрица, состоящая в зависимости от режима работы из 8-ми (одна строка символов 5 x 7 точек), 11-ти (одна строка символов 5 x 10 точек) или 16-

ти (две строки символов 5 x 7 точек) строк по 200 сегментов (когда строка насчитывает 40 символов) в каждой. Собственный драйвер контроллера HD44780 имеет только 40 выходов (SEG1...SEG40) и самостоятельно может поддерживать только 8-ми символьные ЖКИ. Это означает, что ЖКИ-модули форматов до 8 x 2 реализованы на одной единственной микросхеме HD44780, модули, имеющие большее количество символов, содержат дополнительные микросхемы драйверов, например, HD44100, каждая из которых дополнительно предоставляет управление еще 40-ка сегментами.

Особняком стоят ЖКИ-модули формата 16 x 1. Они также реализованы с помощью одной единственной микросхемы HD44780, но одна 16-ти символьная строка в них фактически составлена из двух 8-ми символьных. И хотя это усложняет программное управление, ведь строка оказывается логически разорванной посередине, тем не менее, экономически это оправдано, ибо позволило создать ЖКИ-модуль, содержащий всего одну микросхему. Другой вариант пространственной адресации встречается в 4-х строчных модулях. Из-за проблем разводки токоведущих дорожек, первая и вторая строки этих модулей являются таковыми как обычно, третья же является продолжением первой строки, а четвертая – второй.

У контроллера HD44780 существует набор внутренних флагов, определяющих режимы работы различных элементов контроллера (таблица 3.3). В таблице 3.4 приведены значения управляющих флагов непосредственно после подачи на ЖКИ-модуль напряжения питания. Переопределение значений флагов производится специальными командами, записываемыми в регистр IR, при этом комбинации старших битов определяют группу флагов или команду, а младшие содержат собственно флаги.

Список управляющих комбинаций битов регистра IR и выполняемые ими команды приведены в таблице 3.5. Так как на момент включения ЖКИ-модуль ничего не отображает (флаг D = 0), то для того, чтобы вывести какой-либо текст необходимо, как минимум, включить отображение, установив флаг D = 1. Вот пример широко распространенной последовательности для инициализации ЖКИ-модуля: \$38, \$0C, \$06 (знак «\$» перед числом указывает на шестнадцатеричное основание). \$38 устанавливает режим отображения 2-х строк с матрицей 5 x 8 точек и работу с 8-ми разрядной шиной данных; \$0C включает отображение на экране ЖКИ-модуля, без отображения курсоров; \$06 устанавливает режим автоматического перемещения курсора слева-направо после вывода каждого символа.

Таблица 2.3. Флаги, управляющие работой контроллера HD44780

Обозначение	Назначение флагов
I/D	Режим смещения счетчика адреса АС, 0 – уменьшение, 1 – увеличение.
S	Флаг режима сдвига содержимого экрана. 0 – сдвиг экрана не производится, 1 – после записи в DDRAM очередного кода экран сдвигается в направлении, определяемым флагом I/D: 0 – вправо, 1 – влево. При сдвиге не производится изменение содержимого DDRAM. изменяются только внутренние указатели расположения видимого начала строки в DDRAM.
S/C	Флаг-команда, производящая вместе с флагом R/L операцию сдвига содержимого экрана (так же, как и в предыдущем случае, без изменений в DDRAM) или курсора. Определяет объект смещения: 0 – сдвигается курсор, 1 – сдвигается экран.
R/L	Флаг-команда, производящая вместе с флагом S/C операцию сдвига экрана или курсора. Уточняет направление сдвига: 0 – влево, 1 – вправо.
D/L	Флаг, определяющий ширину шины данных: 0 – 4 разряда, 1 – 8 разрядов.
N	Режим развертки изображения на ЖКИ: 0 – одна строка, 1 – две строки.
F	Размер матрицы символов: 0 – 5 x 8 точек, 1 – 5 x 10 точек.
D	Наличие изображения: 0 – выключено, 1 – включено.
C	Курсор в виде подчеркика: 0 – выключен, 1 – включен.
B	Курсор в виде мерцающего знакоместа: 0 – выключен, 1 – включен.

Таблица 2.4. Значения управляющих флагов после подачи питания

Значение	Описание
I/D = 1	Режим увеличения счетчика на 1
S = 0	Без сдвига изображения
D/L = 1	8-ми разрядная шина данных
N = 0	Режим развертки одной строки
F = 0	Символы с матрицей 5 x 8 точек
D = 0	Отображение выключено
C = 0	Курсор в виде подчеркива выключен
B = 0	Курсор в виде мерцающего знакоместа выключен

Контроллер HD44780 поддерживает операции записи и операции чтения. Чтение регистра DR приводит к загрузке содержимого DDRAM или CGRAM, в зависимости от текущего режима, при этом курсор смещается на одну позицию, как и при записи. Чтение регистра IR возвращает 8 значащих разрядов, причем в 7-ми младших содержится текущее значение счетчика AC (7 разрядов, если адресуется DDRAM, и 6 – если CGRAM), а в старшем – флаг занятости BF. Этот флаг имеет значение 1 когда контроллер занят и 0 – когда свободен. Необходимо учитывать, что большинство операций, выполняемых контроллером, занимают значительное время, около 40 мкс, а время выполнения некоторых доходит до единиц миллисекунд, поэтому цикл ожидания снятия флага BF должен обязательно присутствовать в программах драйвера ЖКИ-модуля и предшествовать совершению любой операции (естественно, кроме операции проверки флага BF).

Таблица 2.5. Управляющие комбинации битов регистра IR

Номера битов								Назначение
D7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	0	0	0	0	1	Очистка экрана, AC = 0, адресация AC на DDRAM
0	0	0	0	0	0	1	-	AC = 0, адресация на DDRAM, сброшены сдвиги, начало строки адресуется в начале DDRAM
0	0	0	0	0	1	I/D	S	Выбирается направление сдвига курсора или экрана
0	0	0	0	1	D	C	B	Выбирается режим отображения
0	0	0	1	S/C	R/L	-	-	Команда сдвига курсора/экрана
0	0	1	DL	N	F	-	-	Определение параметров развертки и ширины шины данных
0	1	AG	AG	AG	AG	AG	AG	Присвоение счетчику AC адреса в области CGRAM
1	AD	AD	AD	AD	AD	AD	AD	Присвоение счетчику AC адреса в области DDRAM

Примечание. После совершения операции записи или чтения DDRAM и появления после нее признака готовности (BF = 0), прочитанное в этом же цикле (вместе с флагом BF) значение AC скорее всего не будет достоверным. Дело в том, что между появлением признака готовности и вычислением контроллером нового значения AC существует некоторый временной интервал, составляющий около 4 мкс при тактовой частоте контроллера 270 кГц. Поэтому, если необходимо получить истинное значение AC, нужно совершить повторную операцию прочтения IR спустя не менее чем 4 мкс (если контроллер работает на частоте 270 кГц время ожидания необходимо пропорционально увеличить).

Вывод на экран символа производится записью его кода в регистр DR. При этом символ размещается в DDRAM по текущему адресу, указываемому AC, а значение AC увеличивается или уменьшается на 1. Чтобы произвести переустановку курсора на нужную позицию, необходимо присвоить AC соответствующее значение (см. таблицу 9). Здесь есть одна тонкость. Когда производится последовательная запись символов и в результате заполняется вся строка, курсор автоматически переходит на вторую строку, но если необходимо принудительно установить курсор, скажем, на начало второй строки, то будет неверным присвоить AC казалось бы логичное значение \$28 (40), правильным является значение \$40 (64). Значения адресов DDRAM в диапазоне \$28...\$3fF (а равно и \$68...\$7F) являются неопределенными и результаты работы с ними могут быть непредсказуемыми.

Необходимо учитывать, что контроллеры, устанавливаемые на ЖКИ-модули, могут иметь различные наборы символов, причем это может зависеть как от производителя контроллера, так и от модификации данной конкретной модели. Например, фирма Powertip выпускает ЖКИ-модули с четырьмя базовыми модификациями наборов символов: японской, европейской, французской и русской.

Несколько слов о процессе инициализации ЖКИ-модуля. Производитель контроллера рекомендует выполнять следующую последовательность действий для инициализации. Выдержать паузу не менее 15 мс между установлением рабочего напряжения питания ($> 4,5$ В) и выполнением каких-либо операций с контроллером. Первой операцией выполнить команду, выбирающую разрядность шины (это должна быть команда \$30 независимо от того, какой разрядности интерфейс вы собираетесь использовать в дальнейшем), причем перед выполнением этой операции не проверять значение флага BF. Далее опять выдержать паузу не менее 4,1 мс и повторить команду выбора разрядности шины, причем перед подачей команды вновь не производить проверку флага BF. Следующим шагом необходимо вновь выдержать паузу, на этот раз 100 мкс, и в третий раз повторить команду установления разрядности шины, вновь без проверки BF. Эти три операции являются инициализирующими и призваны вывести контроллер в исходный режим работы (то есть перевести в режим работы с 8-ми разрядной шиной) из любого состояния. Следом за ними нормальным порядком (без выдерживания пауз, но с проверкой флага BF) выполняется инициализация режимов работы с выдачей инициализирующей последовательности, аналогичной указанной в таблице 7 (содержащей, в том числе команду выбора необходимой разрядности шины).

Необходимо помнить, что когда объявляется режим работы с 4-х разрядной шиной, то есть выдаете команду \$20, то делаете это обычно из 8-ми разрядного режима, который устанавливается автоматически после подачи напряжения питания, а значит вы не сможете адекватно объявить необходимое значение флагов N и F, располагающихся в младшей тетраде команды установки разрядности шины. Поэтому команду необходимо повторить в уже установившемся 4-х разрядном режиме путем последовательной передачи двух тетрад, то есть для 4-х разрядного режима образом.

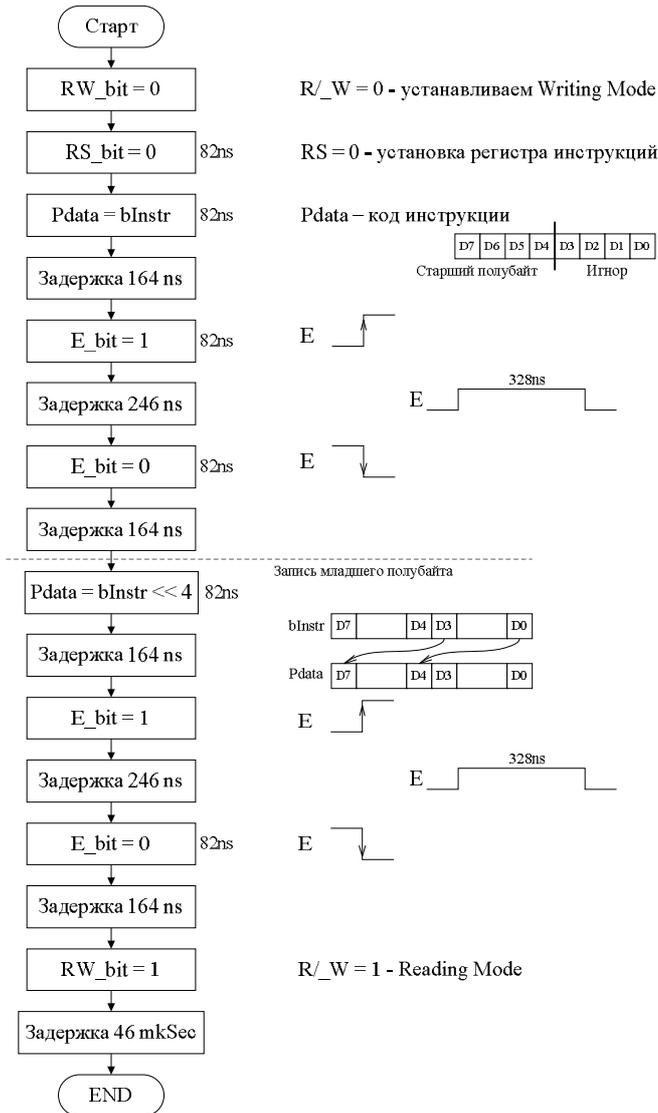


Рис. 4.1. Блок-схема алгоритма записи в регистр инструкций параллельного интерфейса контроллера ЖК-модуля (4-bit Mode)

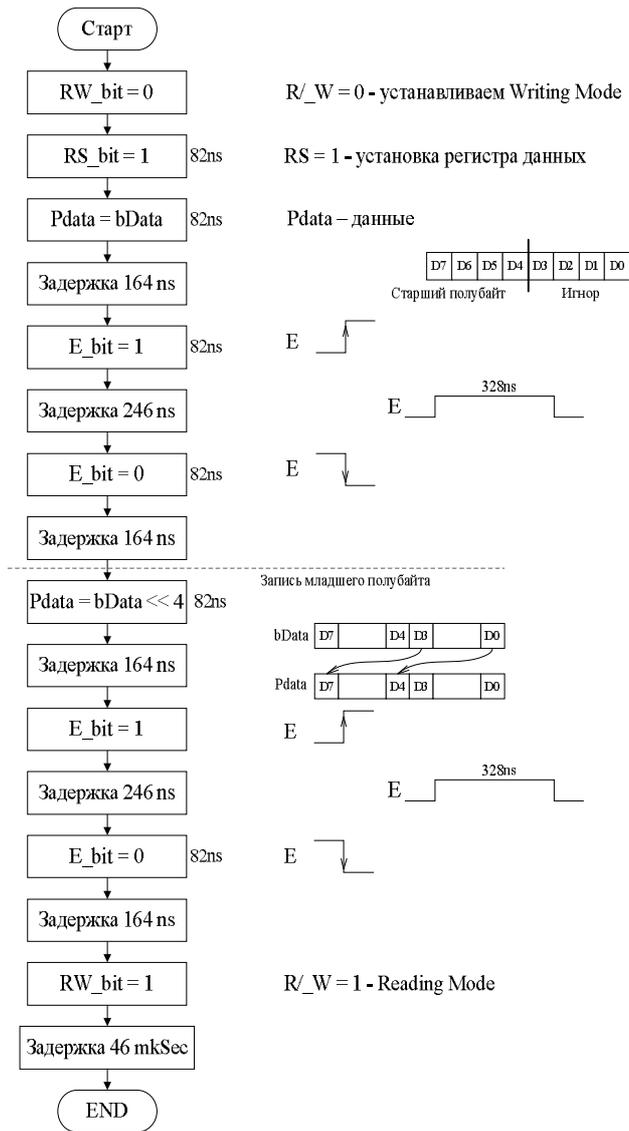


Рис. 4.2. Блок-схема алгоритма записи в регистр данных параллельного интерфейса контроллера ЖК-модуля (4-bit Mode)

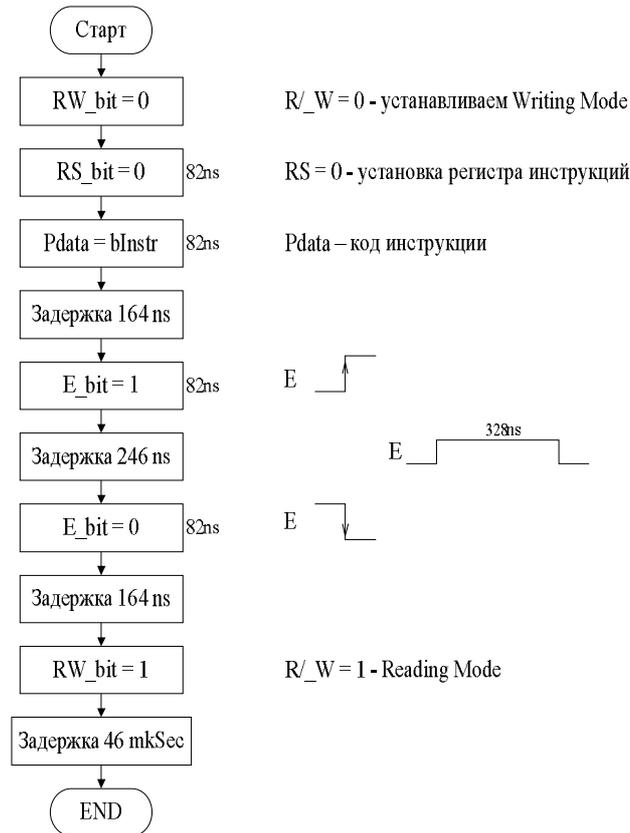


Рис. 4.3. Блок-схема алгоритма записи в регистр инструкций параллельного интерфейса контроллера ЖК-модуля (8-bit Mode)

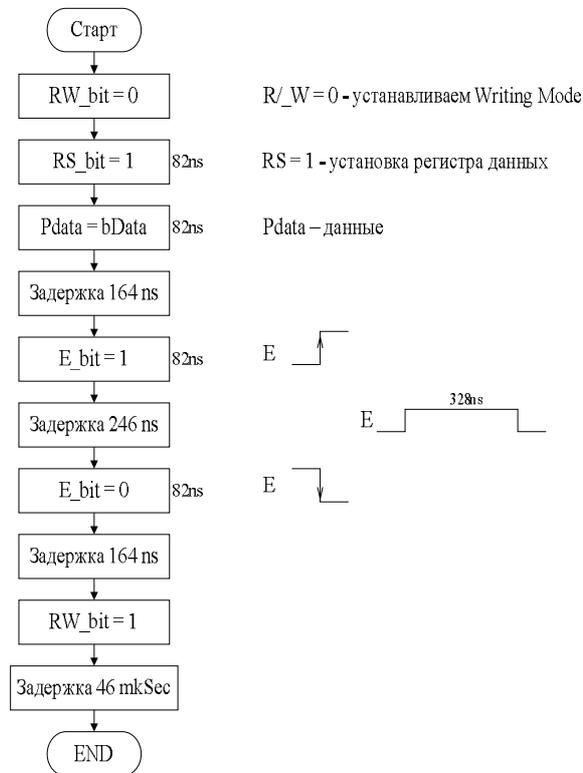


Рис. 4.4. Блок-схема алгоритма записи в регистр данных параллельного интерфейса контроллера ЖК-модуля (8-bit Mode)

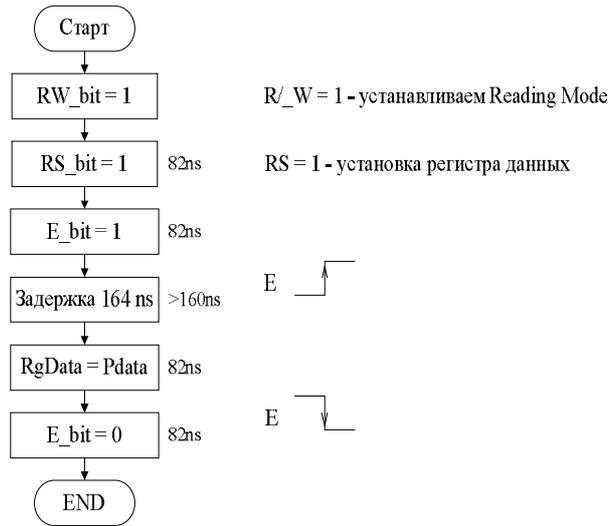


Рис. 4.5. Блок-схема алгоритма чтения из регистра данных параллельного интерфейса контроллера ЖК-модуля (8-bit Mode)

Функции ЖК-модуля



Рис. 4.6. LCD on (включение ЖК-модуля)



Рис. 4.7. LCD off (выключение ЖК-модуля)



Рис. 4.8. LCD clr (очистка экрана)



Рис. 4.9. LCD Entry_Mode_set (установка режима ввода)

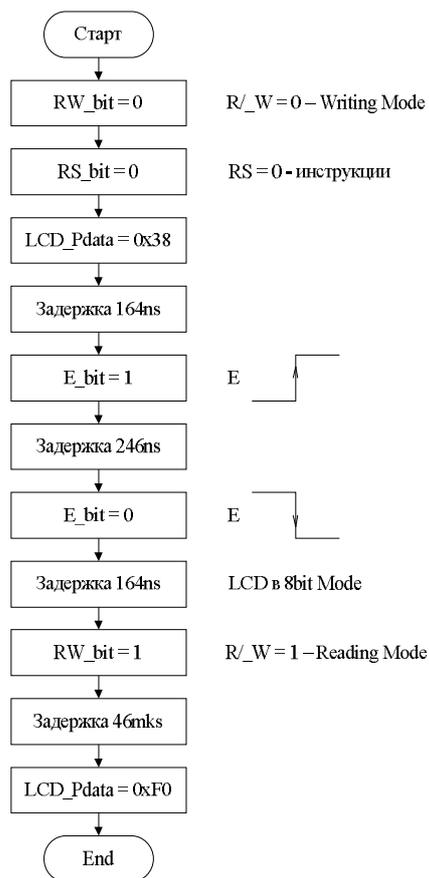


Рис. 4.10. LCD_F_8bitMode_Set (установка 8bitMode для ЖК-модуля)

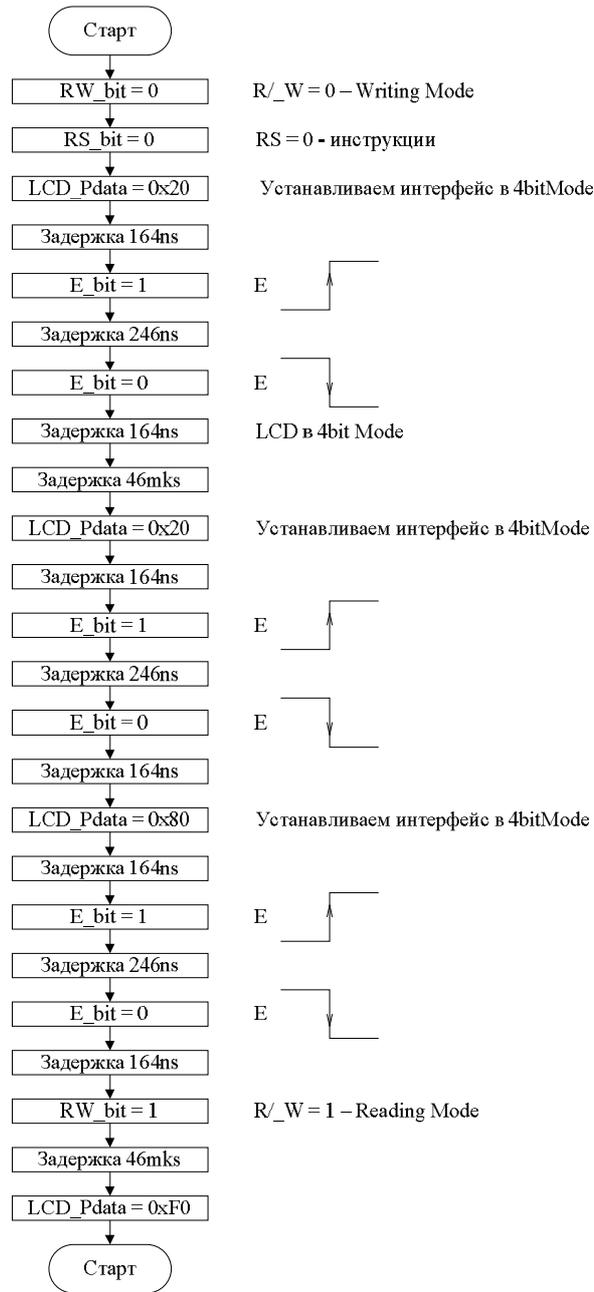


Рис. 4.11. LCD_F_4bitMode_Set (установка 4bitMode для ЖК-модуля)

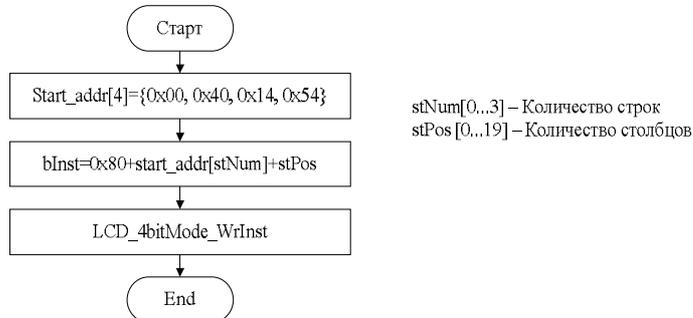


Рис. 4.12. LCD_DD_Ram_addr_set (установка адреса DD Ram)

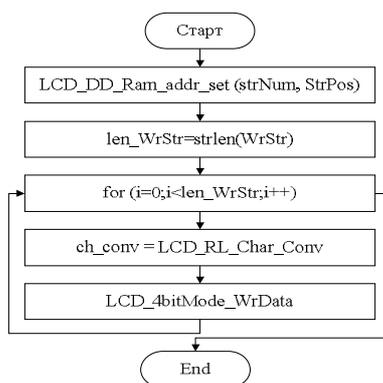


Рис. 4.13. LCD_Wr_Str (запись в строку)

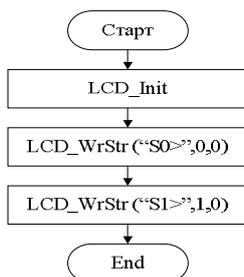


Рис. 4.14. WH2004L_Start (старт индикатора WH2004L)

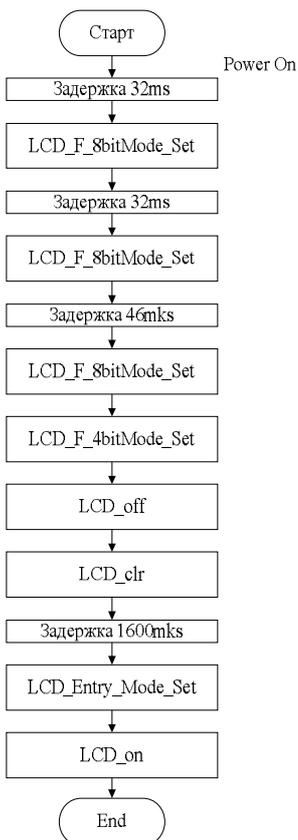


Рис. 4.15. LCD_Init (инициализация ЖК-модуля)

Заключение

В результате проекта разработан макет действующего конвертора интерфейсов «последовательный порт – параллельный интерфейс ЖК-модуля» на микроконтроллере C8051F410 от SiLabs, который позволяет отказаться от сложного параллельного интерфейса в дальнейших разработках. При помощи данного модуля можно создавать удаленные элементы индикации с использованием интерфейсов RS – 422/485, RS – 232. В дальнейших разработках данный модуль можно упростить до мобильных габаритов и использовать как самостоятельный блок, который может устанавливаться в технику, переносные устройства и т.д. Главное преимущество этого проекта в том, что устройство поддерживает любой стандартный алфавитно-цифровой жидкокристаллический индикатор на базе контроллера HD44780/KS0066 и доступность перепрограммирования с помощью персонального компьютера через USB при наличии отладочного программного обеспечения.

Список литературы

1. Жидкокристаллические индикаторы фирмы Winstar. [Электронный ресурс]. URL: http://www.compitech.ru/html.cgi/arhiv/01_06/stat_26.htm.
2. Статьи, обзоры. [Электронный ресурс]. URL: <http://www.efo.ru/doc/Silabs/Silabs.pl?2090>.
3. Хоровиц П., Хилл У.Г. Искусство схемотехники. 7-е изд. – М.: Мир, Бином пресс, 2009. – С. 704.
4. Пахомов Б.И. Самоучитель C/C++ и Borland C++ Builder. – Спб.: БХВ-Петербург, 2006.
5. Дарахвелидзе П. Г., Марков Е. П. Программирование в Delphi 7. – Спб.: БХВ-Петербург, 2003.