

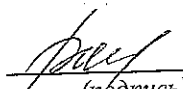
Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

Работа выполнена в СКБ «Промышленная робототехника»

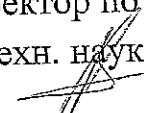
СОГЛАСОВАНО

Начальник отдела ОНиПКРС

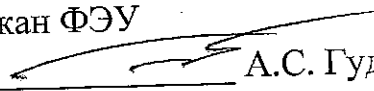

(подпись) Е.М. Димитриади
« 19 » 06 20 23 г.

УТВЕРЖДАЮ

Проректор по научной работе,
д-р техн. наук, профессор


(подпись) А.В. Космынин
« 19 » 06 20 23 г.


Декан ФЭУ


(подпись) А.С. Гудим
« 19 » 06 20 23 г.

Аппаратно-программный комплекс
«Микроконтроллерная система управления фотоэлектрической
энергоустановкой»

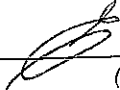
Комплект конструкторской документации

Руководитель СКБ


19.06.2023
(подпись, дата)

С.И. Сухоруков

Руководитель проекта


20.12.2022
(подпись, дата)

В.А. Егоров

Карточка проекта

Название	Микроконтроллерная система управления фотоэлектрической энергоустановкой
Тип проекта	Тип проекта: учебная работа
Исполнители	И.Ю. Деркач, Н.Л. Минебаев
Срок реализации	14.09.2022- 31.12.2022

Использованные материалы и компоненты

QTY	PART-REFS	VALUE	CODE
Resistors			
6	R1-R6	4.7K	Digikey 311-4.7KETR-ND
Integrated Circuits			
1	U1	ATMEGA328P_32PIN	
1	U2	DS1307	
Diodes			
1	D1	LED-YELLOW	
Miscellaneous			
1	BAT1	3V	
1	LCD1	LM044L	
1	RV1	10k	
1	X1	32768Hz	
4	key		

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

ЗАДАНИЕ

на разработку

Выдано студентам: Минебаеву Н.Л., Деркач И.Ю. группы 0МРБ-1

Название проекта: Микроконтроллерная система управления фотоэлектрической энергоустановкой

Назначение: Слежение за Солнцем, с целью увеличения КПД фотоэлектрической энергоустановки

Область использования: альтернативная энергетика

Функциональное описание устройства:

Система вырабатывает управляющие воздействия на азимутальную и угломестную оси фотоэлектрической энергоустановки и осуществляет наведение установки на солнце, с целью увеличения её КПД.

Техническое описание устройства: Блок управления фотоэлектрической энергоустановки, содержащий: вычислитель координат Солнца по азимуту и углу месту на основе однокристального микроконтроллера; интерфейс для сопряжения с электроприводами азимутальной и угломестной осей установки; жидкокристаллический алфавитно-цифровой индикатор для отображения даты, времени и угловых координат Солнца; блок питания с выходным напряжением +5В.

Требования: точность наведения $\pm 1^\circ$.

План работ:

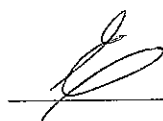
Наименование работ	Срок
Сбор и изучение материалов, необходимых для проектирования	09.2022
Разработка блок-схемы устройства	09.2022
Выбор элементов, разработка принципиальной схемы устройства	10.2022
Разработка программного кода и моделирование системы	10.2022
Изготовление прототипа устройства	11.2022
Тестирование и финальная отладка	12.2022
Оформление отчета	12.2022

Комментарии:

Перечень графического материала:

1. Блок-схема устройства _____
 2. Принципиальная схема устройства _____
 3. Блок схемы алгоритмов _____
 4. Внешний вид устройства _____
-
-
-
-
-
-

Руководитель проекта



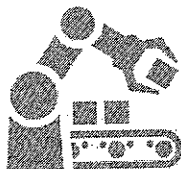
28.12.2022

В.А. Егоров

(подпись, дата)

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

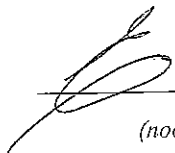


ПАСПОРТ

Аппаратно-программный комплекс


«Микроконтроллерная система управления
фотоэлектрической энергоустановкой»

Руководитель проекта

 28.12.2022
(подпись, дата)

В.А. Егоров

Ответственный исполнитель

 28.12.2022
(подпись, дата)

Н.Л. Минебаев

Комсомольск-на-Амуре 2022

Содержание

1	Общие положения	7
1.1	Наименование изделия	7
1.2	Наименования документов, на основании которых ведется проектирование системы	7
1.3	Перечень организаций, участвующих в разработке системы	7
1.4	Сведения об использованных при проектировании нормативно-технических документах.....	8
2	Назначение и принцип действия	9
2.1	Назначение изделия.....	9
2.2	Области использования изделия.....	9
2.3	Принцип действия	9
3	Состав изделия и комплектность.....	10
4	Технические характеристики.....	11
4.1	Основные технические характеристики блока	11
5	Устройство и описание работы изделия	12
5.1	Устройство изделия.....	12
5.2	Описание работы изделия	13
6	Условия эксплуатации	14
6.1	Правила и особенности размещения изделия	14
6.2	Меры безопасности	15
6.3	Правила хранения и транспортирования	15

1 Общие положения

Настоящий паспорт является документом, предназначенным для ознакомления с основными техническими характеристиками, устройством, правилами установки и эксплуатации устройства «Микроконтроллерная система управления фотоэлектрической энергоустановкой» (далее «МСУ ФОЭ»).

Паспорт входит в комплект поставки изделия. Прежде, чем пользоваться изделием, внимательно изучите правила обращения и порядок работы с ним. В связи с постоянной работой по усовершенствованию изделия, повышающей его надежность и улучшающей условия эксплуатации, в конструкцию могут быть внесены изменения, не отраженные в данном издании.

1.1 Наименование изделия

Полное наименование системы – аппаратно-программный комплекс «Микроконтроллерная система управления фотоэлектрической энергоустановкой» (МСУ ФОЭ).

1.2 Наименования документов, на основании которых ведется проектирование системы

Создание МСУ ФОЭ осуществляется на основании требований и положений следующих документов:

- задание на разработку.

1.3 Перечень организаций, участвующих в разработке системы

Заказчиком создания МСУ ФОЭ является Федеральное государственное бюджетное образовательное учреждение высшего образования «Комсомольский-на-Амуре государственный университет» (далее заказчик), нахо-

дящийся по адресу: 681013, Хабаровский край, г. Комсомольск-на-Амуре, Ленина проспект, д. 17.

Исполнителями работы по созданию МСУ ФОЭ являются студенты группы ОМРБ-1 И.Ю. Деркач, Н.Л. Минебаев.

1.4 Сведения об использованных при проектировании нормативно-технических документах

При проектировании использованы следующие нормативно-технические документы:

ГОСТ 2.001-2013. Единая система конструкторской документации. Общие положения.

ГОСТ 2.102-2013. Единая система конструкторской документации. Виды и комплектность конструкторских документов.

ГОСТ 2.105-95. Единая система конструкторской документации. Общие требования к текстовым документам.

ГОСТ 2.610-2006. Единая система конструкторской документации. Правила выполнения эксплуатационных документов.

ГОСТ 2.004-88. Единая система конструкторской документации. Общие требования к выполнению конструкторских технологических документов на печатающих и графических устройствах вывода ЭВМ.

ГОСТ 2.051-2006. Единая система конструкторской документации. Электронные документы. Общие положения.

ГОСТ 2.052-2006. Единая система конструкторской документации. Электронная модель изделия. Общие положения.

ГОСТ 2.601-2013. Единая система конструкторской документации. Эксплуатационные документы.

2 Назначение и принцип действия

2.1 Назначение изделия

МСУ ФОЭ – представляет из себя прототип устройства слежения за Солнцем на основе однокристалльного микроконтроллера. Слежение за Солнцем представляет собой расчёт его текущий координат на небесной сфере (азимут и наклонение) и вывод их на алфавитно-цифровой индикатор.

В состав изделия входят: Блок управления, содержащий: устройство отсчёта интервалов времени на основе однокристалльного микроконтроллера; светодиодный цифровой индикатор для отображения текущего времени; блок питания с выходным напряжением +5В.

2.2 Области использования изделия

Изделие может использоваться для слежения фотоэлектрической установкой за Солнцем в целях повышения количества производимой электроэнергии.

2.3 Принцип действия

Пользователь включает установку. Пользователь вводит текущую дату, что приводит к обновлению координат Солнца. В случае подключения установки к компьютеру в консоль будет выведено «setup», что индицирует обновление внутренних переменных, а затем «Update», что индицирует обновление координат Солнца. Если Солнце в указанный момент времени над горизонтом, то установка в начале каждой минуты будет писать в консоль «<ADJUSTING>», что указывает на попытку поворота установки к Солнцу, однако физически никакого действия произведено не будет, в связи с отсутствием у прототипа силовой части.

3 Состав изделия и комплектность

В комплект поставки входит:

- Плата устройства
- Сетевой питающий адаптер
- Паспорт.

4 Технические характеристики

4.1 Основные технические характеристики блока

Основные технические характеристики приведены в таблице 1.

Таблица 1 – Технические характеристики блока

Наименование параметра	Значение
Потребляемый ток, мА	65
Диапазон рабочих температур прибора, С°	-55 ... +125
Питание, В	5
Точность наведения, °	±1
Масса нетто, грамм	140

5 Устройство и описание работы изделия

5.1 Устройство изделия

Структурная схема изделия представлена на рисунке 1.

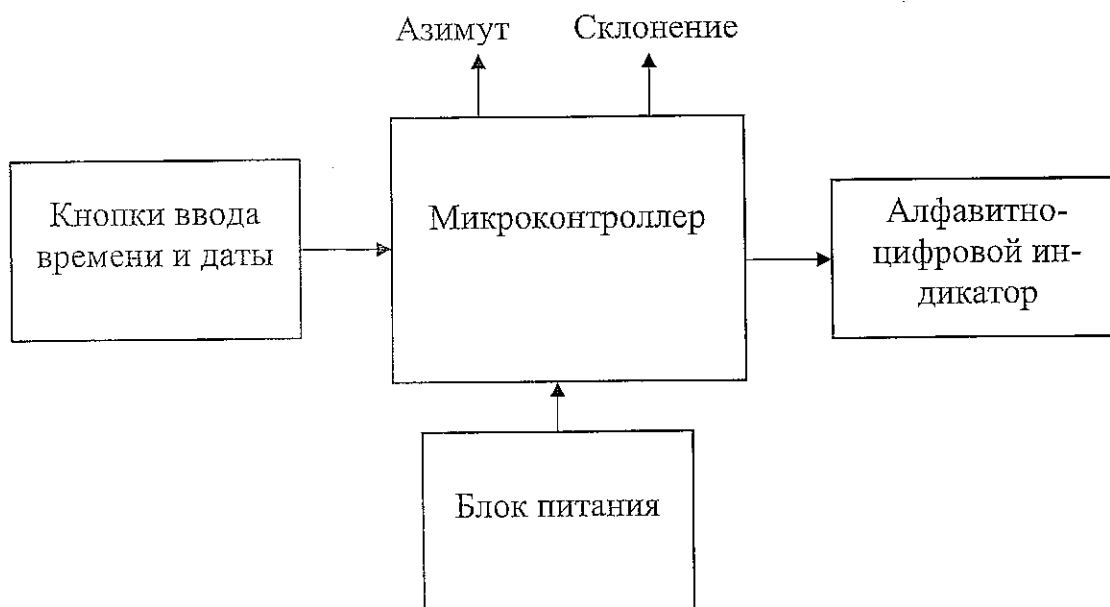


Рисунок 1 – Структурная схема изделия

В состав изделия входят:

- Кнопки коррекции времени;
- Блок управления;
- Алфавитно-цифровой индикатор;
- Блок питания;

Кнопки коррекции времени позволяют установить текущее время.

Блок управления представляет из себя модуль ARDUINO-NANO, на основе микроконтроллера ATM328P:

- Выполняет опрос текущего времени из микросхемы часов DS1307 по интерфейсу I2C;
- В полночь или при изменении времени производится расчёт констант на предстоящий день;

– Затем каждую минуту происходит проверка нахождения Солнца над горизонтом путём вычисления координат Солнца на небесной сфере;

– Если Солнце над горизонтом, производится подача команды на коррекцию положения.

Алфавитно-цифровой индикатор – служит для контроля коррекции времени и ввода расписания коммутации нагрузки.

Блок питания преобразует однофазное переменное напряжение 220V в постоянное напряжение +5 V для питания схемы устройства.

5.2 Описание работы изделия

Пользователь включает установку. Вводит текущих даты и времени. Система выводит текущее положение Солнца, если оно находится над горизонтом.

Блок-схема работы управляющей программы приведена в Приложении А.

Настройка даты и времени:

У установки имеются три кнопки: «set», «+» и «-». Нажатие «set» циклически переключает между настройкой дня недели, года, месяца, дня, часа, минут, секунд и основным экраном. Нажатие кнопки «+» увеличивает выбранное значение, а «-» – уменьшается. Нажатие любой из этих двух кнопок на начальном экране не имеет функции. При попытке увеличить любое значение выше своего предела, значение будет приравнено до минимально допустимого. Например, при попытке увеличения секунд до 60 секунды примут значение 0.

6 Условия эксплуатации

Изделие выпускается в климатическом исполнении УХЛ 4.2 по ГОСТ 15150-69 и предназначен для использования в стационарных условиях в закрытых помещениях при соответствующих климатических условиях:

- интервал температур от +10 до +35 °С;
- относительная влажность воздуха до 80 % при температуре +25 °С;
- высота над уровнем моря не более 2000 м;
- атмосферное давление от 86,6 до 106 кПа (от 650 до 800 мм рт. ст.).

В помещении, где используется изделие не должно возникать условий для конденсации влаги (выпадения росы). Изделие является электронным прибором, требующим бережного обращения.

Для обеспечения безотказной работы, сохранения точности и его сбережения необходимо соблюдать следующие правила:

- изучить паспорт, прежде чем приступить к работе с изделием;
- предохранять изделие от ударов и повреждений;

6.1 Правила и особенности размещения изделия

ВНИМАНИЕ! При эксплуатации изделия запрещается проводить самостоятельно какие-то либо работы по извлечению и установке внутренних компонентов изделия.

6.2 Меры безопасности

Необходимо соблюдать требования техники безопасности и следующие меры предосторожности:

- Внутренние осмотры и ремонт изделия должны производиться только квалифицированными специалистами;
- Не устанавливайте изделие на неустойчивой подставке, стойке или ненадежном кронштейне.
- В момент запуска установка должна быть надёжно закреплена

6.3 Правила хранения и транспортирования

Транспортирование изделия в упакованном виде может производиться железнодорожным, автомобильным (в закрытых транспортных средствах), воздушным, речным и морским видами транспорта в соответствии с правилами перевозок грузов, действующих на транспорт данного вида. Условия транспортирования изделия по части воздействия климатических факторов должны соответствовать группе 5 по ГОСТ 15150.

После транспортирования изделие должно быть выдержано не менее 2 часов в транспортной таре при температуре 20 ± 5 °С и относительной влажности воздуха не более 80 %.

Распакованное изделие должно храниться в отапливаемом и вентилируемом чистом помещении при температуре от +5 до +40 °С и относительной влажности воздуха не более 60 %. При температуре ниже 25 °С допускается увеличение относительной влажности до 80 %. Воздух в помещении не должен содержать примесей, вызывающих коррозию металлов, налеты на поверхностях оптических деталей.

ПРИЛОЖЕНИЕ А

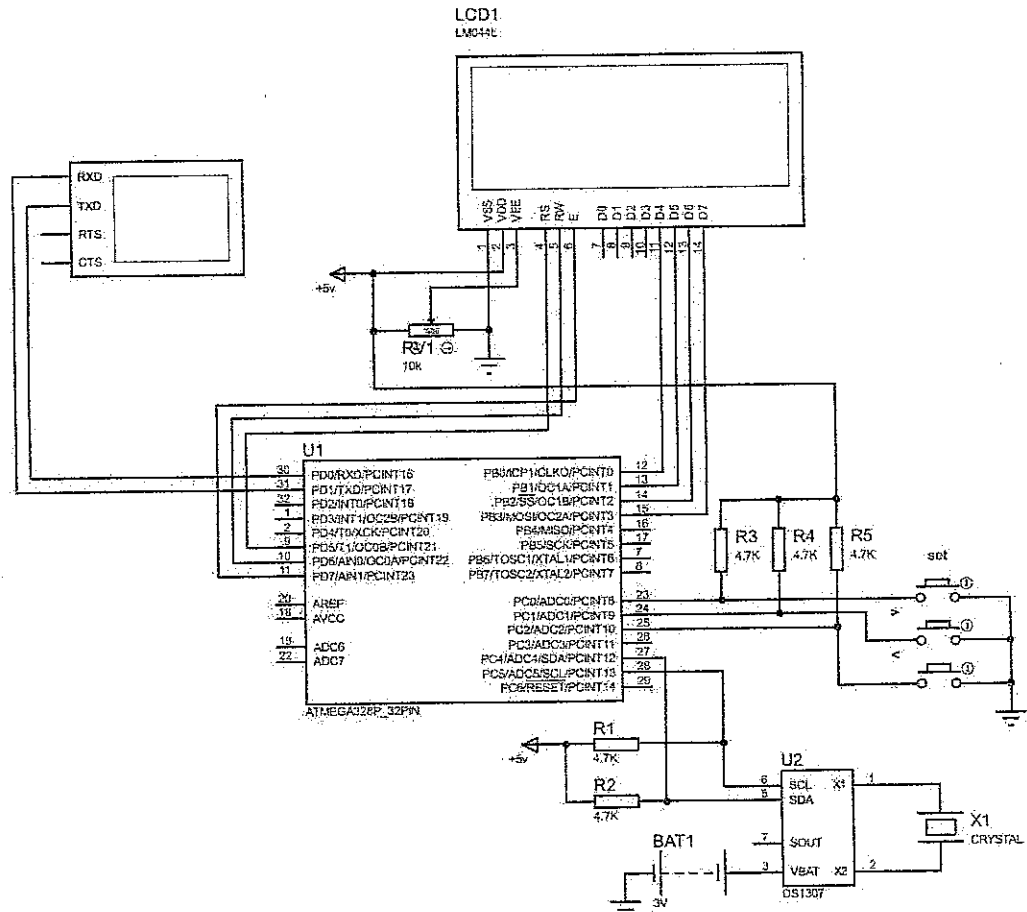


Рисунок А1 – Принципиальная схема изделия

Таблица А1 – Спецификация к принципиальной схеме

Спецификация к принципиальной схеме			
Поз. обозначение	Наименование	Кол.	Примечание
	Резисторы		
R1-R5	4,7 кОм	5	
RV1	10 кОм	1	
	Интегральные схемы		
U1	ATMEGA328P 32PIN	1	
U2	DS1307	1	
	Разное		
X1	Crystal	1	
LCD1	LM044L	1	
BAT1	3 В	1	

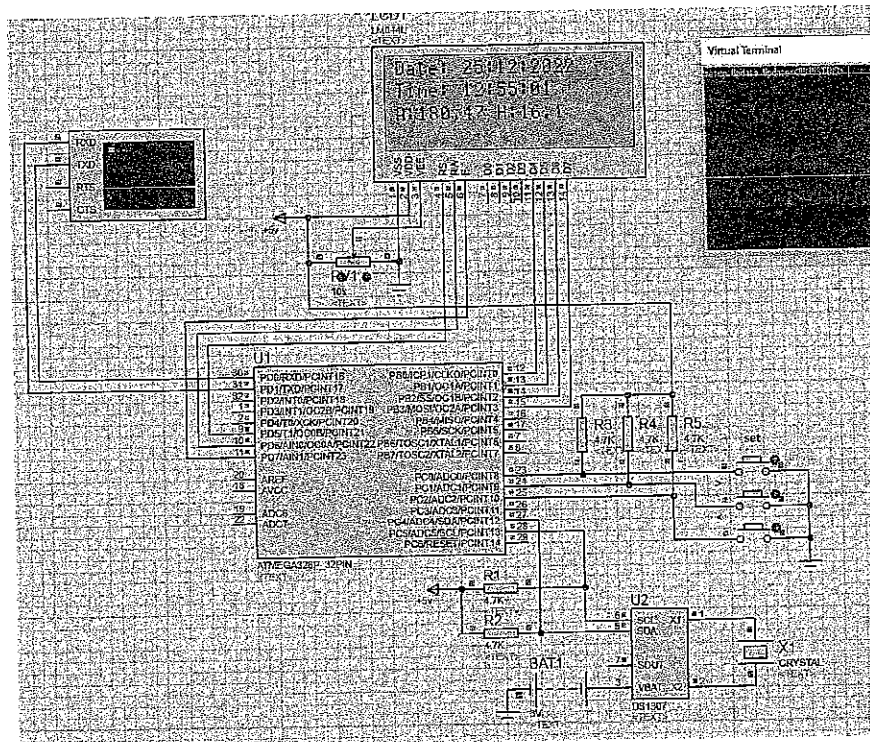


Рисунок А2 – Модель изделия в Isis Proteus
(для даты: 25.12.2022 и времени: 12:55:01, координаты солнца: азимут – 180,47 град., а склонение – 16,1 град.)

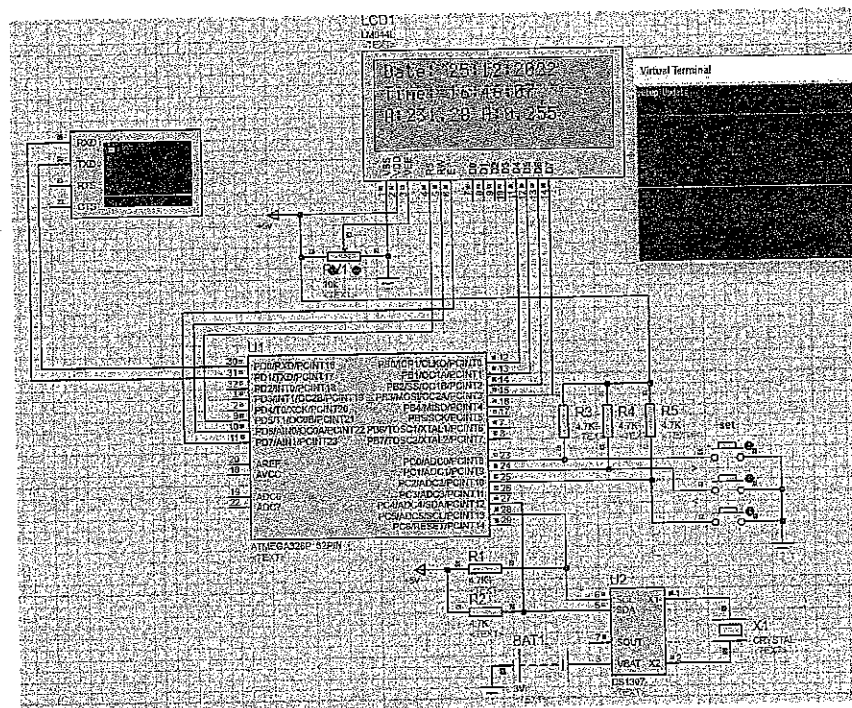


Рисунок А3 – Модель изделия в Isis Proteus
(для даты: 25.12.2022 и времени: 16:46:07, координаты солнца: азимут – 231,28 град., а склонение – 0,255 град.)

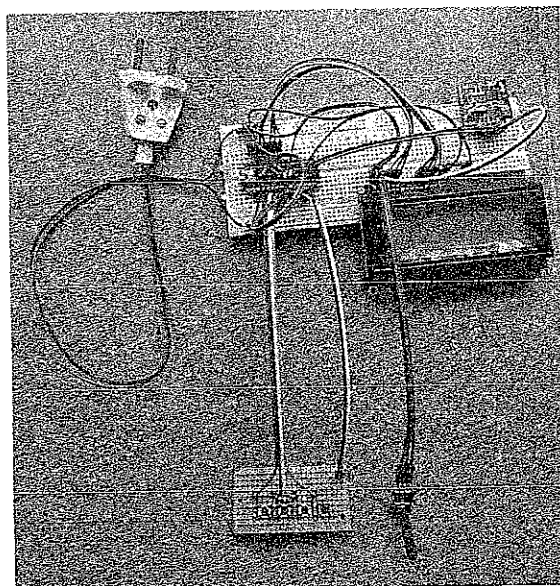


Рисунок А4 – Внешний вид макета изделия

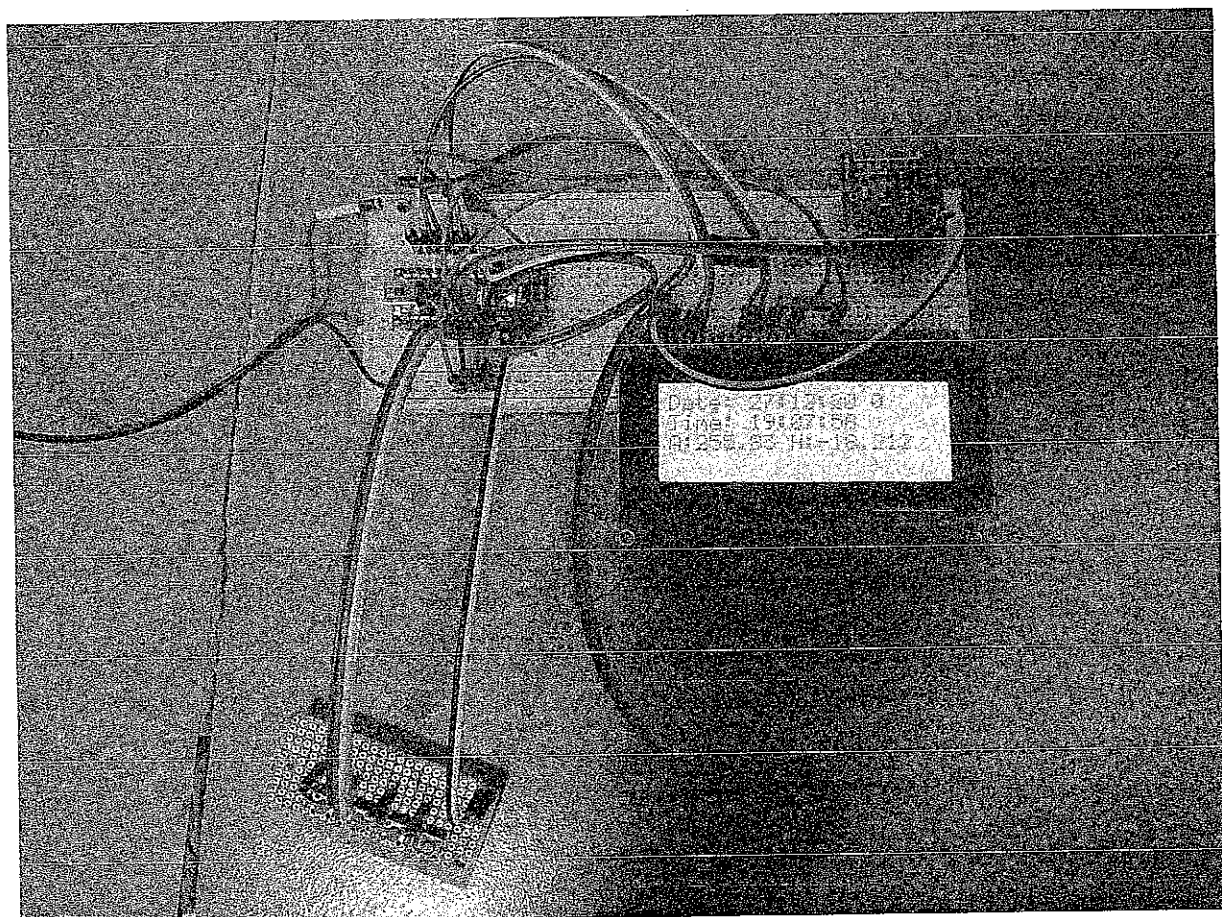



Рисунок А5 – Тестирование системы управления фотоэлектрической энергоустановкой (для даты: 27.12.2022 и времени: 19:07:58, координаты солнца: азимут – 255,83 град., а склонение – -18,212 град., солнце зашло за горизонт)

 Положение Солнца

Дата
Вт 27 декабря 2022 19 : 07

Разница с Гринвичем
10

Широта
50° 34' 1" с.ш. ю.ш.

Долгота
136° 57' 57" в.д. з.д.

Точность вычисления
Знаков после запятой: 2

Азимут (градусы)
257.21

Высота над горизонтом (градусы)
-19.98

Рисунок А6 – Результат расчета солнечного онлайн-калькулятора, расположенного по адресу <https://planetcalc.ru/320/>

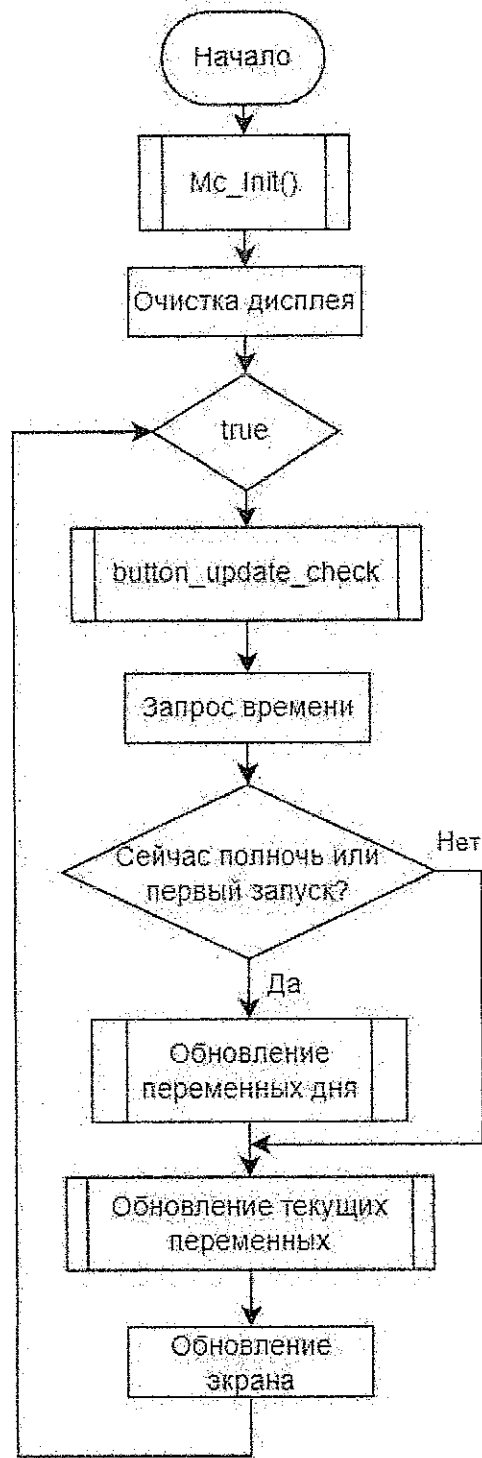


Рисунок А7 – Блок схема функции void main()

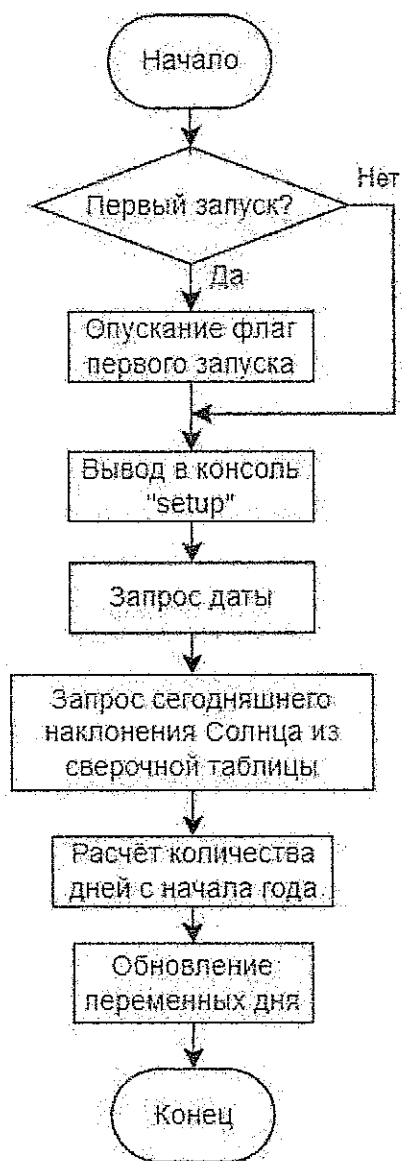


Рисунок А8 – Блок-схема процесса обновления переменных дня

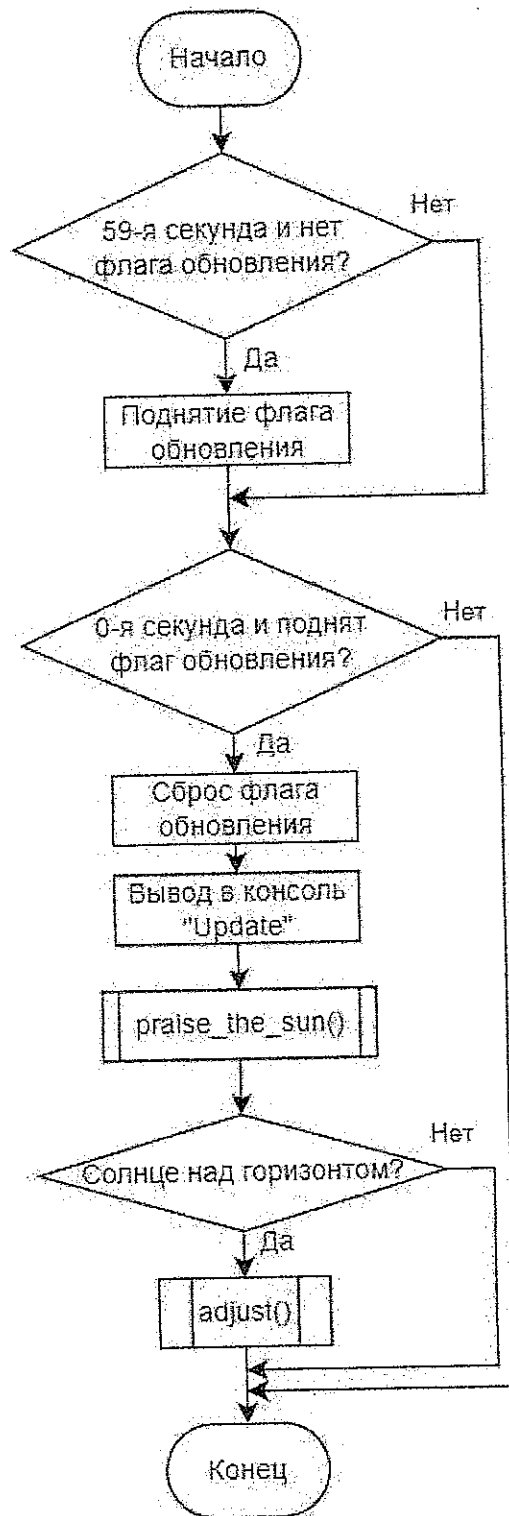


Рисунок А9 – Блок-схема процесса обновления текущих переменных

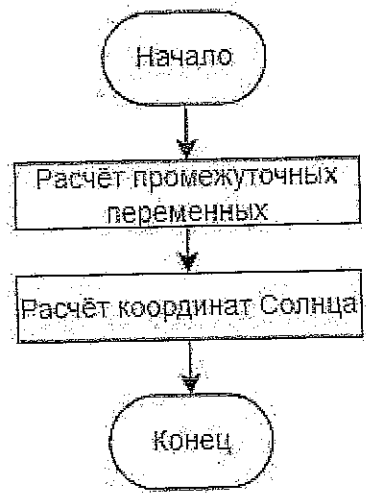


Рисунок А 10 – Блок-схема функции void praise_the_sun()

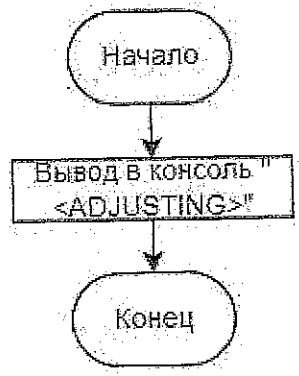


Рисунок А11 – Блок-схема функции void adjust()

Листинг управляющей программы

```
/*
Chip type      : ATmega328P
Program type   : Application
AVR Core Clock frequency: 16,000000 MHz
Memory model   : Small
External RAM size : 0
Data Stack size : 512
*/
#include <mega328p.h>
#include <i2c.h> // I2C Bus functions
#include <ds1307.h> // DS1307 Real Time Clock functions
#include <alcd.h> // Alphanumeric LCD functions
#include <stdio.h> // Standard Input/Output functions
#include <delay.h>
// #include <stdafx.h>
// #include <conio.h>
#include <math.h>
#define RADEG (180.0/PI)
#define DEGRAD (PI/180.0)
#define sindeg(x) sin((x)*DEGRAD)
#define cosdeg(x) cos((x)*DEGRAD)
#define atan2deg(y,x) (RADEG*atan2((y),(x)))
#define asindeg(x) (RADEG*asin(x))
#define Tm hour+minute/60.0

// Временно не настраиваемые переменные
float UTC, Np = 10;
float LON = 136.96589, lat = 50.56703;
// ==

float Tsun, Tclock, EOT;
float Decl, t; // склонение и часовой угол Солнца (auto, input)
float A, h; // азимут и угол-места Солнца над горизонтом (output)
float B; // параметр уравнения времени (auto)
int N; // номер дня в году (auto)

char lcd_data0[20];

unsigned char week_day, day, month, year1;
int year;
unsigned char hour, minute, sec;
unsigned char flag = 0b10000000; // флаг [запущен в первый раз?][срабатывал ли в текущей минуте?][--][--][--][--]
unsigned char set = 0;

void adjust(void);
void praise_the_sun();
void button_update_check(void);
void update_screen(void);
void MC_Init(void);

static __flash float lookup_table[12][31] =
{
    {-23.0667, -22.9833, -22.9, -22.8, -22.7, -22.6, -22.4667, -22.35, -22.2167, -22.0833, -21.9333, -21.7833, -21.6167, -
    21.45, -21.2667, -21.1, -20.9, -20.7, -20.5, -20.3, -20.0833, -19.8667, -19.6333, -19.4, -19.1667, -18.9167, -18.6667, -18.4167, -
    18.15, -17.8833, -17.6167},
    {-17.3333, -17.05, -16.7667, -16.4667, -16.1667, -15.8667, -15.5667, -15.25, -14.9333, -14.6167, -14.3, -13.9667, -
    13.6333, -13.3, -12.9667, -12.6167, -12.2667, -11.9167, -11.5667, -11.2167, -10.8667, -10.5, -10.1333, -9.76667, -9.4, -9.03333,
    -8.65, -8.28333, -8.05},
    {-7.81667, -7.43333, -7.05, -6.66667, -6.28333, -5.9, -5.5, -5.11667, -4.73333, -4.33333, -3.95, -3.55, -3.16667, -
    2.76667, -2.36667, -1.98333, -1.58333, -1.18333, -0.8, -0.4, 0, 0.4, 0.783333, 1.183333, 1.583333, 1.966667, 2.366667, 2.75,
    3.15, 3.533333, 3.916667},
    {4.3, 4.7, 5.083333, 5.466667, 5.85, 6.216667, 6.6, 6.983333, 7.35, 7.716667, 8.116667, 8.466667, 8.833333,
    9.183333, 9.55, 9.9, 10.26667, 10.61667, 10.96667, 11.31667, 11.65, 12, 12.33333, 12.66667, 13, 13.31667, 13.63333,
    13.96667, 14.26667, 14.58333},
    {14.9, 15.2, 15.5, 15.78333, 15.08333, 16.36667, 16.65, 16.91667, 17.2, 17.45, 17.71667, 17.98333, 18.23333,
    18.48333, 18.71667, 18.96667, 19.18333, 19.41667, 19.63333, 19.85, 20.06667, 20.26667, 20.46667, 20.65, 20.83333,
    21.01667, 21.2, 21.36667, 21.51667, 21.68333, 21.83333},
};
```



```

    {21.96667, 22.1, 22.23333, 22.36667, 22.48333, 22.58333, 22.7, 22.78333, 22.88333, 22.96667, 23.03333, 23.11667,
    23.18333, 23.23333, 23.28333, 23.33333, 23.36667, 23.4, 23.41667, 23.43333, 23.43333, 23.43333, 23.43333, 23.41667, 23.4,
    23.38333, 23.35, 23.31667, 23.26667, 23.21667},
    {23.15, 23.08333, 23.01667, 22.93333, 22.85, 22.75, 22.65, 22.55, 22.43333, 22.31667, 22.18333, 22.06667, 21.91667,
    21.76667, 21.61667, 21.46667, 21.3, 21.13333, 20.96667, 20.78333, 20.6, 20.4, 20.2, 20, 19.78333, 19.56667, 19.35, 19.13333,
    18.9, 18.66667, 18.41667},
    {18.16667, 17.9167, 17.66667, 17.4, 17.13333, 16.86667, 16.6, 16.31667, 16.03333, 15.75, 15.45, 15.16667, 14.86667,
    14.55, 14.25, 13.93333, 13.61667, 13.3, 12.98333, 12.65, 12.31667, 11.98333, 11.65, 11.31667, 10.96667, 10.63333, 10.28333,
    9.93333, 9.58333, 9.216667, 8.866667},
    {8.5, 8.15, 7.783333, 7.416667, 7.05, 6.666667, 6.3, 5.933333, 5.55, 5.166667, 4.8, 4.416667, 4.033333, 3.65,
    3.266667, 2.883333, 2.5, 2.1, 1.716667, 1.333333, 0.95, 0.55, 0.166667, -0.23333, -0.61667, -1, -1.4, -1.78333, -2.16667, -
    2.56667},
    {-2.95, -3.33333, -3.73333, -4.11667, -4.5, -4.88333, -5.26667, -5.65, -6.03333, -6.41667, -6.8, -7.16667, -7.53333, -
    7.91667, -8.3, -8.66667, -9.03333, -9.4, -9.75, -10.1167, -10.4833, -10.8333, -11.2, -11.55, -11.9, -12.2333, -12.5833, -12.9167, -
    13.25, -13.5833, -13.9167},
    {-14.2333, -14.5667, -14.8833, -15.1833, -15.5, -15.8, -16.1, -16.4, -16.6833, -16.9667, -17.25, -17.5333, -17.8, -
    18.0667, -18.3333, -18.5833, -18.8333, -19.0833, -19.3167, -19.55, -19.7833, -20, -20.2167, -20.4333, -20.6333, -20.8333, -
    21.0167, -21.2, -21.3833, -21.55},
    {-21.7167, -21.8667, -22.0167, -22.1667, -22.3, -22.4167, -22.5333, -22.65, -22.7667, -22.8667, -22.95, -23.0333, -
    23.1167, -23.1833, -23.2333, -23.2833, -23.3333, -23.3667, -23.4, -23.4167, -23.4333, -23.4333, -23.4333, -23.4333, -23.4167, -
    23.3833, -23.35, -23.3167, -23.2667, -23.2, -23.1333}
};
static __flash float DAYS[12] =
{
    0, 28, 59, 89, 120, 150,
    181, 212, 242, 273, 303, 334
};

void main(void)
{
    MC_Init();
    lcd_clear();

    while (1)
    {
        button_update_check();
        rtc_get_time(&hour, &minute, &sec);
//hour = 12; minute = 0; sec = 0;
//printf("iter ");
        if (((hour == 0) && (minute == 0) && (sec == 0)) || (flag & 128))
        {
            if (flag & 128)
                flag -= 128;
            printf("setup ");
                rtc_get_date(&week_day, &day, &month, &year1); //считать дату
//day = 11; month = 11; year1 = 22;
year = 2000 + year1;
Decl = lookup_table[month-1][day-1]; // смотрим сегодняшнее наклонение Солнца
N = DAYS[month-1]; // смотрим кол-во дней, прошедших с начала года до начала этого месяца
N += day; // добавляем кол-во дней, прошедших с начала этого месяца
if (month > 2) // учитываем високосность
    if ((year % 400) || (!(year % 100) && (year % 4)))
        N++;
B = (360uL * (N - 81)) / 365;
EOT = (9.87 * sindeg(2 * B) - 7.53 * cosdeg(B) - 1.5 * sindeg(B)) / 60; // (в часах)
goto first;
        }
        if ((sec == 59) && !(flag & 64))
            flag += 64;
        if ((sec == 0) && (flag & 64))
        {
            //flag -= 64;
            first:
            printf("Update ");
            praise_the_sun(); // обновляем текущие углы поворота
            if (h >= 0)
                adjust();
        }

        update_screen();
    }
}

```

```

    }
}

void adjust(void)
{
    printf("<ADJUSTING> ");
}

void praise_the_sun()
{
    UTC = Tm - Np;
    Tclock = UTC + LON / 15; // (в часах)
    Tsun = Tclock + BOT;
    t = (Tsun + 12) * 15; // (в градусах)
    // расчет азимута от направления на север, по часовой стрелке
    h = asindeg(sindeg(Decl) * sindeg(lat) + cosdeg(Decl) * cosdeg(lat) * cosdeg(t));
    A = atan2deg(cosdeg(Decl) * sindeg(t),
        cosdeg(Decl) * sindeg(lat) * cosdeg(t) - sindeg(Decl) * cosdeg(lat)) + 180;
}

void button_update_check(void)
{
    set = 0;
    if (PINC.0 == 0)
    {
        delay_ms(200);
        set = 1;

        while (set == 1)
        {
            delay_ms(200);
            switch (PINC & 0b00000111) // Отслеживаем нажатие кнопок
            {
                case 0x05: week_day++; if (week_day > 7) { week_day = 1; }; break; // Если нажата кнопка ">"
                // то увеличиваем количество дней недели на 1
                case 0x03: week_day--; if (week_day == 0) { week_day = 7; }; break; // Если нажата кнопка "<"
                // то уменьшаем количество дней недели на 1
                case 0x06: delay_ms(200); set = 2; break; // Если нажата кнопка "Set", переходим к настройке
                // дня
            };
            lcd_clear();
            // sprintf(lcd_data0, "Hour: %u%u", hour/10, hour%10); // Запись в буфер значения дня недели
            sprintf(lcd_data0, "Week_day: %02i", week_day); // Запись в буфер значения дня недели
            lcd_puts(lcd_data0); // Вывод на экран текущее значения дня недели
            // if (!(flag & 128)) // Обновление по редактированию времени
            //     flag += 128;
        };
        while (set == 2)
        {
            delay_ms(200);
            switch (PINC & 0b00000111) // Отслеживаем нажатие кнопок
            {
                case 0x05: day++; if (day > 31) { day = 1; }; break; // Если нажата кнопка ">" то увеличиваем
                // количества дней на 1
                case 0x03: day--; if (day == 0) { day = 31; }; break; // Если нажата кнопка "<" то уменьшаем
                // количества дней на 1
                case 0x06: delay_ms(200); set = 3; break; // Если нажата кнопка "Set", переходим к настройке
                // месяца
            };
            lcd_clear(); // Очистка экрана
            // sprintf(lcd_data0, "Min: %u%u", min1/10, min1%10); // Запись в буфер значения дня
            sprintf(lcd_data0, "Day: %02i", day); // Запись в буфер значения дня
            lcd_puts(lcd_data0); // Вывод на экран текущее количество дней
        };
        while (set == 3)
        {
            delay_ms(200);
            switch (PINC & 0b00000111) // Отслеживаем нажатие кнопок
            {

```

```

        case 0x05: month++; if (month > 12) { month = 1; }; break; //Если нажата кнопка ">" то
увеличиваем количество месяцев на 1
        case 0x03: month--; if (month == 0) { month = 12; }; break; //Если нажата кнопка "<" то
уменьшаем количество месяцев на 1
        case 0x06: delay_ms(200); set = 4; break; //Если нажата кнопка "Set", переходим к настройке
года
    };

    lcd_clear(); // Очистка экрана
    //sprintf(lcd_data0,"Sec: %u%u",sec/10,sec%10); //Запись в буфер значения секунд
    sprintf(lcd_data0, "Month: %02i", month); //Запись в буфер значения секунд
    lcd_puts(lcd_data0); // Вывод на экран текущего значения месяца
};
while (set == 4)
{
    delay_ms(200);
    switch (PINC & 0b00000111) // Отслеживаем нажатие кнопок
    {
        case 0x05: year++; if (year > 99) { year = 0; }; break; //Если нажата кнопка ">" то увеличиваем
количество минут на 1
        case 0x03: year--; if (year == 255) { year = 99; }; break; //Если нажата кнопка "<" то уменьшаем
количество минут на 1
        case 0x06: set = 5; rtc_set_date(week_day, day, month, year); break; //Если нажата кнопка "Set",
отправляем в микросхему часов новые значения даты и переходим к настройке времени
    };

    lcd_clear(); // Очистка экрана
    //sprintf(lcd_data0,"Sec: %u%u",sec/10,sec%10); //Запись в буфер значения секунд
    sprintf(lcd_data0, "Year: %02i", year); //Запись в буфер значения секунд
    lcd_puts(lcd_data0); // Вывод на экран текущее количество минут
};

while (set == 5)
{
    delay_ms(200);
    switch (PINC & 0b00000111) // Отслеживаем нажатие кнопок
    {
        case 0x05: hour++; if (hour > 23) { hour = 0; }; break; //Если нажата кнопка ">" то увеличиваем
количество часов на 1
        case 0x03: hour--; if (hour == 255) { hour = 23; }; break; //Если нажата кнопка "<" то
уменьшаем количество часов на 1
        case 0x06: delay_ms(200); set = 6; break; //Если нажата кнопка "Set", переходим к настройкам
минут
    };

    lcd_clear(); // Очистка экрана
    //sprintf(lcd_data0,"Hour: %u%u",hour/10,hour%10); //Запись в буфер значения часов
    sprintf(lcd_data0, "Hour: %02i", hour); //Запись в буфер значения часов
    lcd_puts(lcd_data0); // Вывод на экран текущее значения часов
};
while (set == 6)
{
    delay_ms(200);
    switch (PINC & 0b00000111) // Отслеживаем нажатие кнопок
    {
        case 0x05: minute++; if (minute > 59) { minute = 0; }; break; //Если нажата кнопка ">" то
увеличиваем количество минут на 1
        case 0x03: minute--; if (minute == 255) { minute = 59; }; break; //Если нажата кнопка "<" то
уменьшаем количество минут на 1
        case 0x06: delay_ms(200); set = 7; break; //Если нажата кнопка "Set", переходим к настройкам
минут
    };

    lcd_clear(); // Очистка экрана
    //sprintf(lcd_data0,"Min: %u%u",min1/10,min1%10); //Запись в буфер значения минут
    sprintf(lcd_data0, "Min: %02i", minute); //Запись в буфер значения минут
    lcd_puts(lcd_data0); // Вывод на экран текущее количество минут
};
while (set == 7)
{
    delay_ms(200);

```

```

switch (PINC & 0b00000111) // Отслеживаем нажатие кнопок
{
case 0x05: sec++; if (sec > 59) { sec = 0; }; break; //Если нажата кнопка ">" то увеличиваем
количество минут на 1
case 0x03: sec--; if (sec == 255) { sec = 59; }; break; //Если нажата кнопка "<" то уменьшаем
количество минут на 1
case 0x06: set = 0; rtc_set_time(hour, minute, sec); break; //Если нажата кнопка "Set",
отправляем в микросхему часов новые значения времени
};

lcd_clear(); // Очистка экрана
//sprintf(lcd_data0, "Sec: %u%u", sec/10, sec%10); //Запись в буфер значения секунд
sprintf(lcd_data0, "Sec: %02i", sec); //Запись в буфер значения секунд
lcd_puts(lcd_data0); // Вывод на экран текущее количество минут
};
};

void update_screen(void)
{
// В pres кол-во нулей отвечает за кол-во знаков после запятой (не больше 2-х)
#define pres 100
lcd_clear(); //очистка экрана

sprintf(lcd_data0, "Date: %02i:%02i:20%2i ", day, month, year1); // Запись в буфер значения даты
lcd_gotoxy(0, 0);
lcd_puts(lcd_data0); //Выводим на экран текущую дату

sprintf(lcd_data0, "Time: %02i:%02i:%02i ", hour, minute, sec); // Запись в буфер значения времени
lcd_gotoxy(0, 1);
lcd_puts(lcd_data0); //Выводим на экран текущее время

sprintf(lcd_data0, "A:%i.%i H:%i.%i", (int)A, (int)(A*pres)-(int)A*pres, (int)h, (unsigned char)((h*pres)-(int)h*pres));
lcd_gotoxy(0, 2);
lcd_puts(lcd_data0);
delay_ms(200);
};

void MC_Init(void)
{
// Crystal Oscillator division factor: 1
#pragma optsize-
CLKPR = (1 << CLKPCE);
CLKPR = (0 << CLKPCE) | (0 << CLKPS3) | (0 << CLKPS2) | (0 << CLKPS1) | (0 << CLKPS0);
#ifdef _OPTIMIZE_SIZE_
#pragma optsize+
#endif
// Input/Output Ports initialization
// Port B initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
DDRB = (0 << DDB7) | (0 << DDB6) | (0 << DDB5) | (0 << DDB4) | (0 << DDB3) | (0 << DDB2) | (0 << DDB1) | (0
<< DDB0);
// State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
PORTB = (0 << PORTB7) | (0 << PORTB6) | (0 << PORTB5) | (0 << PORTB4) | (0 << PORTB3) | (0 << PORTB2) |
(0 << PORTB1) | (0 << PORTB0);
// Port C initialization
// Function: Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
DDRC = (0 << DDC6) | (0 << DDC5) | (0 << DDC4) | (0 << DDC3) | (0 << DDC2) | (0 << DDC1) | (0 << DDC0);
// State: Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
PORTC = (0 << PORTC6) | (0 << PORTC5) | (0 << PORTC4) | (0 << PORTC3) | (0 << PORTC2) | (0 << PORTC1) |
(0 << PORTC0);
// Port D initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
DDRD = (0 << DDD7) | (0 << DDD6) | (0 << DDD5) | (0 << DDD4) | (0 << DDD3) | (0 << DDD2) | (0 << DDD1) | (0
<< DDD0);
// State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
PORTD = (0 << PORTD7) | (0 << PORTD6) | (0 << PORTD5) | (0 << PORTD4) | (0 << PORTD3) | (0 << PORTD2) |
(0 << PORTD1) | (0 << PORTD0);
};

```

```

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=0xFF
// OCA output: Disconnected
// OCB output: Disconnected
TCCR0A = (0 << COM0A1) | (0 << COM0A0) | (0 << COM0B1) | (0 << COM0B0) | (0 << WGM01) | (0 <<
WGM00);
TCCR0B = (0 << WGM02) | (0 << CS02) | (0 << CS01) | (0 << CS00);
TCNT0 = 0x00;
OCR0A = 0x00;
OCR0B = 0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OCA output: Disconnected
// OCB output: Disconnected
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A = (0 << COM1A1) | (0 << COM1A0) | (0 << COM1B1) | (0 << COM1B0) | (0 << WGM11) | (0 <<
WGM10);
TCCR1B = (0 << ICNC1) | (0 << ICES1) | (0 << WGM13) | (0 << WGM12) | (0 << CS12) | (0 << CS11) | (0 <<
CS10);
TCNT1H = 0x00;
TCNT1L = 0x00;
ICR1H = 0x00;
ICR1L = 0x00;
OCR1AH = 0x00;
OCR1AL = 0x00;
OCR1BH = 0x00;
OCR1BL = 0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=0xFF
// OCA output: Disconnected
// OCB output: Disconnected
ASSR = (0 << EXCLK) | (0 << AS2);
TCCR2A = (0 << COM2A1) | (0 << COM2A0) | (0 << COM2B1) | (0 << COM2B0) | (0 << WGM21) | (0 <<
WGM20);
TCCR2B = (0 << WGM22) | (0 << CS22) | (0 << CS21) | (0 << CS20);
TCNT2 = 0x00;
OCR2A = 0x00;
OCR2B = 0x00;

// Timer/Counter 0 Interrupt(s) initialization
TIMSK0 = (0 << OCIE0B) | (0 << OCIE0A) | (0 << TOIE0);

// Timer/Counter 1 Interrupt(s) initialization
TIMSK1 = (0 << ICIE1) | (0 << OCIE1B) | (0 << OCIE1A) | (0 << TOIE1);

// Timer/Counter 2 Interrupt(s) initialization
TIMSK2 = (0 << OCIE2B) | (0 << OCIE2A) | (0 << TOIE2);

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// Interrupt on any change on pins PCINT0-7: Off
// Interrupt on any change on pins PCINT8-14: Off
// Interrupt on any change on pins PCINT16-23: Off
EICRA = (0 << ISC11) | (0 << ISC10) | (0 << ISC01) | (0 << ISC00);
EIMSK = (0 << INT1) | (0 << INT0);

```

```

PCICR = (0 << PCIE2) | (0 << PCIE1) | (0 << PCIE0);

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART0 Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA = (0 << RXC0) | (0 << TXC0) | (0 << UDRE0) | (0 << FE0) | (0 << DOR0) | (0 << UPE0) | (0 << U2X0) | (0
<< MPCM0);
UCSR0B = (0 << RXCIE0) | (0 << TXCIE0) | (0 << UDRIE0) | (1 << RXEN0) | (1 << TXEN0) | (0 << UCSZ02) | (0
<< RXB80) | (0 << TXB80);
UCSR0C = (0 << UMSEL01) | (0 << UMSEL00) | (0 << UPM01) | (0 << UPM00) | (0 << USBS0) | (1 << UCSZ01) |
(1 << UCSZ00) | (0 << UCPOL0);
UBRR0H = 0x00;
UBRR0L = 0x67;

// Analog Comparator initialization
// Analog Comparator: Off
// The Analog Comparator's positive input is
// connected to the AIN0 pin
// The Analog Comparator's negative input is
// connected to the AIN1 pin
ACSR = (1 << ACD) | (0 << ACBG) | (0 << ACO) | (0 << ACI) | (0 << ACIE) | (0 << ACIC) | (0 << ACIS1) | (0 <<
ACIS0);
ADCSRB = (0 << ACME);
// Digital input buffer on AIN0: On
// Digital input buffer on AIN1: On
DIDR1 = (0 << AIN0D) | (0 << AIN1D);

// ADC initialization
// ADC disabled
ADCSRA = (0 << ADEN) | (0 << ADSC) | (0 << ADIF) | (0 << ADIE) | (0 << ADPS2) | (0 <<
ADPS1) | (0 << ADPS0);

// SPI initialization
// SPI disabled
SPCR = (0 << SPIE) | (0 << SPE) | (0 << DORD) | (0 << MSTR) | (0 << CPOL) | (0 << CPHA) | (0 << SPR1) | (0 <<
SPR0);

// TWI initialization
// TWI disabled
TWCR = (0 << TWEA) | (0 << TWSTA) | (0 << TWSTO) | (0 << TWEN) | (0 << TWIE);
// Bit-Banged I2C Bus initialization
// I2C Port: PORTC
// I2C SDA bit: 4
// I2C SCL bit: 5
// Bit Rate: 100 kHz
// Note: I2C settings are specified in the
// Project|Configure|C Compiler|Libraries|I2C menu.
i2c_init();
// DS1307 Real Time Clock initialization
// Square wave output on pin SQW/OUT: Off
// SQW/OUT pin state: 0
rtc_init(0, 0, 0);
// Alphanumeric LCD initialization
// Connections are specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTD Bit 5
// RD - PORTD Bit 6
// EN - PORTD Bit 7
// D4 - PORTB Bit 0
// D5 - PORTB Bit 1
// D6 - PORTB Bit 2
// D7 - PORTB Bit 3
// Characters/line: 20
lcd_init(20);
};

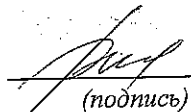
```

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

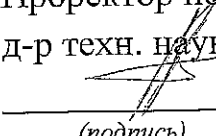
СОГЛАСОВАНО

Начальник отдела ОНиПКРС

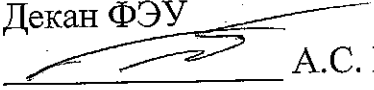

(подпись) Е.М. Димитриади
« 19 » 06 20 23 г.

УТВЕРЖДАЮ

Проректор по научной работе,
д-р техн. наук, профессор


(подпись) А.В. Космынин
« 19 » 06 20 23 г.

Декан ФЭУ


(подпись) А.С. Гудим
« 19 » 06 20 23 г.

АКТ

о приемке в эксплуатацию проекта

Аппаратно-программный комплекс

«Микроконтроллерная система управления фотоэлектрической
энергоустановкой»

г. Комсомольск-на-Амуре

« 20 » 12 2022 г.

Комиссия в составе представителей:

со стороны заказчика

- С.И. Сухоруков – руководитель СКБ,
- А.С. Гудим – декан ФЭУ

со стороны исполнителя

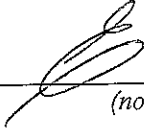
- В.А. Егоров – руководитель проекта,
- Н.Л. Минебаев – ОМРБ-1

составила акт о нижеследующем:

«Исполнитель» передает проект «Микроконтроллерная система управления фотоэлектрической энергоустановкой», в составе:


1. Блок управления фотоэлектрической установкой.
2. Рабочие программы управления изделием.
3. Паспорт изделия

Руководитель проекта


20.12.2022
(подпись, дата)

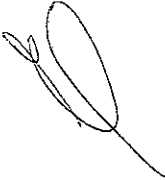
В.А. Егоров

Исполнитель проекта


(подпись, дата)

Н.Л. Минебаев

Таблица учета проектной работы в учебных дисциплинах

Дисциплина	Форма учтенной работы (номер ЛР, КП, КР, РГЗ, зачет, итоговая оценка, экзамен)	Преподаватель (дата, ФИО, подпись)	Примечание (ЗУН полученные при выполнении проекта)
Микропроцессорные устройства систем управления	КП	20.12.22 В.А Егоров 	31(ПК-6-2) Правила составления структуры и алгоритма работы микроконтроллера на различных стадиях проектирования системы электропривода 32(ПК-6-2) Типовые решения по структуре и алгоритмам работы микропроцессорной системы электропривода У1(ПК-6-2) Составлять алгоритмы работы микроконтроллера на различных стадиях проектирования системы электропривода У2(ПК-6-2) Осуществлять сбор и обработку справочной информации по типовым решениям о структуре и алгоритме работы микропроцессорной системы электропривода Н1(ПК-6-2) Анализ технического задания на составление алгоритма работы при проектировании микропроцессорной системы электропривода Н2(ПК-6-2) Выбор оптимальных технических решений по структуре и алгоритму работы микропроцессорной системы электропривода