

ЛитерА LA6-12.2

Контроллер для первоначального обучения

Краткое описание

1. Состав.

Плата состоит из следующих электронных компонентов:

- микропроцессор;
- 56 светодиодов (7 независимых портов по 8 бит каждый);
- 4 реле (On-Off);
- 6 электронных ключей (+5В);
- 4 кнопки (замыкание на ноль);
- динамик для воспроизведения звука;
- модуль связи с ПЭВМ по шине USB;
- винтовые клеммники для соединений с внешними устройствами.

Для взаимодействия с внешними устройствами, таким образом, можно воспользоваться:

реле — для управление мощными нагрузками (с полной гальванической развязкой);

ключами — для управление маломощными нагрузками;

кнопками — для получения сигнала от концевых датчиков, кнопок и т. п.

2. Основные технические характеристики.

| <i>Характеристика</i> | <i>Значение</i> |
|---|---|
| Источник напряжения питания | - от USB - внешнее |
| Напряжение внешнего источника питания, В | +9...+18В |
| Защита от переплюсовки подключения внешнего ИП | есть |
| Максимальный ток потребления, мА | 400 ограничен предохранителем |
| Количество реле управления | 4 |
| Режим коммутации каждого реле | частота < 2 Гц напряжение < 60В ток < 1А |
| Количество ключей управления | 6 |
| Режим коммутации каждого ключа | частота < 1 кГц напряжение = 5В ток < 50 мА |
| Количество подключаемых внешних кнопок контроля (концевых выключателей) | 4 |
| Режим коммутации каждой кнопки | замыкание (U=5В I>1мА) |
| Связь с ПЭВМ | USB |

2. Перечень используемых объектов.

Для использования доступны следующие объекты:

- порт, состоящий из 7-ми объектов (порт_1 ... порт_7), каждый из которых состоит из 8-ми объектов (8 бит), состояние которых (0/1) индицируется светодиодом;
- реле (реле_1...реле_4), состояние каждого включено / выключено;
- ключ (ключ_1...ключ_6), состояние каждого включен / выключен;
- звук, состояние включен (воспроизводится нота) / выключен;
- кнопка (кн_1...кн_4), состояние каждой нажата (включена) / отжата (выключена);
- переменная А, возможные значения = 0...255;
- переменная В, возможные значения = 0...255.

3. Программирование контроллера.

3.1 Установка программного обеспечения LitterA LA612.

ОС Windows XP / 7 / 8

1. Программное обеспечение не нуждается в установке (portable) и может располагаться в любом месте. Для начала работы — просто стартуйте файл LitterA_LA612v20.exe из рабочего каталога.
2. Для взаимодействия с платой необходимо наличие виртуального COM-порта, который устанавливается в операционную систему в виде драйвера.
(см. установочный файл CP210x_VCP_Win_..xx.. .exe, где xx — версия ОС).
После установки драйвера необходимо задать номер порта для чего:
 - подключить кабель USB к ПЭВМ и плате «LitterA»;
 - в свойствах «Мой компьютер» выбрать «Диспетчер оборудования» и далее в перечне устройств в разделе «Порты COM/ LPT» найти виртуальный COM-порт «Silicon Labs CP210x UART USB Bridge» и открыть его «Свойства». Далее перейти на вкладку «Параметры порта» - «Дополнительно» и установить имя порта «COM1» или «COM2» или «COM3». Подтвердить сделанные изменения.
3. После старта программы «LitterA_LA612v20» убедиться, что связь с ПЭВМ установлена. Если этого не произошло, следует нажать кн. «Настройки» и выбрать то же имя COM-порта, которое было установлено в настройках виртуального COM-порта в диспетчере устройств.

ОС Linux Ubuntu / Mint

1. Подключить кабель USB к ПЭВМ и плате «LitterA».
2. В консоле дать команду `dmesg | tail -1` и посмотреть к какому номеру ttyUSB произошло подключение устройства Silicon Labs CP210x.
3. В консоле дать команду `ln -s /dev/ttyUSBX ~/.wine/dosdevices/com1` где X — ранее установленный номер.
4. В консоле дать команду `sudo chmod 777 -v /dev/ttyUSBX` где X — тот же установленный номер.
5. Стартовать программу LitterA.exe в окружении Wine.
6. Убедиться, что связь с ПЭВМ установлена. Если этого не произошло, следует нажать кн. «Настройки» и выбрать имя COM-порта = COM1.

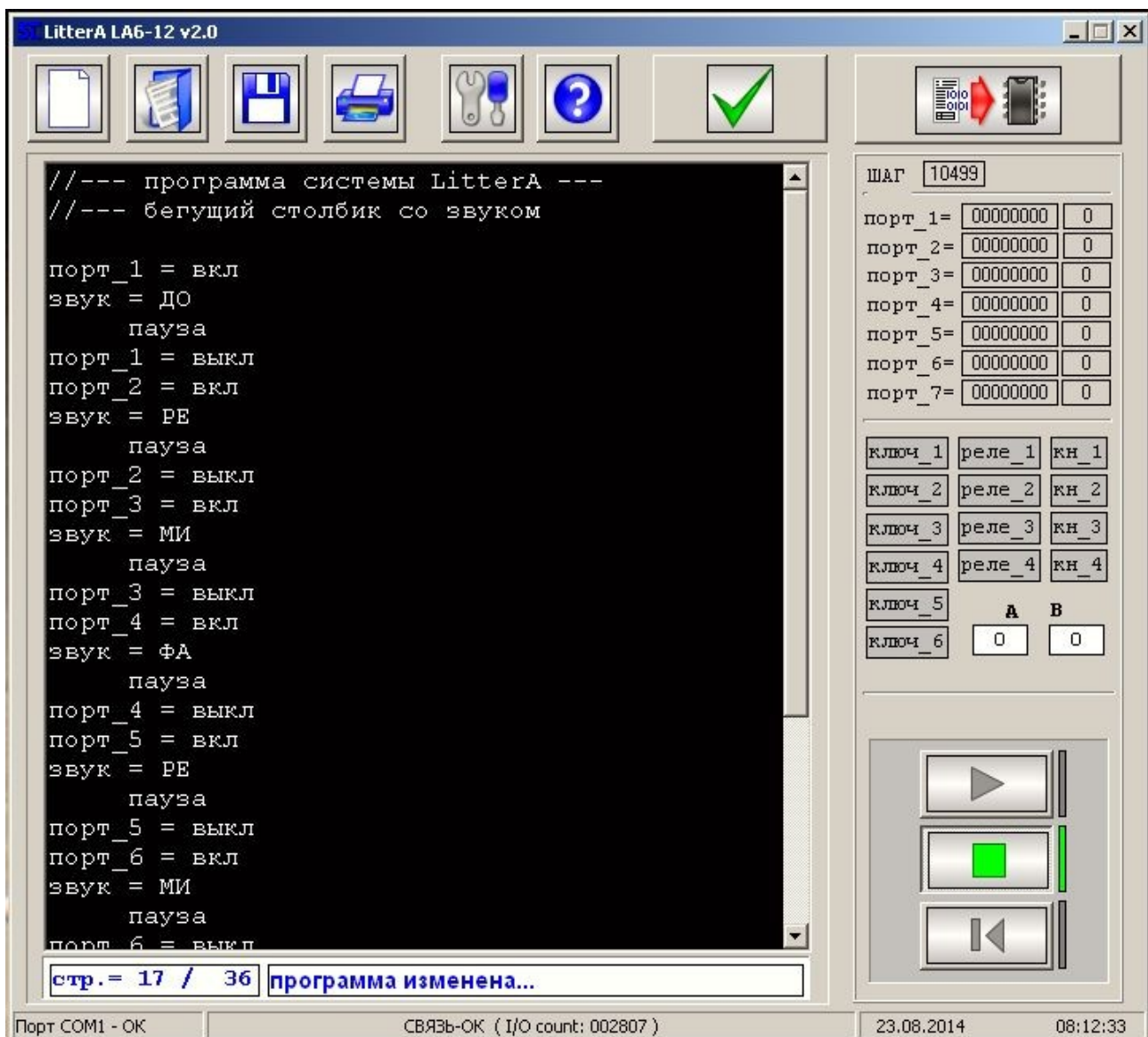
3.2 Режимы работы ПО LitterA.

Программа имеет два режима работы: режим редактора и режим консоли управления.

Редактор — предназначен для ввода текста программы, ее проверки и загрузки на выполнение в плату контроллера. Имеет стандартные функции: открыть / сохранить файл, распечатать файл. Формат файлов программ — текстовой (.txt). Таким образом, программы могут быть созданы и отредактированы любым текстовым редактором.

В правой части окна редактора имеется панель отладки, куда выводится информация в реальном времени о состоянии всех объектов контроля и управления. Панель доступна только при наличии связи с платой контроллера.

Редактор имеет возможность конфигурирования: цвет фона, цвет текста, номер порта связи.



вид окна Редактор

Описание кнопок управления (слева направо / сверху вниз):

- создать новый — очистка окна для ввода новой программы;
- открыть файл
- сохранить файл
- распечатать файл
- конфигурация — открытие окна для задания конфигурации редактора;
- помощь (F1) — открытие окна с кратким справочником команд;
- проверить (F5) — обязательная команда перед отправкой программы в плату контроллера на исполнение. Результат проверки выводится в строку сообщений.
- загрузить (F8) — команда для загрузки программы в плату контроллера на исполнение, может быть выполнена только после проверки программы;
- кнопки «Старт» / «Пауза» / «Исходное» — предназначены для управления ходом исполнения программы, загруженной в плату контроллера. Дублируют кнопки управления, расположенные на плате.

Справа у каждой кнопки расположен индикатор состояния, указывающий какой статус команда управления реально имеет в плате контроллера.

«Старт» - начать исполнение программы, «Пауза» - временный останов программы, «Исходное» - прекращение выполнения программы, все объекты приводятся в выключенное состояние.

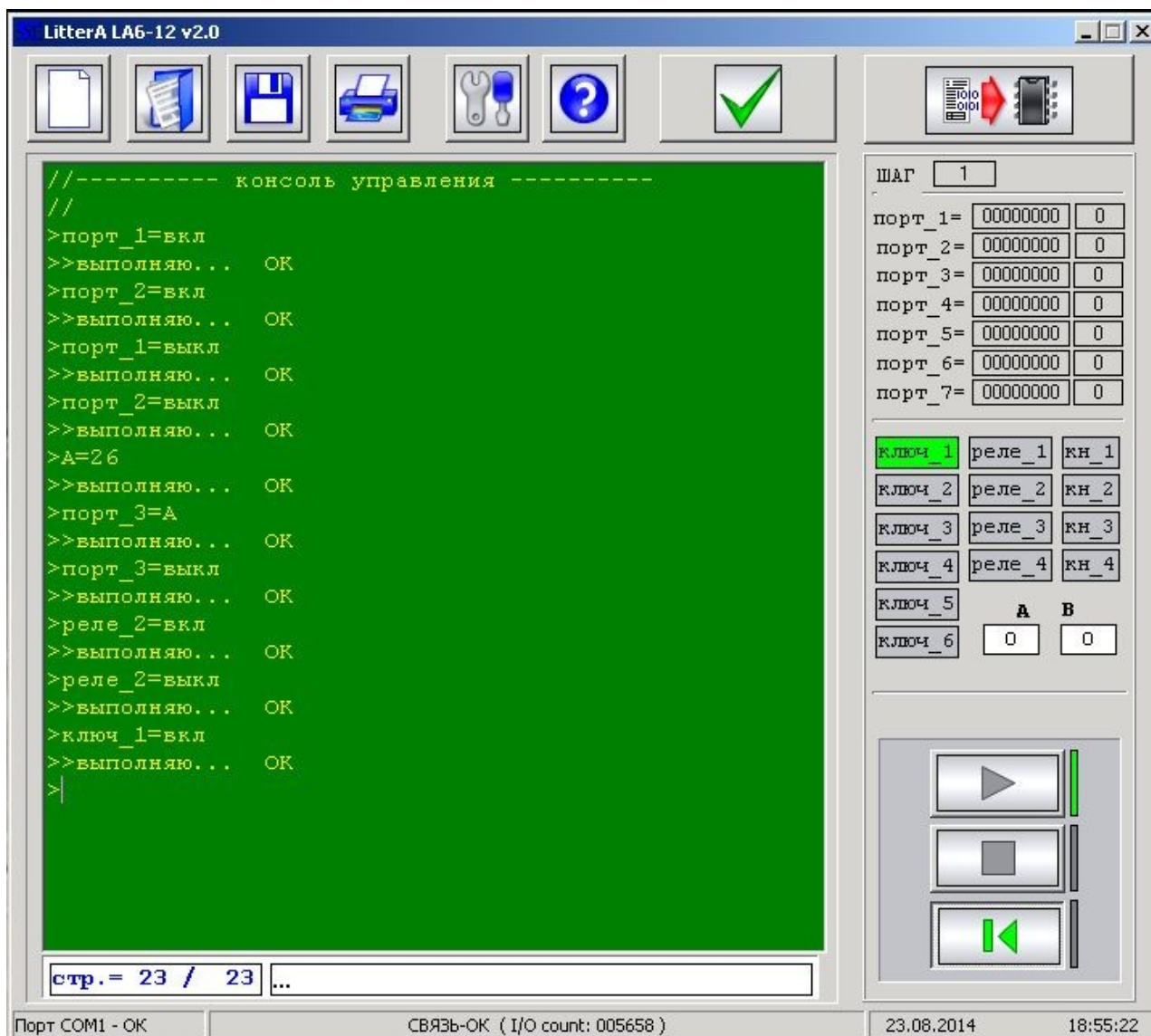
Основные требования к тексту программ:

- длина строки, включая пробелы и табуляцию — не более 128 символов;
- признак комментария — два символа // ;
- все команды вводятся маленькими буквами;
- все параметры команд допустимо вводить маленькими или заглавными буквами;
- всего строк программы — не более 1000.

Консоль управления — режим предназначен для проведения отладочных работ.

Переключение между редактором и консолью — кн. F10.

Команды управления объектами вводятся обычным образом и после нажатия Enter немедленно исполняются.



вид окна Консоль

Переключение в режим Консоль может быть выполнено в любой момент, при этом плата контроллера автоматически будет переведена в состояние «Исходное».

3.3 Перечень используемых команд.

3.3.1 Команды управления объектами.

Объект порт

порт = вкл / порт = выкл — вкл / выкл сразу все светодиоды

порт_x = вкл / порт_x = выкл — вкл/выкл все 8 светодиодов (8 бит) порта x=1...7

порт_x.y = 1 / порт_x.y = вкл — включить один светодиод порта (бит=1) y=1...8

порт_x.y = 0 / порт_x.y = выкл — выключить один светодиод порта (бит=0)

порт_x = 00010001 — установить соответствующие биты порта x в 0 или 1 (соответственно, включить или выключить светодиоды). Нумерация бит слева направо.

Например порт_1 = 00001111

будут выключены светодиоды 1-й/2/3/4-й и включены 5-й/6/7/8-й.

Объект реле

реле_x = вкл / реле_x = выкл — включить / выключить реле x=1...4

На плате контроллера используются механические герконовые реле, контактам которых для замыкания / размыкания необходимо время несоизмеримо большее, времени выполнения одной команды при исполнении программы. При написании программы, поэтому, необходимо контролировать, чтобы между командой реле_x = вкл и командой реле_x = выкл был промежуток времени не менее 0,5 сек. (см. команда Пауза).

Объект ключ

ключ_x = вкл / ключ_x = выкл — включить / выключить ключ x=1...6

Объект звук

звук = до - включить динамик и воспроизвести ноту До (ре,ми,фа, соль,ля,си)

звук = до+ - включить динамик и воспроизвести ноту До на октаву выше
(ре+,ми+,фа+, соль+,ля+,си+)

звук = выкл - выключить динамик

3.3.2 Команды управления скоростью выполнения команд.

Контроллер при выполнении загруженной программы выполняет команды одну за другой последовательно. Время выполнения любой команды составляет очень малую величину (менее 1/1000 секунды). Поэтому, часто, для наблюдения результатов необходимо иметь возможность выполнения команд через заданный промежуток времени.

Например

```
порт_1 = вкл  
порт_1 = выкл
```

в данном примере команда на выключение выполнится сразу же за командой на включение,и, как следствие — будет казаться, что порт_1 не включался вовсе.

| | |
|------------------|---|
| пауза | сделать паузу до выполнения следующей команды в 1 секунду |
| пауза+ | сделать паузу до выполнения следующей команды в 0,1с (100 мс) |
| пауза++ | сделать паузу до выполнения следующей команды в 0,2с (200 мс) |
| ... | |
| пауза++++ | сделать паузу до выполнения следующей команды в 0,5с (500 мс) |

Например

```
порт_1 = вкл  
пауза  
порт_1 = выкл
```

включить светодиоды порта 1 и через 1 секунду выключить.

3.3.2 Команды выполнения арифметических операций.

Для простых арифметических операций существуют две переменные: А и В.
ВАЖНО: допустимые значения переменных = 0...255.

| | | |
|----------------------|---------------------|-----------------------------------|
| A=значение | B=значение | — присвоить значение переменной |
| A=A+значение | B=B+значение | — сложить переменную со значением |
| A=A-значение | B=B-значение | — вычесть значение из переменной |
| A=A+B / B=B+A | | — сложить переменные |
| A=A-B / B=B-A | | — вычесть переменные |
| A=B / B=A | | — приравнять переменные |

ВАЖНО: при наборе команд арифметики — пробелы не допускаются!

3.3.3 Команды управления ветвлением программы.

| | | |
|------------------------|---|---|
| метка_х: | - | — установить в программе точку для перехода $x=1...15$ |
| переход метка_х | | — перейти на метку (в точку перехода) |
| возврат | | — перейти на команду, следующую за последней выполненной командой «переход метка_х» |

Например

```
порт = выкл
звук = выкл
переход метка_2
метка_1:
порт_7.8 = вкл
звук = ДО
пауза++
порт_7.8 = выкл
звук = выкл
возврат
метка_2:
порт_1 = вкл
переход метка_1
порт_1 = выкл
порт_2 = вкл
переход метка_1
порт_2 = выкл
порт_3 = вкл
переход метка_1
порт_3 = выкл
порт_4 = вкл
переход метка_1
порт_4 = выкл
порт_5 = вкл
переход метка_1
...
```

*после старта программа выключит порт и звук и перейдет в точку метка_2: включит порт_1 и перейдет в точку метка_1: включит порт_7.8, включит звук, подождет 0,2 сек, выключит порт_7.8, выключит звук и выполнит команду **возврат**, т. е. перейдет в точку «порт_1 = выкл».*

Если программу запустить на исполнение, то мы увидим следующий результат: столбик светодиодов будет двигаться вправо, при этом каждое перемещение будет сопровождаться коротким звуковым сигналом и вспыхиванием светодиода в нижнем правом углу.

3.3.4 Команды проверки условия.

Команды проверки условия имеют одинаковый синтаксис:

если условие **тогда** команда

Условием может быть :

| | | | |
|----------------------|----------------------|-------------------------------|---------|
| кн_x==вкл | кн_x==выкл | — кнопка включена / выключена | x=1...4 |
| A==значение | B==значение | — переменная равна значению | |
| A>значение | B>значение | — переменная больше значения | |
| A<значение | B<значение | — переменная меньше значения | |

ВАЖНО: при проверке условий равенства вводится двойной знак равно!

Командой может быть любая команда, рассмотренная выше.

Например

если V==0 тогда A=A+1

если кн_1==вкл тогда ключ_1 = вкл

если V>100 тогда V = 0

если кн_3==выкл тогда порт_1 = вкл

3.3.5 Команды консоли управления.

В качестве команд могут быть команды управления портами, ключами, реле и звуком.

Дополнительные команды

ППЗУ = записать записать в микросхему ППЗУ программу, которая загружена в плату. После включения питания платы, программа будет считана из ППЗУ и может быть сразу же исполнена (ком. Старт). Это позволяет быть плате автономной и работать без участия ПЭВМ. (ППЗУ — перезаписываемое постоянное запоминающее устройство).

ППЗУ = стереть очистить микросхему ППЗУ

сброс выполнить полный перезапуск платы

очистить очистить окно консоли управления

4. Основной алгоритм работы.

1. Загрузить ранее сохраненную программу в окно редактора команд или составить новую программу.
2. Выполнить проверку программы.
3. Загрузить программу в плату.
4. Выдать ком. Старт.
5. При необходимости воспользоваться консолью и записать программу в ППЗУ платы (теперь при выключении питания платы программа не будет утеряна).