**Реферат**

Курсовой проект содержит 35 страниц, 10 рисунков, 7 таблиц, 49 литературных источников.

ПЛАТФОРМА, ШАГОВЫЙ МОТОР, СТАКАНЧИК, НАСОС, СХЕМОТЕХНИКА, МИКРОКОНТРОЛЛЕР

Объект работы: Аппарат по розливу воды в стаканчики

Цель работы: Разработка системы управления аппарата по розливу воды в стаканчики

В данной курсовой работе проводится описание поэтапной разработки аппарата по розливу воды в стаканчики. Работа включает в себя разработку структурной схемы системы, выбор элементной базы, с описанием отдельных элементов системы, в том числе микроконтроллера. На основе выбранных элементов происходит построение функциональной схемы системы. Проводится разработка алгоритма работы системы и программного кода. В результате получили систему управления аппарата по розливу воды в стаканчики.

**Содержание**

Введение

1. Анализ задачи

1.1 Принцип работы

1.2 Платформа

1.3 Проверка уровня воды в баке

1.4 Наполнение стаканчика водой

1.5 Условия работы

2. Выбор и обоснование элементной базы

2.1 Электромотор

2.2 Водяной насос

2.3 Электромагнитное реле

2.4 Микроконтроллер

3. Разработка функциональной схемы

3.1 Источник питания

3.2 Микроконтроллер

3.3 Тактовый генератор

3.4 Проверка уровня воды

3.5 Индикатор уровня воды

3.6 Схема управления шаговым двигателем

3.7 Схема управления насосом

4. Алгоритм работы

4.1 Описание блок схемы

4.2 Описание хода разработки программного обеспечения

Заключение

Список использованных источников

Приложения

#

# Введение

Курсовой проект по курсу «Микропроцессорные средства» ставит задачей познакомить с азами проектирования устройств, с применением микроконтроллера и написания для него программного обеспечения.

В этой пояснительной записке приводится описание процесса конструирования аппарата по розливу воды в стаканчики. Данный аппарат содержит микроконтроллер с программой, мотор для закачки воды, установка для подачи стаканчиков.

Проведено описание работы, которая была проведена в ходе конструирования, электронная схема аппарата, в которой показано взаимодействие с остальными исполнительными устройствами, описан ход конструирования схемы.

Написание программного обеспечения является одной из основных частей проекта. Приводится блок-схема алгоритма программы, описан ход написания.

К записке прилагаются плакаты с изображением электронной схемы и с блок-схемой алгоритма микропрограммы.

1. Анализ задачи

Проведем анализ поставленной задачи.[1-7] Данный аппарат состоит из множества частей, помимо системы управления на электронной плате, устройство имеет движущийся поднос с шестью стаканчиками, электромотор, приводящий его в движение, закрытый бак для воды на 2 литра, трубки для налива воды, насос, с помощью которого качается вода; запускать аппарат будем кнопкой «ПУСК». Примерная схема аппарата приведена на рисунке 1.1.

Рисунок 1.1 – Схема аппарата по розливу воды в стаканчики

## Принцип работы

Пользователь устанавливает на поднос пустые стаканчики, включает аппарат, при этом начинает светить красный индикатор. Включает питание насоса. И для запуска аппарата нажимает на кнопку «ПУСК». Если уровень воды недостаточен, начнет мигать зеленый индикатор и необходимо долить воды в бак. Если воды достаточно, платформа начнёт движение и передвинет стаканчик под кран; далее аппарат наполнит стаканчик водой и после некоторой паузы передвинет платформу, чтобы наполнить следующий стаканчик и так далее, пока все они не будут заполнены. После окончания работы пользователь выключает аппарат.

**1.2 Платформа**

Необходимо заметить, что стаканчики стоят на подвижном подносе. Если резко начать вращение подноса, то пустые стаканчики вероятнее всего опрокинутся. Очевидно, что для стаканчика уже наполненного водой коэффициент трения о поднос будет выше, так как его масса будет больше. Поэтому при расчете скорости вращения подноса будем опираться на массу пустого стаканчика. Поднос будет вращаться с некоторой скоростью, заданной эмпирическим путём с помощью электродвигателя, установленного в нижней части подноса.

При плавном ускорении и плавном замедлении стаканчики сохранят устойчивость на подносе, и будут двигаться до целевого положения за меньшее время. Для достижения такого движения воспользуемся шаговым мотором. Будем постепенно раскручивать ротор шагового мотора под управлением программы, чтобы с помощью него повернуть платформу на необходимый угол.

Будем считать, что первый стаканчик уже находится под краном, и при повороте на угол 30° под краном будет стоять второй стаканчик, это позволит избежать ошибки «начального положения».

Когда стаканчики установлены под краном, наливается необходимое количество воды, при этом сначала проверяется, есть ли вода в баке.

**1.3** **Проверка уровня воды в баке**

Чтобы проверить уровень воды в баке воспользуемся оптическим датчиком. Минимальный уровень, соответствует объёму воды необходимого для одного стаканчика (200 мл). Максимальный объём воды ограничен только ёмкостью бака (2 л).

Уровень воды проверяется с помощью светодиода и фотодиода, который установлен на трубке на уровне минимального объёма воды. Трубка присоединена к баку. В этой трубке находиться легкий поплавок, когда уровень воды мал, он перекрывает фотодиод и тем самым информирует о недостаточном объёме воды. Поплавок сделан так, чтобы перекрывать фотодиод даже когда в баке вода отсутствует.

Если уровень воды не достаточен, то необходимо проинформировать пользователя об этом факте. Будем делать это с помощью диода, который будет светить, когда вода есть, и мигать когда воды – нет.

Бак и стаканчики изготовлены из пищевой пластмассы, бак закрытый и герметичный.

**1.3** **Наполнение стаканчика водой**

Вода льётся из тонкого шланга присоединённого через насос к баку с водой. Когда стаканчик под краном, насос начинает качать воду некоторое время (установленное эмпирическим путём), чтобы в стаканчике оказалось необходимое количество воды (200 мл). Насос подключается отдельно, имеет свое питание и включается помощью низковольтного реле.

**1.4 Условия работы**

Условия роботы будут нормальными, то есть аппарат работает в среде с комнатной температуры, нормальным давлением и влажностью. Установлен аппарат на горизонтальной поверхности на нулевом уровне моря. Работает от бытовой электрической сети 220 В.

2. Выбор и обоснование элементной базы

При осуществлении конструирования аппарата, выделим основные компоненты, необходимые для реализации задачи:

1. электромотор, для вращения подноса со стаканчиками;
2. водяной насос, с помощью которого нальём воды;
3. электромагнитное реле;
4. микроконтроллер, который будет управлять работой аппарата.

Замечание

При поиске элементов, использовалась глобальная сеть Internet. К сожалению, большинство сайтов предлагают данные товары китайского и тайваньского производства. Наиболее популярные сайты:

1. http://chinasuppliers.alibaba.com
2. http://www.made-in-china.com

**2.1** **Электромотор**

Воспользуемся шаговым мотором.[8-17] Они позволяют на заданном шаге вращать ротор в нужное положение, благодаря подаче сигнала на ту или иную катушку возбуждения.

В зависимости от угла, который мы хотим получить необходимо, задать количество шагов.

Изучив характеристики шаговых моторов, предлагаемые как нашими производителями, так и зарубежными, я пришёл к выводу воспользоваться шаговым мотором российского производства НПФ «Электроприбор»[13]; рассмотрим серии FL20STH и FL28STH (рисунок 2.1).

Рисунок 2.1 – Вид шагового мотора серии FL20STH (справа), FL28STH (слева)

Таблица 2.1 – Технические характеристики моторов

|  |  |
| --- | --- |
| Величина полного шага, град | 1,8 |
| Погрешность углового шага, град | ±0,09 |
| Погрешность сопротивления обмоток двигателя, % | 10 |
| Погрешность индуктивности обмоток двигателя, % | 20 |
| Максимальное радиальное биение вала двигателя, мм | 0,02 |
| Максимальное осевое биение вала двигателя, мм | 0,08 |
| Максимальная допустимая осевая нагрузка на валу, Н | 10 |
| Максимальная допустимая радиальная нагрузка на валу, Н | 28 |
| Температурный диапазон эксплуатации от минус 20oС до плюс 50oС |

Таблица 2.2 – Технические характеристики моторов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование | Рабочий ток/ фаза | Крутящий момент | Момент инерции ротора | Вес |
|  | А | кг\*см | г\*cм2 | кг |
| FL20STH30-0604A | 0,6 | 0,18 | 0,2 | 0,06 |
| FL28STH32-0956A | 0,95 | 0,43 | 0,9 | 0,11 |
| FL28STH45-0956A | 0,95 | 0,75 | 1,2 | 0,14 |
| FL28STH51-0956A | 0,95 | 0,9 | 1,8 | 0,2 |
| FL28STH51-0674A | 0,67 | 1,2 | 1,8 | 0,2 |

Согласно приведенным техническим характеристикам (таблица 2.1 и таблица 2.2), воспользуемся мотором FL28STH32-0956A, который потребляет приемлемый ток, легок и достаточно дешев.

**2.2** **Водяной насос**

Выберем, обычный небольшой водяной насос [8, 9, 12, 18, 19, 20], с помощью шлангов будем закачивать воду в стаканчики. Рассмотрим несколько моделей насосов.

Рисунок 2.2 – Вид насоса модели 2013

Таблица 2.3 – Технические характеристики насосов

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Модель | Выход (ватт) | Диаметрвыхода (мм) | Вольтаж | Макс.способность (Л/мин) | Макс.давление (M) | Габариты (L\*W\*H)(мм) | Масса (кг) |
| 1207 | 7/12 | 13 | 50Гц или 60Гц 100/120В 200/220В | 8/10 | 0.9/1.0 | 75\*62\*95 | 0.85 |
| 2013 | 9/14 | 13 | 11/14 | 1.3/1.6 | 80\*67\*105 | 0.95 |
| 3025 | 14/20 | 24 | 15/18 | 1.8/2.2 | 85\*70\*135 | 1.40 |

Согласно техническим характеристикам насосов (таблица 2.3), будем брать насос модели 2013 14 Вт (см. рисунок 2.2), который подходит по габаритам и массе.

## Электромагнитное реле

Рассмотрим некоторые виды электромагнитных реле [21] (таблица 2.4).

Таблица 2.4 – Электромагнитные реле

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Модель | Rном. Ом | Ток | Время | Uраб. B |
| Iср. mA | Iот. mA | tср. mC | tот. mC |
| РС4.524.203 | 30 | 108 | 18 | 11 | 7 | 5...7 |
| РС4.524.214 | 36 | 95 | 15 | 11 | 7 | 5...7 |
| РС4.524.374 | 45 | 75 | 12 | 8 | 2 | 5..7 |
| РС4.524.315 | 45 | 80 | 11 | 6..8 | 2..4 | 4...8 |
| РФ4.500.421 | 40 | 86 | 12 | 9 | 4 | 5..8 |
| РС4.569.439 | 58 | 51 | 11 | 4 | 2 | 5..8 |
| РС4.524.316 | 1 600 | 10 | 1 | 6..8 | 2..4 | 12...13 |
| РС4.524.371 | 4 200 | 8 | 1 | 8 | 2 | 9..10 |
| РС4.591.003 | 330 | 21 | 5 | 8 | 2 | 38..43 |
| РС4.524.380 | 1 600 | 14 | 2 | 8 | 2 | 16..17 |
| РС4.524.379 | 1 600 | 14 | 2 | 8 | 2 | 16..17 |
| РС4.524.231 | 980 | 23 | 3 | 11 | 7 | 27...30 |
| РС4.524.230 | 3 400 | 11 | 2 | 11 | 7 | 13...15 |
| РС4.590.060 | 210 | 28 | 15 | 22 | 10 | 6 |
| РС4.524.209 | 500 | 30 | 5 | 11 | 7 | 23..32 |
| РС4.524.320 | 630 | 23 | 3 | 6..8 | 2..4 | 24..32 |
| РС4.524.319 | 630 | 23 | 3 | 6..8 | 2..4 | 24..32 |

Так как нам требуется низковольтное электромагнитное реле, выберем модель РС4.524.315, время срабатывания и отпускания у которого вполне приемлемы.

* 1. **Микроконтроллер**

Существует очень много производителей микроконтроллеров [22-30]. Их продукция различается качеством, ценой, а также, самое главное, техническими характеристиками, такими как: производительность микроконтроллера, потребляемое напряжение и ток, количество выводов, таймеров, объём памяти и так далее.

Рассмотрим некоторых производителей, продукция, которых наиболее доступна на рынке [27, 29](таблица 2.5):

1. Infineon Technologies
2. Atmel

Таблица 2.5 – Микроконтроллеры

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Device | Flash (Kbytes) | Mask ROM (Kbytes) | EEPROM (Kbytes) | RAM (Bytes) | F.max (MHz) | Vcc (V) | I/O Pins | 16-bit Timers | Watchdog |
|  | **Atmel** |
| AT80C51RD2 |  |  |  | 1280 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| AT83C5103 |  | 12 |  | 512 | 16 | 2.7-5.5 | 19 | 2 |  |
| AT83C51RB2 |  | 16 |  | 1280 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| AT83C51RC2 |  | 32 |  | 1280 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| AT83EB5114 |  | 4 |  | 256 | 24 | 3.0-3.6 | 11 | 2 | Yes |
| AT87C5103 |  |  |  | 512 | 16 | 2.7-6.0 | 19 | 2 |  |
| AT89C1051 | 1 |  |  | 64 | 24 | 2.7-6.0 | 15 | 1 |  |
| AT89C2051 | 2 |  |  | 128 | 24 | 2.7-6.0 | 15 | 2 |  |
| AT89C2051x2 | 2 |  |  | 128 | 16 | 2.7-6.0 | 15 | 2 |  |
| AT89C4051 | 4 |  |  | 128 | 24 | 2.7-6.0 | 15 | 2 |  |
| AT89C51AC2 | 32 |  | 2 | 1280 | 40 | 2.7-6.0 | 34 | 3 | Yes |
| AT89C51AC3 | 64 |  | 2 | 2304 | 60 | 2.7-6.0 | 32 | 3 | Yes |
| AT89C51ED2 | 64 |  | 2 | 2048 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| AT89C51IC2 | 32 |  |  | 1280 | 60 | 2.7-5.5 | 34 | 3 | Yes |
| AT89C51ID2 | 64 |  | 2 | 2048 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| AT89C51RB2 | 16 |  |  | 1280 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| AT89C51RC | 32 |  |  | 512 | 33 | 4.0-6.0 | 32 | 3 | Yes |
| AT89C51RC2 | 32 |  |  | 1280 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| AT89C51RD2 | 64 |  |  | 2048 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| AT89C55WD | 20 |  |  | 256 | 33 | 4.0-6.0 | 32 | 3 | Yes |
| AT89LP2052 | 2 |  |  | 256 | 20 | 2.4-5.5 | 15 | 2 | Yes |
| AT89LP4052 | 4 |  |  | 256 | 20 | 2.4-5.5 | 15 | 2 | Yes |
| AT89LS51 | 4 |  |  | 128 | 16 | 2.7-4.0 | 32 | 2 | Yes |
| AT89LS52 | 8 |  |  | 256 | 33 | 2.7-4.0 | 32 | 3 | Yes |
| AT89LS53 | 12 |  |  | 256 | 12 | 2.7-6.0 | 32 | 3 | Yes |
| AT89LS8252 | 8 |  | 2 | 256 | 12 | 2.7-6.0 | 32 | 3 | Yes |
| AT89LV55 | 20 |  |  | 256 | 12 | 2.7-5.5 | 32 | 3 |  |
| AT89S51 | 4 |  |  | 128 | 33 | 4.0-5.5 | 32 | 2 | Yes |
| AT89S52 | 8 |  |  | 256 | 33 | 4.0-5.5 | 32 | 3 | Yes |
| AT89S53 | 12 |  |  | 256 | 24 | 4.0-6.0 | 32 | 3 | Yes |
| AT89S8252 | 8 |  | 2 | 256 | 24 | 4.0-6.0 | 32 | 3 | Yes |

Таблица 2.5 – Микроконтроллеры (продолжение)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| AT89S8253 | 12 |  | 2 | 256 | 24 | 2.7-5.5 | 32 | 3 | Yes |
| T89C5115 | 16 |  | 2 | 512 | 40 | 2.7-5.5 | 20 | 2 | Yes |
| TS80C31X2 |  |  |  | 128 | 60 | 2.7-5.5 | 32 | 2 |  |
| TS80C32X2 |  |  |  | 256 | 60 | 2.7-5.5 | 32 | 3 |  |
| TS80C51RA2 |  |  |  | 512 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TS80C51RD2 |  |  |  | 1280 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TS80C52X2 |  | 8 |  | 256 | 60 | 2.7-5.5 | 32 | 3 |  |
| TS80C54X2 |  | 16 |  | 256 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TS80C58X2 |  | 32 |  | 256 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TS83C51RB2 |  | 16 |  | 512 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TS83C51RC2 |  | 32 |  | 512 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TS83C51RD2 |  | 64 |  | 1024 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TS87C51RB2 |  |  |  | 512 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TS87C51RC2 |  |  |  | 512 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TS87C51RD2 |  |  |  | 1024 | 40 | 2.7-5.5 | 32 | 3 | Yes |
| TS87C52X2 |  |  |  | 256 | 60 | 2.7-5.5 | 32 | 3 |  |
| TS87C54X2 |  |  |  | 256 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TS87C58X2 |  |  |  | 256 | 60 | 2.7-5.5 | 32 | 3 | Yes |
| TSC80251G2D |  |  |  | 1024 | 24 | 2.7-5.5 | 32 | 3 | Yes |
| TSC83251G2D |  | 32 |  | 1024 | 24 | 2.7-5.5 | 32 | 3 | Yes |
| TSC87251G2D |  |  |  | 1024 | 16 | 2.7-5.5 | 32 | 3 | Yes |
|  | **Infineon Technologies** |
| C504-L/-2R | 16 |  |  | 512 | 24 | 2.7-5.5 | 32 | 4 | Yes |
| C504-2E |  |  |  | 1028 | 20 | 2.7-5.5 | 34 | 3 | Yes |
| C505A-4E |  |  |  | 1028 | 20 | 2.7-5.5 | 34 | 3 | Yes |
| C505CA-4E/4R/2R-L/-2R |  | 16 |  | 1028 | 20 | 2.7-5.5 | 34 | 3 | Yes |
| C505L-4E |  |  |  | 512 | 20 | 2.7-5.5 | 46 | 3 | Yes |
| C508-4R/-2R/-L | 32 | 16 |  | 1280 | 20 | 2.7-5.5 | 48 | 3 | Yes |
| C508-4E | 32 | 16 |  | 1280 | 20 | 2.7-5.5 | 48 | 3 | Yes |
| C509-L |  |  |  | 3328 | 16 | 2.7-5.5 | 64 | 5 | Yes |
| C515C-L/-8R |  | 64 |  | 2304 | 10 | 2.7-5.5 | 57 | 3 | Yes |
| C515C-8E |  | 64 |  | 2304 | 10 | 2.7-5.5 | 57 | 3 | Yes |
| C515-L |  |  |  | 256 | 24 | 2.7-5.5 | 56 | 3 | Yes |
| C517A-L |  |  |  | 2304 | 18 | 2.7-5.5 | 68 | 4 | Yes |
| C868-1RG | 8 |  |  | 512 | 40 | 2.7-5.5 | 18 | 3 | Yes |
| C868-1RR | 8 |  |  | 512 | 40 | 2.7-5.5 | 18 | 3 | Yes |
| C868-1SG | 8 |  |  | 512 | 40 | 2.7-5.5 | 18 | 3 | Yes |
| C868-1SR | 8 |  |  | 512 | 40 | 2.7-5.5 | 18 | 3 | Yes |
| XC866 | 12 |  | 4 | 768 | 26,67 | 2.7-5.5 | 27 | 3 | Yes |

Проанализируем сколько выводов нам необходимо (таблица 2.6).

Из таблицы видно, что достаточно одного порта в/в для подключения внешних устройств. Для подсчета времени работы воспользуемся таймером, поэтому в микроконтроллере должен быть хотя бы один таймер/счетчик. Аппарат будет работать в условиях комнатной температуры, вполне достаточно иметь корпус, рассчитанный на коммерческое использование (0ºС–70ºС).

Таблица 2.6 – Анализ количества необходимого количества выходов в/в

|  |  |  |
| --- | --- | --- |
| Устройство | Комментарий | Необходимое количество выходов в/в |
| Насос | Работает от сети, включается через реле | 1 |
| Шаговый мотор | Питаются все четыре обмотки | 4 |
| Индикатор уровня воды | Для работы светодиода | 1 |
| Датчик уровня воды | Опрос датчика | 1 |
|  | Всего | 7 |

Будем пользоваться микроконтроллером Atmel AT89C1051, так как прост и используется в обычном DIP корпусе, обладает 1К Flash памяти, имеет достаточное количество выводов, работает на приемлемой частоте и напряжении, имеет 1 16-битный таймер.

1. **Разработка функциональной схемы**
	1. **Источник питания**

Рисунок 3.1 – Источник питания

Питается аппарат от источника 220В 50Гц, с помощью внешнего источника питания получим напряжение, значение которого не превышает 12В (рисунок 3.1). Необходимо подать на стабилизатор напряжение, имеющее пульсации в пределах 10%. Для этого воспользуемся полярным конденсатором. Рассчитаем его емкость. [31-40]

Для подстраховки от возможных отклонений напряжения в сети максимальный размер пульсаций не должен превышать 2В за период. Тогда С = 5000 мкФ.

Далее напряжение подается на трехвыводной стабилизатор напряжения 7805, с выхода которого получим постоянное напряжение в 5В.

Светодиод установлен, чтобы сигнализировать о включенном питании, резистор установлен для обеспечения необходимого тока светодиода. Так как светодиод светит при 20 мА, рассчитать сопротивление резистора не сложно: по закону Ома получим R = U / I = (5-2) / 0,02 = 150 Ом.

3.2 **Микроконтроллер**

Как было указано выше, для работы аппарата был выбран микроконтроллер Atmel AT89C1051 [29] (рисунок 3.2).

PDIP/SOIC

Рисунок 3.2 – Назначение выводов Atmel AT89C1051

Технические характеристики:

• Совместим с MCS-51™ продуктами;

• 1K байт программируемой flash памяти – рассчитанной: 1,000 запись/удаление циклов;

• 2.7V до 6V рабочий диапазон;

• 0 Hz to 24 MHz;

• 64 байт SRAM;

• 15 программируемых I/O выходов;

• Один 16-Bit Таймер/Счетчик;

• Три источника прерывания;

• Внутренний Аналоговый компаратор;

Описание:

AT89C1051 это низковольтный, высокопроизводительный CMOS 8-битный микроконтроллер с 1К байт программируемой памятью. Устройство собрано с использованием высоко плотной технологии и совместимо с индустриальным стандартом инструкций MCS-51™. Используя многослойный 8-битный CPU с памятью в монолитном чипе, делает Atmel AT89C1051 мощным микроконтроллером, обеспечивающим высокую гибкость и стоимостную эффективность решений множества ориентированных на контроль устройств.

В дополнение AT89C1051 проектировался со статической логикой для операций упавшей до нуля частоты и поддерживает два программно выбираемых энергосберегающих режима.

Подключение устройств к микроконтроллеру:

Список подключений и описание см. таблица 3.1. [42-46]

Таблица 3.1 – Подключения

|  |  |
| --- | --- |
| № ножки | Описание |
| 20 | Питание +5В |
| 19 | Р1.7 используется для подачи напряжения на одну из обмоток шагового мотора |
| 18 | Р1.6 используется для подачи напряжения на одну из обмоток шагового мотора |
| 17 | Р1.5 используется для подачи напряжения на одну из обмоток шагового мотора |
| 16 | Р1.4 используется для подачи напряжения на одну из обмоток шагового мотора |
| 15 | Р1.3 используется для запуска насоса через реле |
| 14 | Р1.2 используется для установки светодиода сигнализирующего о недостаточном уровне воды в баке. |
| 13 | Р1.1 не используется |
| 12 | Р1.0 не используется |
| 11 | Р3.7 Обеспечивает проверку уровня воды |
| 10 | Земля |
| 9 | Р3.5 не используется |
| 8 | Р3.4 не используется |
| 7 | Р3.3 не используется |
| 6 | Р3.2 не используется |
| 5 | Вход на инвертированный амплитудный осциллятор |
| 4 | Выход с инвертированного амплитудного осциллятора |
| 3 | Р3.1 не используется |
| 2 | Р3.0 не используется |
| 1 | Сброс Устанавливаем кнопку для сброса. |

**3.3 Тактовый генератор**

Используем осциллятор с частотой 24 МГц (рисунок 3.3). Ёмкость конденсаторов равна 30 пФ, что рекомендует производитель, описывая данную схему в технической документации. [29, 31-40]

Рисунок 3.3 – Тактовый генератор

**3.4** **Проверка уровня воды**

Рисунок 3.4 – Схема проверки уровня воды

Схема проверки воды в баке состоит из излучающего фотоны светодиода и принимающего фотодиода [31-41] (рисунок 3.4).

Фотодиод находиться на трубке, соединенный с баком. Он установлен на уровне соответствующий минимальному уровню воды. В трубке находиться поплавок, который перекрывает фотодиод, когда уровень воды мал.

Для работы светодиода необходимо обеспечить ток 20мА, для этого установлены резисторы. Рассчитывается он просто: по закону Ома получим

R = U / I = (5-2) / 0,02 = 150 Ом.

Схема соединена с портом микроконтроллера Р3.7, с помощью которого программно будем проверять достаточно воды в баке или нет. Если высокий уровень, то воды достаточно, а если низкий, то воды не достаточно и необходимо проинформировать об этом пользователя, с помощью светодиода, отвечающего за низкий уровень воды (он будет мигать).

**3.5** **Индикатор уровня воды**

Рисунок 3.5 – Индикатор уровня воды

Индикатор представляет собой светодиод зеленного цвета (рисунок 3.5), который будет светить, когда уровень воды достаточен, и мигать, если необходимо долить воду в бак. [31-41]

Для работы светодиода необходимо обеспечить ток 20мА, для этого установлены резисторы. Рассчитываются они просто: по закону Ома получим

R = U / I = (5-2) / 0,02 = 150 Ом

Схема присоединена к порту Р1.2, с помощью которого будем программно управлять светодиодом.

**3.6** **Схема управления шаговым двигателем**

Как указывалось выше, для работы используется шаговый двигатель российского производства FL28STH32-0956A [13]. Вот некоторые его характеристики:

* Рабочий ток 0,95А;
* Крутящий момент 0,43 кг\*см;
* Момент инерции ротора 0,9 г\*cм2;
* Вес 0,11 кг.

Для работы мотора необходимо обеспечить ток в 0,95А для этого установлены блоки усиления (рисунок 3.6). Транзистор выбран так, чтобы обеспечить необходимый ток для работы шагового двигателя, а конкретно, если двигатель потребляет 0,95А, а выход микроконтроллера 20мА, то соответственно необходимый коэффициент усиления ≈50, для его обеспечения воспользуемся схемой Дарлингтона. [31-40]

Рисунок 3.6 – Сема управления шаговым мотором

Схема присоединена к 4 портам: с Р1.4 по Р1.7. Каждый порт отвечает за свою обмотку, таким образом, программно будем подавать сигнал на ту или иную обмотку и тем самым будем раскручивать ротор мотора.

**3.7** **Схема управления насосом**

Для работы используется насос 2013 14Вт, имеющий отдельное питание от сети 220В, который включается с помощью низковольтного реле (рисунок 3.7). С помощью трубок он будет качать воду в стаканчики.

В схеме используется низковольтное реле российского производства РС4.524.315 [21] работающее от напряжения 4В и срабатывает при 80 мА, для обеспечения таких показателей достаточно внутреннего сопротивления реле.

Рисунок 3.7 – Схема управления насосом

Реле срабатывает при открытом транзисторе, который открывается подачей с порта Р1.3 логического нуля. При подаче логического нуля транзистор открывается, и ток проходит через реле, он срабатывает и запускает насос.

1. **Алгоритм работы**
	1. **Описание блок схемы**

Разработку программного обеспечения начнём с создания продуманного алгоритма, который приведён в виде блок-схемы (Приложение Б). [47, 48, 49]

Первым функциональным действием аппарата будет подсчет наполненных стаканчиков, для этого введём специальную переменную, в которую в начале работы обнулим.

Следующим действием будет проверка на наличие достаточного количества воды в баке. Для этого микроконтроллером будет опрошен фотодиодный датчик, и если окажется, что он перекрыт поплавком, то значит, уровень воды маловат и требуется долить воды в бак. Информировать пользователя о недостатке воды будет светодиод, который будет мигать. Это будет происходить за счет того, что микроконтроллер будет попеременно подавать сигнал на включение и выключение, через порт, к которому присоединён светодиод.

Если уровень воды достаточен, включим светодиод индикации уровня воды, подав сигнал на порт, к которому присоединен светодиод.

Далее проверяем, что имеются пустые стаканчики, для этого проверяем специальную переменную, которая считает количество заполненных стаканчиков. Если мы заполнили водой все стаканчики, то завершаем работу аппарата. А если все же остались пустые стаканчики, то работа аппарата продолжается.

Для позиционирования следующего стаканчика необходимо повернуть платформу, на которой они установлены. Это делается с помощью шагового мотора, который проворачивает насколько оборотов ротора и с помощь червячно-реечного механизма поворачивает платформу на определенный угол.

Когда мы знаем, что стаканчик находиться под краном, запускаем насос. Для этого микроконтроллер подаёт сигнал на соответствующий порт и выжидает некоторое время, достаточное для заполнения стаканчика водой. Когда время выходит мотор выключается.

Далее для удобства выжидается набольшая пауза, и аппарат переходит в режим заполнения следующего стаканчика.

* 1. **Описание хода разработки программного обеспечения**

Опишем ход реализации программы (Приложение В). [47, 48, 49]

Проверка синтаксиса и отладка программы осуществлялось с помощью программы фирмы Keil Software μVision v.2.04b.

Перед началом указывается адрес начала программы.

org 000h

ljmp BEGIN

Далее перечисляются необходимые переменные.

N\_GLASS:DB6H;количество стаканчиков

N\_MOTOR\_COUNT:DB2H;количество оборотов ротора мотора

N\_STEP:DB8H;шагом одного оборота ротора мотора

STEPS:DB90H, 10H, 30H, 20H, 60H, 40, 0C0H, 80H;шаги

TH\_MOTOR:DB;задержка перед следующим шагом

TL\_MOTOR:DB;задержка перед следующим шагом

TH\_PUMP:DB;время работы мотора

TL\_PUMP:DB;время работы мотора

TH\_LED:DB;задержка смены состояния светодиода

TL\_LED:DB;задержка смены состояния светодиода

TH\_PAUSE:DB;задержка для паузы

TL\_PAUSE:DB;задержка для паузы

В начале программы осуществляем настройку таймера/счетчика:

MOVP0, #0H;на выводы порта P0 - логический ноль

MOVTMOD, #01H;настройка таймера

Указывает что, таймер/счетчик будет работать в 1 режиме, то есть в этом режиме таймерный регистр имеет разрядность 16 бит. При переходе из состояния "все единицы" в состояние "все нули" устанавливается флаг прерывания от таймера TF0.

Далее начинается работа аппарата:

MOV R0, #0H;R0 - количество заполненных стаканчиков; обнуляем

NEXT\_GLASS:

JNBP3.7, LED\_WINK;если сигнала нет - воды недостаточно, мигает светодиод

Обнуляется регистр R0, в котором будет храниться количество уже заполненных стаканчиков. Снимается сигнал с порта P3.7, на котором стоит датчик уровня воды, и если оказалось что уровень нулевой, что означает отсутствие необходимого количества воды, осуществляем переход на процедуру, которая заставляет мигать индикатор уровня воды.

START:

CLRP1.2;светит светодиод

MOVDPTR, #N\_GLASS;загружаем адрес ячейки с количеством стаканчиков

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе количество стаканчиков

SUBB A, R0;контрольное вычитание

JZEXIT;если все стаканчики заполнены на выход

Снимаем сигнал с порта Р1.2, тем самым заставляем светить датчик уровня воды. Далее осуществляем проверку на наличие пустых стаканчиков, для этого делаем пробное вычитание, если все стаканчики заполнены, заканчиваем работу аппарата.

Осуществляем поворот ротора двигателя за счет подачи сигналов на порт. Осуществляем это через цикл. После каждой итерации делаем небольшую задержку, для этого заполняем регистры TH0 и TL0, и вызываем процедуру таймера.

MOTOR:

MOVDPTR, #N\_MOTOR\_COUNT;загружаем адрес ячейки с количеством оборотов ротора мотора

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе количество оборотов ротора мотора

MOVR1, A;сохраняем это число в R1

NEXT\_ROUND:

MOVDPTR, #N\_STEP;загружаем адрес ячейки с количеством шагов

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе количество шагов

MOVR2, A;сохраняем это число в R2

MOVDPTR, #STEPS;загружаем адрес ячейки с последовательностью шагов

NEXT\_STEP:

MOVR6, DPH;сохраняем адрес в регистрах R6 и R7, они пригодятся позже

MOVR7, DPL

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе очередной шаг

MOVP1, A;вывод в очередного шага в порт P1

MOVDPTR, #TH\_MOTOR;загружаем адрес ячейки с временем задержки

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время задержки

MOVTH0, A;загружаем время задержки в TH0

MOVDPTR, #TL\_MOTOR;загружаем адрес ячейки с временем задержки

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время задержки

MOVTL0, A;загружаем время задержки в TH0

CALLTIMER;вызов процедуры опроса таймера

MOVDPH, R6;выгружаем из R6 и R7 адрес ячейки текущего шага

MOVDPL, R7

INCDPTR;получение адреса ячейки следующего шага

DECR2;декремент количества шагов

MOVA, R2;пересылка в аккумулятор для контроля

JNZNEXT\_STEP;если не все шаги пройдены - продолжаем

DECR1;декремент количества поворотов ротора

MOVA, R1;пересылка в аккумулятор для контроля

JNZNEXT\_ROUND;если не все обороты сделаны - продолжаем

MOVP1, #0H;закончили работу с двигателем

Запуск насоса осуществляется с помощью подачи сигнала логического нуля на порт Р1.3, выдерживается необходимая пауза с помощью процедуры таймера и сигнал с порта снимается.

PUMP:

SETBP1.3;включение насоса

MOVDPTR, #TH\_PUMP;загружаем адрес ячейки с временем работы насоса

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время работы насосы

MOVTH0, A;загружаем время в TH0

MOVDPTR, #TL\_PUMP;загружаем адрес ячейки с временем работы насоса

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время работы насосы

MOVTL0, A;загружаем время в TL0

CALLTIMER;вызов процедуры опроса таймера

CLRP1.3;выключить насос

Происходит учёт заполненного стаканчика, регистр R0 инкрементируется. Выжидается небольшая пауза с помощью процедуры таймера и происходит переход на следующую итерацию заполнения следующего стаканчика.

INCR0;инкремент количества заполненных стаканчиков

MOVDPTR, #TH\_PAUSE;загружаем адрес ячейки с временем паузы

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время паузы

MOVTH0, A;загружаем время паузы в TH0

MOVDPTR, #TL\_PAUSE;загружаем адрес ячейки с временем паузы

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время паузы

MOVTL0, A;загружаем время паузы в TL0

CALLTIMER;вызов процедуры опроса таймера

JMPNEXT\_GLASS;переходим к заполнению следующего стаканчика

Функция таймера производит запуск установкой бита TR0. Далее опрашивает бит переполнения TF0, и если это бит устанавливается, работа таймера завершается.

TIMER:;процедура опроса таймера

SETBTR0;запускаем таймер

TIMER\_STEP:

JBTF0, TIMER\_STEP;ждем переполнения таймера

CLRTR0;выключаем таймер

RET;выход из процедуры

Процедура мигания светодиода, отвечающего за уровень воды. Заключается в выполнении включения или выключения светодиода через некоторые промежутки времени.

LED\_WINK:;процедура мигания светодиода

JNBP1.2, SET\_WINK;если светодиод - выключен, включим

SETBP1.2;выключить светодиод

JMPWAIT\_WINK;обеспечим задержку смены состояния

SET\_WINK:

CLRP1.2;включим светодиод

WAIT\_WINK:

MOVDPTR, #TH\_LED;загружаем адрес ячейки с временем задержки

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время задержки

MOVTH0, A;загружаем время задержки в TH0

MOVDPTR, #TL\_LED;загружаем адрес ячейки с временем задержки

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время задержки

MOVTL0, A;загружаем время задержки в TL0

CALLTIMER;вызов процедуры опроса таймера

JMPLED\_WINK;переход на следующий цикл

**Заключение**

При проектировании аппарата по розливу воды в стаканчики, созданном на основе микроконтроллера Atmel AT89C1051, с использованием насоса 2013 и шагового мотора FL28STH32-0956A, мы научились создавать полноценные устройства, организовывать их работу.

Работой микроконтроллера управляет микропрограмма, реализующая логику аппарата. Создавая программное обеспечение, мы получили опыт программирования на языках низкого уровня.

Несомненно, выполнение данного курсового проекта привело к пониманию функционирования микроконтроллеров и способов реализации логики разнообразных устройств.

**Список использованных источников**

1. Масааки И. Гемба кайдзен: Путь к снижению затрат и повышению качества/ пер. с англ. – М.: Альпина Бизнес Букс, 2005
2. Массааки И. Кайдзен: путь к успеху японских компаний/ Пер. с англ. – М.: Альпина Бизнес Букс, 2004
3. Хэрри М., Шредер Р. 6 SIGMA. Концепция идеального менеджмента. – М. : «Эксмо», 2003
4. Ньюэлл Ф. Почему не работают системы CRM. Как добиться успеха, позволив клиентам управлять отношениями с вашей компанией. – М.: Добрая книга, 2004
5. Митник Кевин. Искусство обмана. – М.: Компания АйТи, 2004
6. Голдрад Элияху, Кокс Джефф. Цель: процесс непрерывного совершенствования. – Мн. «Попурри», 2004
7. Делл Майкл. От Dell без посредников: стратегии, которые совершили революцию в компьютерной индустрии – М. 2004
8. http://chinasuppliers.alibaba.com
9. http://www.made-in-china.com
10. http://www.commlinx.com.au
11. http://www.discovercircuits.com
12. http://www.directindustry.com
13. http://electroprivod.ru
14. http://www.cs.uiowa.edu
15. http://www.anaheimautomation.com
16. http://stepmotor.ru
17. http://www.commlinx.com.au
18. http://www.caopump.com
19. http://www.legoeducationstore.com
20. http://www.allproducts.com.tw
21. http://radio-spravochnik.by.ru/
22. http://www.futurlec.com/
23. http://www.st.com
24. http://www.keil.com
25. http://www.cpu-world.com
26. http://microcontroller.com
27. http://www.infineon.com
28. http://www.ti.com
29. http://www.atmel.com
30. http://www.maxim-ic.com
31. П. Хоровиц, У. Хилл. Искусство схемотехники: в 2-х томах./Перевод с англ. под ред. М. В. Гальперина. – М.: Мир, 1986
32. Т. Мотоока, Х. Хорикоси, М. Сакаути, Х. Танака, Х. Танака, Т. Сайто. Компьютеры на СБИС. В двух книгах./Перевод с японского под ред. В. М. Кисельникова. – М.:Мир, 1988
33. Г.И.Пухальский,Т.Я.Новосельцева.- Цифровые устройства: Учебное пособие для втузов.- СПб.: Политехник, 1996 г.
34. http://www.engineering-sample.com/
35. http://cxem.net
36. http://kazus.ru
37. http://www.techlib.com
38. http://www.uoguelph.ca
39. http://www.fujitsu.com
40. http://www.usdigital.com
41. http://www.ledtronics.com

**Приложения**

**Приложение А**

**Функциональная схема аппарата**

**Приложение Б**

**Блок-схема алгоритма**

**Приложение В**

**Программа**

NAME PROGRAM

org 000h

ljmp BEGIN

N\_GLASS:DB6H;количество стаканчиков

N\_MOTOR\_COUNT:DB2H;количество оборотов ротора мотора

N\_STEP:DB8H;шагом одного оборота ротора мотора

STEPS:DB90H, 10H, 30H, 20H, 60H, 40, 0C0H, 80H;шаги

TH\_MOTOR:DB;задержка перед следующим шагом

TL\_MOTOR:DB;задержка перед следующим шагом

TH\_PUMP:DB;время работы мотора

TL\_PUMP:DB;время работы мотора

TH\_LED:DB;задержка смены состояния светодиода

TL\_LED:DB;задержка смены состояния светодиода

TH\_PAUSE:DB;задержка для паузы

TL\_PAUSE:DB;задержка для паузы

LED\_WINK:;процедура мигания светодиода

JNBP1.2, SET\_WINK;если светодиод - выключен, включим

SETBP1.2;выключить светодиод

JMPWAIT\_WINK;обеспечим задержку смены состояния

SET\_WINK:

CLRP1.2;включим светодиод

WAIT\_WINK:

MOVDPTR, #TH\_LED;загружаем адрес ячейки с временем задержки

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время задержки

MOVTH0, A;загружаем время задержки в TH0

MOVDPTR, #TL\_LED;загружаем адрес ячейки с временем задержки

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время задержки

MOVTL0, A;загружаем время задержки в TL0

CALLTIMER;вызов процедуры опроса таймера

JMPLED\_WINK;переход на следующий цикл

TIMER:;процедура опроса таймера

SETBTR0;запускаем таймер

TIMER\_STEP:

JBTF0, TIMER\_STEP;ждем переполнения таймера

CLRTR0;выключаем таймер

RET;выход из процедуры

BEGIN:

MOVP0, #0H;на выводы порта P0 - логический ноль

MOVTMOD, #01H;настройка таймера

MOV R0, #0H;R0 - количество заполненных стаканчиков; обнуляем

NEXT\_GLASS:

JNBP3.7, LED\_WINK;если сигнала нет - воды недостаточно, мигает светодиод

START:

CLRP1.2;светит светодиод

MOVDPTR, #N\_GLASS;загружаем адрес ячейки с количеством стаканчиков

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе количество стаканчиков

SUBB A, R0;контрольное вычитание

JZEXIT;если все стаканчики заполнены на выход

MOTOR:

MOVDPTR, #N\_MOTOR\_COUNT;загружаем адрес ячейки с количеством оборотов ротора мотора

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе количество оборотов ротора мотора

MOVR1, A;сохраняем это число в R1

NEXT\_ROUND:

MOVDPTR, #N\_STEP;загружаем адрес ячейки с количеством шагов

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе количество шагов

MOVR2, A;сохраняем это число в R2

MOVDPTR, #STEPS;загружаем адрес ячейки с последовательностью шагов

NEXT\_STEP:

MOVR6, DPH;сохраняем адрес в регистрах R6 и R7, они пригодятся позже

MOVR7, DPL

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе очередной шаг

MOVP1, A;вывод в очередного шага в порт P1

MOVDPTR, #TH\_MOTOR;загружаем адрес ячейки с временем задержки

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время задержки

MOVTH0, A;загружаем время задержки в TH0

MOVDPTR, #TL\_MOTOR;загружаем адрес ячейки с временем задержки

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время задержки

MOVTL0, A;загружаем время задержки в TH0

CALLTIMER;вызов процедуры опроса таймера

MOVDPH, R6;выгружаем из R6 и R7 адрес ячейки текущего шага

MOVDPL, R7

INCDPTR;получение адреса ячейки следующего шага

DECR2;декремент количества шагов

MOVA, R2;пересылка в аккумулятор для контроля

JNZNEXT\_STEP;если не все шаги пройдены - продолжаем

DECR1;декремент количества поворотов ротора

MOVA, R1;пересылка в аккумулятор для контроля

JNZNEXT\_ROUND;если не все обороты сделаны - продолжаем

MOVP1, #0H;закончили работу с двигателем

PUMP:

SETBP1.3;включение насоса

MOVDPTR, #TH\_PUMP;загружаем адрес ячейки с временем работы насоса

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время работы насосы

MOVTH0, A;загружаем время в TH0

MOVDPTR, #TL\_PUMP;загружаем адрес ячейки с временем работы насоса

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время работы насосы

MOVTL0, A;загружаем время в TL0

CALLTIMER;вызов процедуры опроса таймера

CLRP1.3;выключить насос

INCR0;инкремент количества заполненных стаканчиков

MOVDPTR, #TH\_PAUSE;загружаем адрес ячейки с временем паузы

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время паузы

MOVTH0, A;загружаем время паузы в TH0

MOVDPTR, #TL\_PAUSE;загружаем адрес ячейки с временем паузы

CLR A;очищаем аккумулятор

MOVC A,@A+DPTR;в аккумуляторе время паузы

MOVTL0, A;загружаем время паузы в TL0

CALLTIMER;вызов процедуры опроса таймера

JMPNEXT\_GLASS;переходим к заполнению следующего стаканчика

EXIT:

END