

Интеллектуальное освещение

Олег Вальпа

Приведено описание, разработанной автором статьи системы интеллектуального управления освещением вивария для животных, созданное на базе панели оператора Weintek и модуля вывода.



Дополнительные материалы к статье вы можете скачать с нашего сайта по этой ссылке

Введение

В современном мире существуют такие замечательные устройства, как панели оператора, которые представляют собой компактные микропроцессорные пульта управления с цветным сенсорным графическим экраном, внутренними энергонезависимыми часами реального времени и несколькими последовательными интерфейсами типа RS-232, RS-485, USB и Ethernet. Такие панели можно встраивать в двери шкафов автоматики или непосредственно в стену помещения. Благодаря производительному процессору и большому объёму памяти эти панели оператора позволяют одновременно служить человеко-машинным интерфейсом и выполнять различные функции автоматического управления технологическими процессами.

Производятся данные панели компаниями многих стран, в том числе и дружественными для России. Сенсорный экран панели может иметь размеры: 4,3", 7", 10", 15" и т.д. Внешний вид панелей оператора представлен на рис. 1.

Стоимость панелей зависит от размера сенсорного экрана и составляет от нескольких десятков до нескольких сотен долларов, что вполне позволяет им конкурировать в области автоматизированных систем. К одним из самых распространённых панелей относятся панели компаний: Weintek [1], Kinco [2], Leadshine [3] и др.

Проект

Рассмотрим реализацию одной из автоматизированных систем управления с применением панели оператора на конкретном примере. Для большинства вивариев, содержащих подопытных животных, таких как мыши, кролики, морские свинки и т.п., необходимы системы искусственного освещения с функциями плавного рассвета и заката. Такие системы исключают стрессовые ситуации для животных и приближают их жизнь к естествен-

ному существованию. Данную систему управления вполне можно реализовать на базе панели оператора, которая одновременно будет служить и человеко-машинным интерфейсом, и вычислительно-управляющим устройством. В качестве панели оператора применена 7-дюймовая модель MT8071iE компании Weintek с бесплатно предоставляемой средой разработки EasyBuilder Pro.

В качестве осветителя применён потолочный светодиодный светильник с регулированием яркости управляющим напряжением от 0 до

10 вольт. В отличие от систем с интерфейсом DALI, данная система значительно дешевле и доступнее.

Поскольку панель оператора имеет только последовательные выходные интерфейсы, дополнительно будет использован модуль вывода с аналоговым выходным сигналом от 0 до 10 вольт, подключаемый к панели оператора по интерфейсу RS-485. Управление таким модулем от панели оператора будет производиться с применением стандартного и широко распространённого промышленного протокола Modbus RTU. В каче-



Рис. 1. Внешний вид панелей оператора

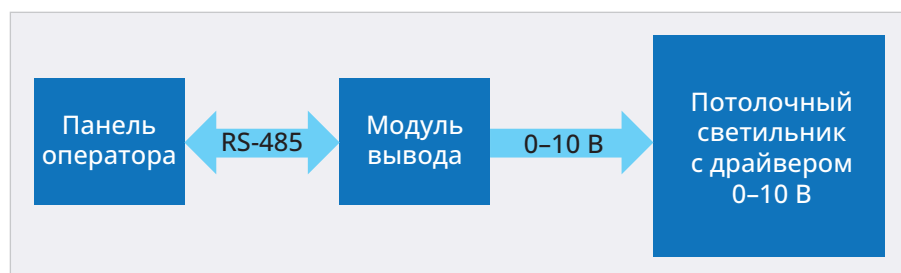


Рис. 2. Структурная схема системы управления

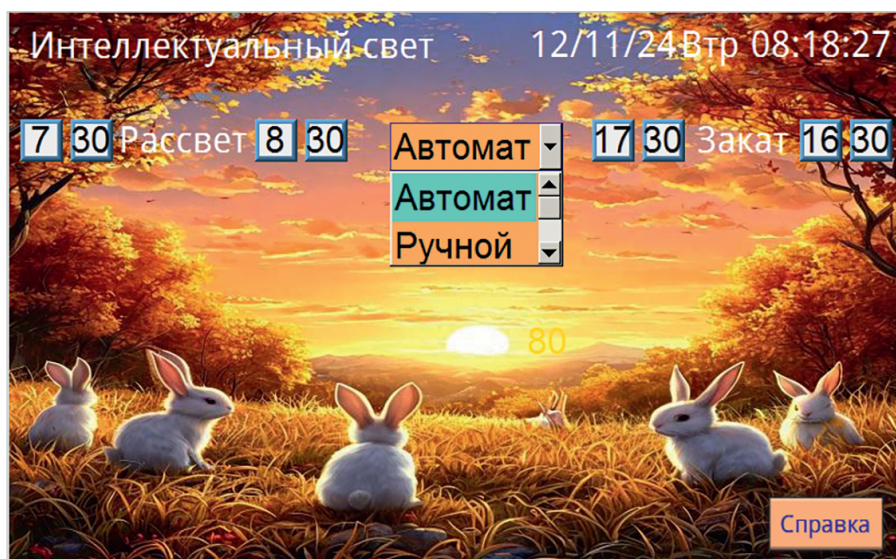


Рис. 3. Внешний вид главного экрана



Рис. 4. Главный экран среды разработки EasyBuilder Pro

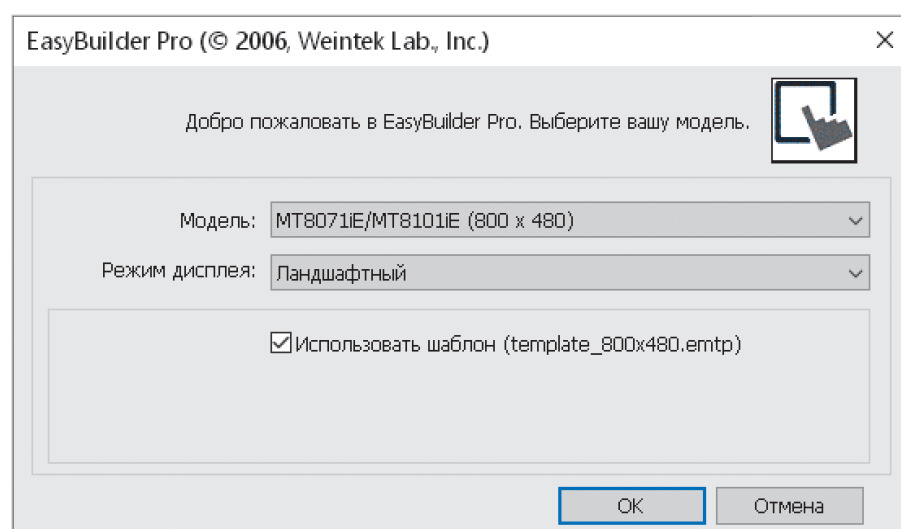


Рис. 5. Окно выбора модели панели оператора

Среда разработки

Внешний вид главного экрана системы управления, получившей название «Интеллектуальный свет», создан в среде разработки EasyBuilder Pro и представлен на рис. 3.

На этом экране кроме названия системы отображаются календарь с часами текущего времени, значения установок для начала и окончания рассвета и заката, элемент выбора режима работы системы, цифровой индикатор уровня световой яркости, ползунок для настройки яркости света в ручном режиме и кнопка вызова справочного окна. Кроме того, на экран помещён рисунок с кроликами на природе для придания всему экрану программы красочного и эстетического внешнего вида.

Алгоритм работы программы простой, он производит сравнение текущего значения времени со значениями установок рассвета и заката. До начала рассвета управление светом имеет нулевой уровень, а после каждую минуту увеличивается до максимума, с дискретностью, равной отношению максимального значения к времени интервала рассвета. Далее сигнал управления сохраняет максимальное значение до начала заката, после чего плавно уменьшается до нулевого значения с дискретностью, равной отношению максимального значения к времени интервала заката. Таким образом, при изменении оператором времени начала или окончания рассвета и заката система автоматически будет формировать плавный рассвет и закат с заданным интервалом. При переключении системы в ручной режим, например, для уборки, ремонта или профилактических работ, система полностью передаёт управление оператору, который может произвольно регулировать освещённость вивария с помощью движка на экране панели оператора.

Процесс разработки программы для панели оператора представляет собой интересное и увлекательное творчество, которое сочетает в себе создание красочного интерфейса с органами управления, связываемыми настройками среды разработки.

После установки среды разработки на персональный компьютер в каталоге среды присутствует папка «project» с множеством готовых масштабных проектов для самых разных

стве готового модуля вывода можно использовать любой подходящий модуль, например, от компании КИП-Сервис [4], Контэл [5], IECON [6] и др. Кроме того, производители панелей

операторов также производят модули ввода-вывода, и в проекте можно использовать любой из них.

Структурная схема системы управления приведена на рис. 2.

сфер автоматизации. Изучение этих примеров поможет в ускорении освоения программирования панелей оператора.

Главный экран среды разработки EasyBuilder Pro представлен на рис. 4.

В центре находится поле для элементов главного экрана программы, на который размещают готовые библиотечные элементы текстовых меток, кнопок, индикаторов, часов с календарем, фоновых рисунков и т.п. из закладок «Объекты» и «Рисование». Слева отображается карта памяти панели оператора, состоящая из ячеек оперативной памяти LW, оперативной битовой памяти LB и постоянной энергонезависимой памяти RW, используемой в основном для установок. Также в левой области можно открыть вкладку окон программы и добавлять с её помощью новые окна. Справа отображается библиотека рисунков для разработки программ. Данную библиотеку можно свободно редактировать путём удаления и добавления рисунков. Сверху находятся главное меню среды разработки и панель с кнопками управления проектом.

Создание новой программы начинается с помощью меню: **Файл** → **Создать** или путём нажатия компьютерных клавиш **Ctrl+N**. При этом откроется окно выбора модели панели оператора (рис. 5).

В дальнейшем модель панели оператора можно будет изменить или трансформировать, и после компиляции проект будет нормально работать на другой модели.

После выбора модели откроется окно системных настроек, с помощью которых добавляется модуль вывода, выбирается номер последовательного порта панели для связи с добавленным модулем и определяются другие необходимые параметры панели оператора при необходимости. Данное окно приведено на рис. 6.

Модуль вывода можно выбрать после нажатия кнопки «Создать» из окна параметров устройства, приведённого на рис. 7. Если ни одно устройство не подходит для проекта, можно выбрать Free Protocol и в дальнейшем подстроить его под любой модуль вывода.

Далее открывшееся пустое поле заполняется необходимыми элементами, описанными ранее, и произво-

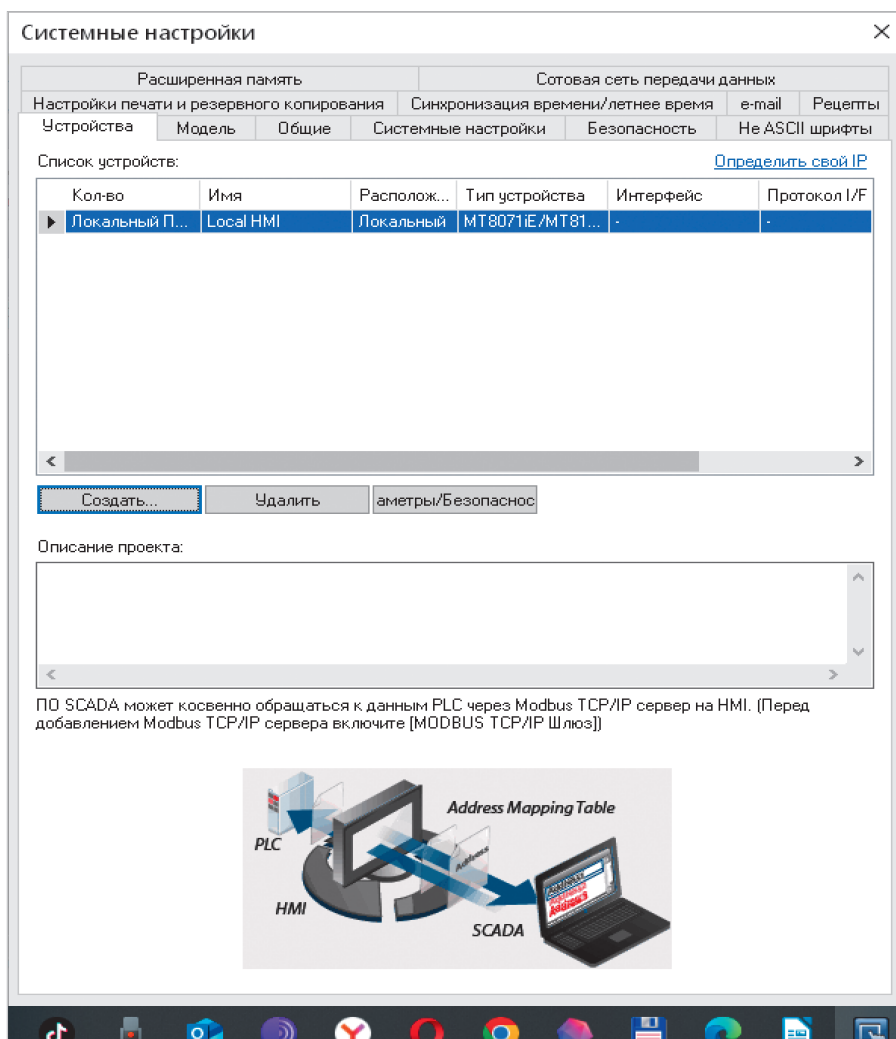


Рис. 6. Окно системных настроек панели оператора



Рис. 7. Окно параметров устройства панели оператора

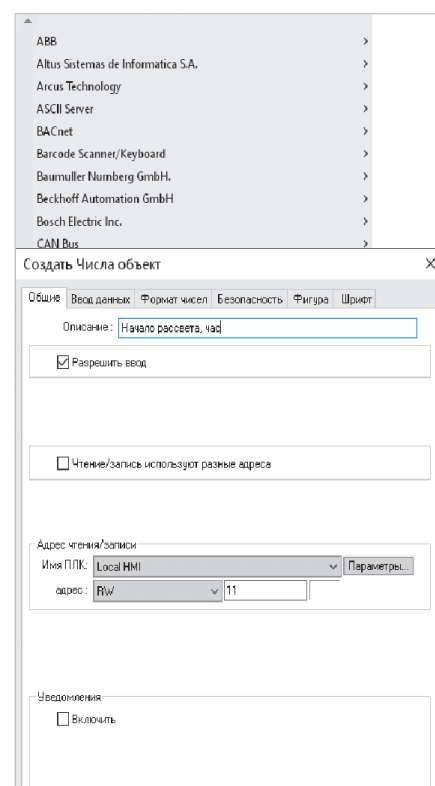


Рис. 8. Окно атрибута числового элемента панели оператора

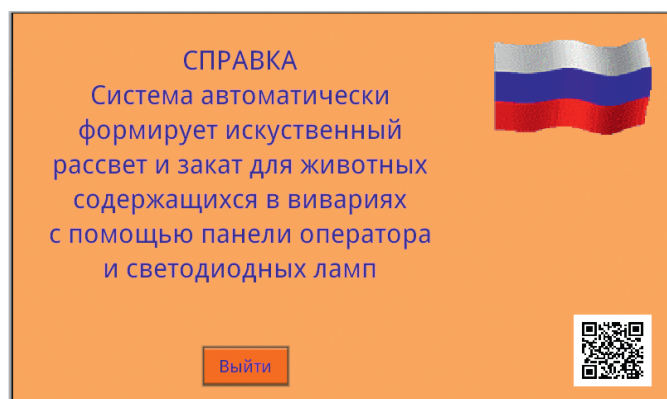


Рис. 9. Окно справки программы

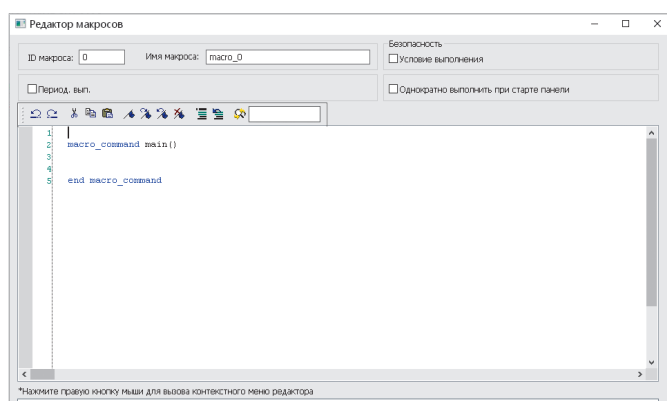


Рис. 10. Окно макросов панели оператора

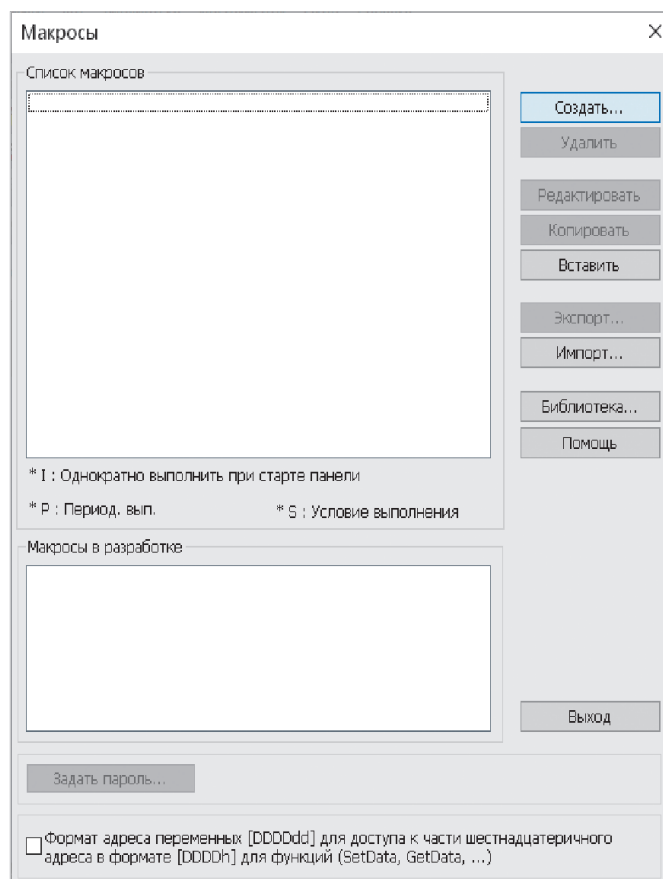


Рис. 11. Окно редактора макросов панели оператора

дится настройка некоторых из них путём привязки к определённому адресу памяти панели оператора через атрибуты элемента. В представленном здесь проекте числовые элементы установок начала и окончания рассвета в часах и минутах привязаны последовательно к ячейкам энергонезависимой памяти RW 11, 12, 13 и 14. Числовые элементы установок начала и окончания заката в часах и минутах привязаны последовательно к ячейкам памяти RW 21, 22, 23 и 24, соответственно. Пример такой привязки показан на рис. 8.

Элемент выбора режима работы привязан к ячейке памяти LW0, а движок и числовой элемент уровня освещённости привязаны к одной ячейке памяти LW1.

Первичную проверку работы программы можно проверить уже на этом этапе путём запуска программы с помощью меню: **Инструменты** → **Офлайн-симуляция** или клавишами **Ctrl+T**. При этом на экране монитора появится главное окно программы, и в ручном режиме изменения положения движка будут приводить к изменению цифрового индикатора освещённости от 0 до 100 процентов.

Справочное окно программы создаётся в левом поле среды разработки, на закладке «Окна», ниже названия главного окна. Название нового окна можно задать произвольным. После выбора нового окна на него аналогично размещаются необходимые элементы из закладок «Объекты» и «Рисование». Окно справки программы приведено на рис. 9.

Макрос

Для завершения разработки программы необходимо создать макрос, который будет каждую секунду производить операции, описанные в алгоритме работы программы, и отправлять вычисленное значение освещённости в модуль вывода через последовательный порт RS-485. Создание макроса начинается с помощью меню **Инструменты** → **Макросы** или клавишами **Ctrl+M**. При этом на экране монитора откроется окно макросов, показанное на рис. 10.

Далее необходимо нажать кнопку «Создать», и среда разработки автоматически сформирует заготовку макроса с именем `macro_0` в окне редактирования, представленного на рис. 11.

Данный редактор содержит во вкладке «Функции» набор всех функций, необходимых для реализации любых программных алгоритмов, встроенную справочную помощь и кнопки инструментов разработки с отладкой. Код программы макроса, разработанный автором статьи, приведён в листинге 1.

Листинг 1. Код программы макроса

```
macro_command main() // Главный макрос
// Определение переменных
short mode, s, ao // Режим, сигнал регулировки и выходной сигнал
// Текущее время, начало и конец рассвета и заката в часах и минутах
short tvh, tvh, nrh, nrm, krh, krm, nzh, nzm, kzh, kzm
// Текущее время, начало и конец рассвета и заката в суммарных минутах
short tv, nr, kr, nz, kz
short dr, dz // Длительность рассвета и заката
GetData(mode, "Local NMI", LW, 0, 1) // Читать режим работы: 0-Автомат 1-Ручной
```

```

GetData(nrh, "Local HMI", RW,
11, 1) // Читать часы начала
рассвета
GetData(nrm, "Local HMI", RW,
12, 1) // Читать минуты начала
рассвета
GetData(krh, "Local HMI", RW,
13, 1) // Читать часы конца
рассвета
GetData(krm, "Local HMI", RW,
14, 1) // Читать минуты конца
рассвета
GetData(nzh, "Local HMI", RW,
21, 1) // Читать часы начала
заката
GetData(nzm, "Local HMI", RW,
22, 1) // Читать минуты начала
заката
GetData(kzh, "Local HMI", RW,
23, 1) // Читать часы конца за-
ката
GetData(kzm, "Local HMI", RW,
24, 1) // Читать минуты конца
заката
GetData(tvm, "Local HMI", LW,
9018, 1) // Читать минуты теку-
щего времени
GetData(tvh, "Local HMI", LW,
9019, 1) // Читать часы текуще-
го времени
// Вычислить суммарное значение
всех времен в минутах
tv = tvm + tvh*60
nr = nrm + nrh*60
kr = krm + krh*60
nz = nzm + nzh*60
kz = kzm + kzh*60
// Вычислить длительность вре-
мени рассвета и заката в мину-
тах
dr = kr - nr
if dr <= 0 then // Защита от не-
верной настройки рассвета
dr = 1
end if
dz = kz - nz
if dz <= 0 then // Защита от не-
верной настройки заката
dz = 1
end if
if mode==0 then // Если режим
0-Автомат
if (tv <= nr) then // Если те-
кущее время до начала рассвета
s = 0 // Минимальный сигнал ре-
гулировки = 0%
else
if (tv < kr) then // Если теку-
щее время до конца рассвета
s = 100*(tv-nr)/dr // Вычислить
сигнал регулировки
else

```

```

if (tv <= nz) then // Если те-
кущее время до начала заката
s=100 // Максимальный сигнал
регулировки = 100%
else
if (tv <= kz) then // Если те-
кущее время до конца рассвета
s = 100-100*(tv-nz)/dr
else
s = 0 // Минимальный сигнал ре-
гулировки = 0%
end if // Конец проверки (tv <=
kz)
end if // Конец проверки (tv <=
kr)
end if // Конец проверки (tv <=
nr)
SetData(s, "Local HMI", LW, 1,
1)
else // Иначе ручной режим
end if // Конец проверки mode==0
// Преобразование величины сиг-
нала в шкалу ЦАП модуля вывода
от 0 до 1023
ao=s*1023/100
SetData(ao, "Local HMI", LW, 2,
1) // Запись сигнала регулятора
в память панели оператора
// Отправка сигнала регулирова-
ния в модуль вывода
SetDataEx(ao, "MODBUS RTU", 4x,
1, 1)
end macro_command

```

Код программы снабжён подробны-
ми комментариями, которые позволя-
ют легко понять назначение всех её
строк.

Номера ячеек памяти LW 9018 и LW
9019 для внутренних часов времени
панели оператора находятся в библи-
отеке адресных меток через меню:
Библиотека – Адрес.

После копирования данного кода
в макрос необходимо выполнить его
сохранение и компиляцию соответ-
ствующей кнопкой редактора. Кро-
ме того, требуется задать регуляр-
ную, ежесекундную периодичность
выполнения макроса в виде записи
10×100 ms.

Теперь программа полностью гото-
ва к работе, и её можно загрузить в
панель оператора. Загрузку можно
осуществить с помощью USB-флешки
или через порт Ethernet, предвари-
тельно назначив IP-адрес для панели
оператора через системное меню. Вход
в это меню производится с помощью
нажатия стрелки в правом нижнем

углу включённой панели оператора с
последующим выбором кнопки-шесте-
рёнки и вводом пароля по умолчанию
111111.

Для загрузки программы с помо-
щью USB-флешки необходимо
выполнить подготовку загрузочного
кода программы с помощью меню:
Инструменты – Генерация данных
для USB-диска, предварительно под-
ключив чистую отформатирован-
ную в FAT32 флешку к компьютеру,
на котором выполняется разработ-
ка программы. При необходимости
загрузочный код можно сохранить и
на жёстком диске компьютера с после-
дующим переносом его на флешку.
Далее флешку с загрузочным кодом
подключают к USB-порту включён-
ной панели оператора и ожидают
появления окна загрузки, в котором
необходимо нажать кнопку Download.
После чего панель оператора запросит
пароль, который по умолчанию имеет
значение 111111, и отобразит на экра-
не каталог флешки. Данный каталог
необходимо развернуть с помощью
кнопок «+» и выбрать загрузочный
файл. Далее панель оператора автома-
тически выполнит загрузку и запустит
программу. Теперь панель подклю-
чается в схему системы управления,
и автоматический процесс управле-
ния освещением начинается.

Наличие Ethernet-порта в панели опе-
ратора и поддержка встроенного про-
токола Modbus TCP позволяют подклю-
чить её к системе диспетчеризации. Для
этого назначается IP-адрес для панели
оператора через её системное меню.

Файл проекта размещён на сайте
журнала [7] и может быть использован
для дальнейшего развития. Данный
проект можно развивать, например,
с помощью добавления новых окон с
графиками, журналом событий и т.п.

Литература

1. URL: <https://www.weintek.com/globalw/>.
2. URL: <https://en.kinco.cn/product/hmi.html>.
3. URL: <https://leadshine.com/products/control/hmi.html>.
4. URL: https://kipservis.ru/seneca/modul_analogovogo_signala-z-3ao.htm.
5. URL: https://kontel.ru/index.php?route=product/category&path=155_198.
6. URL: <https://www.iecon.ru/catalog/moduli-vvoda-vyvoda/ebm-c/>.
7. URL: <https://www.cta.ru>.

