

УДК 681.2-5

Программирование 16-разрядных микроконтроллеров RL78 компании Renesas на примере разработки контроллера солнечной батареи

Ибрагимов С.А.¹, Семиляк А.И.¹

¹ФГБОУ ВПО «Дагестанский Государственный Технический Университет», 367015, Республика Дагестан, г. Махачкала, проспект Имама Шамиля, 70. Тел./факс: (8722) 623761, e-mail: dstu@dstu.

Энергия имеет многочисленные экономические и технические аспекты. Человеческому обществу нужна энергия, потребность в которой возрастает с каждым днем. Загрязнение окружающей среды и глобальное потепление приводят к глобальным изменениям климата. Дефицит энергии и ограниченность топливных ресурсов показывают неизбежность перехода к альтернативным источникам энергии, основой которых служит энергия Солнца.

Данная статья открывает цикл статей посвященных техническим аспектам эффективного использования солнечной энергии.

Для увеличения КПД солнечной батареи необходимо правильное позиционирование модулей солнечной батареи относительно расположения Солнца на небосклоне, то есть необходимо поворачивать батарею вслед за движением Солнца. Для выполнения поставленной задачи производится разработка системы позиционирования солнечной батареи на основе 16-разрядного микроконтроллера RL78 компании Renesas.

В первой статье, посвященной разработке системы позиционирования солнечной батареи, рассмотрены аспекты программирования аналого-цифрового преобразователя 16-разрядного микроконтроллера RL78 компании Renesas.

Ключевые слова: аналого-цифровой преобразователь, датчик, микроконтроллер, программа, фотодатчик, компилятор, стартовый комплект, солнечная батарея,

Programming 16-bit microcontrollers of Renesas RL78 an example of developing solar controller

Ibragimov S. A.¹, Semilyak A. I.¹

¹"Dagestan State Technical University", 367015, Republic of Dagestan, Makhachkala, the prospectus of Imam Shamil, 70 Tel. / Fax (8722) 623761, e-mail: dstu@dstu.

Energy has numerous economic and technical aspects. Human society needs energy, the demand for which is growing every day. Pollution and global warming lead to global climate change. Energy shortages and limited fuel resources indicate the inevitability of the transition to alternative energy sources, which are not based solar energy.

This article begins a series of articles devoted to the technical aspects of the efficient use of solar energy.

To increase the efficiency of solar cell must be the correct positioning of the solar cell modules on the location of the sun on the horizon, that is, it is necessary to rotate the battery following the movement of the sun. To accomplish this, covers the development of the positioning system of solar cell based on a 16-bit microcontroller RL78 of Renesas.

In the first article devoted to the development of solar cell positioning system, reviewed aspects of programming analog-to-digital converter 16-bit microcontroller RL78 of Renesas

Keywords: analog-to-digital converter, a sensor, a microcontroller, a program, a photosensor, a compiler, a starter kit, solar battery.

Контроллер солнечной батареи является устройством со слежением за точкой максимальной мощности солнечной батареи. Солнечная батарея своей рабочей поверхностью должна быть обращена к солнцу под прямым углом. В таком положении батарея должна находиться как можно больший период времени. Для увеличения продолжительности правильного позиционирования модулей батареи относительно солнца необходимо поворачивать батарею вслед за движением Солнца.

Установка панели в нужном положении позволит получить солнечную батарею с более высоким КПД. При повороте солнечных элементов синхронно с угловой скоростью вращения Земли, мы сможем располагать солнечные элементы под прямым углом к Солнцу. Вращение осуществляется механической системой следящего устройства с применением шагового двигателя. В качестве двух датчиков, информирующих систему о расположении Солнца, используются фотодиоды. Микроконтроллер, используя информацию от фотодиодов, выдает ведущему механизму информацию о положении Солнца. Под воздействием управляющего сигнала панель устанавливается в нужном направлении. Как только солнце сместится к западу, микропроцессорный регулятор запустит шаговый двигатель до тех пор, пока снова не восстановится нужное направление панели на солнце. [2]

В разработке используется стартовый комплект YRPBRL78G13 с установленным 16-разрядным микроконтроллером RL78 (R5F100LEA) компании Renesas. В микроконтроллере используется высокопроизводительное ядро 78KOR, которое обеспечивает производительность до 41 MIPS. Микроконтроллер RL78 имеет также улучшенные характеристики периферийных устройств. Основная особенность микроконтроллеров RL78- очень низкое энергопотребление. В рабочем режиме оно составляет от 70 до 140 мкА/ МГц.

Кроме того, в микроконтроллере RL78 реализован целый набор энергосберегающих режимов. [4]

Для разработки программного обеспечения контроллера используются программные средства поддержки Applilet3 и IAR Embedded Workbench. Applilet3 предназначена для генерации шаблонов исходного текста с уже включенной первичной инициализацией микроконтроллера. Applilet3 имеет удобный графический интерфейс и значительно облегчает работу программиста.

Интегрированная среда IDE IAR EMBEDDED WORKBENCH, входящая в стартовый набор, позволяет создавать полноценные программные проекты для микроконтроллеров семейства RL78. Программное обеспечение от фирмы IAR - набор программно-аппаратных средств, предназначенный для разработки и отладки систем на базе микроконтроллеров, в том числе и семейства RL78. Интегрированная среда IDE IAR EMBEDDED WORKBENCH представляет собой объединение внутрисхемного эмулятора, программного отладчика симулятора, Си компиляторов, текстового редактора, менеджера проектов и программатора в составе одной интеллектуальной среды разработки. Интерфейс программной среды от IAR унифицирован и поддерживает все этапы разработки программного обеспечения - от написания исходного текста программы до ее компиляции и отладки.

Фотодатчик как первичный измерительный преобразователь является важнейшей частью измерительной системы и системы управления устройства позиционирования солнечной батареи. Датчик преобразует световой поток в оптическом диапазоне в электрические сигналы. Свет является излучением высокой частоты и поэтому световой поток характеризуется такими энергетическими характеристиками как мощность (Вт) и световой поток (лм), который и регистрируется фотоприемником. В настоящее время наиболее распространенными типами фотодатчиков являются полупроводниковые фотодиоды и фототранзисторы. [1]

Режим работы фотодиода при совпадении направления фототока фотодиода с направлением обратного тока через р-п переход называется фотогальваническим режимом. При наличии внешнего обратного напряжения при изменении освещенности меняется обратный ток через р-п переход, что соответствует фотодиодному режиму работы. Как показывает опыт для обеспечения информирования системы о расположении Солнца возможно использование фотогальванического режима работы фотодиода.

Макетный образец для снятия данных с фотодатчика представлен на рис.1

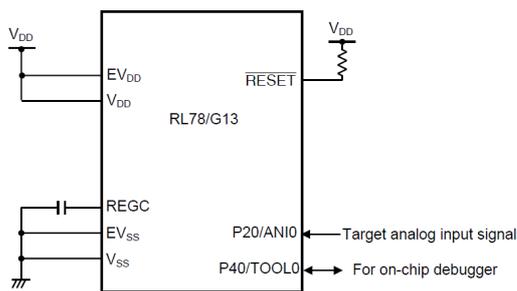


Рис.1 Макетный образец устройства Рис.2. Схема подключения фототранзистора

Проведенные исследования показали, что фотодиоды выдают одинаковые данные вне зависимости от угла падения световых лучей, поэтому было принято решение их замены на фототранзисторы. Фототранзисторы в зависимости от того, под каким углом падает на них свет, на выходе выдают различные значения напряжения, что не наблюдалось у фотодиодов. Это различие играет достаточно важную роль и позволяет решить задачу правильного позиционирования модулей батареи относительно солнца.

Разработка алгоритма программы

Программа использует фототранзистор, подсоединенный к аналоговому входу микроконтроллера. В зависимости от яркости света изменяется уровень напряжения на аналоговом входе микроконтроллера. На выходе аналого-цифрового преобразователя выставляется 10-ти разрядное значение, которое отображается на индикаторе. Формат отображения величины светового потока в десятичном представлении. В качестве

аналогового входа используется канал 0 АЦП (рис.2). При установке времени аналого-цифрового преобразования большего, чем время наиболее длинного преобразования получается наиболее точный результат (33 наносекунды для 32 МГц тактирования периферии микроконтроллера).

В данном примере программа осуществляет цифровое преобразование аналогового напряжения, который вводится через контакт ANIO, используя триггер программного обеспечения и последовательный режим преобразования АЦП. Ожидается конец аналого-цифрового преобразования в режиме HALT. После завершения преобразования, образец кода сдвигает результат преобразования на 6 бит вправо и помещает результат во внутреннее ОЗУ в стартовом наборе RL78/G13.

Программа включает следующие важнейшие части, представляющие процесс аналого-цифрового преобразования: - код инициализации программы (включая инициализацию аналого-цифрового преобразования), которая вызывается перед стартом главной программы main (); - цикл главной программы, которая читает результат аналого-цифрового преобразования и отображает его на дисплее; - подпрограмму, генерируемую программой Applilet для старта, остановки и чтения преобразования.

Программа также включает секции, которые напрямую не относятся к процессу аналого-цифрового преобразования: - подпрограмма отображения данных на ЖКИ;

- подпрограмма, генерируемая программой Applilet, для обработки старта таймера, остановки и установки интервала; - подпрограмма с кодом пользователя для обработки прерываний кнопок (программа Applilet генерирует шаблон программы обслуживания прерываний, в который добавляется код пользователя).

Представленные алгоритмы описывают только инициализацию, главную программу и подпрограммы связанные с аналого-цифровым преобразованием. Алгоритм не включает инициализацию для ЖКИ, передачу данных в ЖКИ, подпрограммы связанные с таймером и подпрограммы обслуживания прерываний, а также прерывания от переключателей.

Старт программы и инициализации

Для программ написанных на языке C для микроконтроллера с ядром 78K0R файл с объектным кодом получается после обработки линковщиком программы пользователя. Этот файл обеспечивает код запуска для программ на C. Этот код запуска вызывается функцией с именем hdwinit(); пользователь может расположить здесь код для инициализации периферийного оборудования.

Когда программа Applilet3 генерирует программу на C для ядра 78K0R, то она автоматически добавляет функцию hdwinit() в программу пользователя и вызывает функцию SystemInit(). Функция SystemInit() в процессе работы вызывает подпрограммы

инициализации для каждого периферийного узла. Когда функция `hdwinit()` заканчивает свою работу код запуска вызывает функцию `main()` в программе пользователя. Таким образом, при старте функции `main()` происходит инициализация периферии микроконтроллера для прерываний от кнопки и таймера. Функция `main()` не нуждается в вызове индивидуальных подпрограмм инициализации периферии.

Функция `SystemInit()` вызывает подпрограмму, которая компонует функцию аналого-цифрового преобразования, шаги по выполнению которой приведены ниже.

1. При инициализации АЦП осуществляются следующие действия:

- вывод P20/AN10 используется для аналогового входа;

- устанавливается режим последовательного преобразования аналого-цифрового преобразования;

- при А/Ц преобразовании используется триггер программного обеспечения;

- для окончания преобразования используется прерывание (INTAD).

2. Программа устанавливает бит `ADCS` в регистре `ADM0` в 1 (для начала аналого-цифрового преобразования) и выполняет инструкцию `HALT`, чтобы перевести микроконтроллер в режиме `HALT` и ожидать прерывания для окончания аналого-цифрового преобразования.

3. После завершения преобразования входного напряжения от контакта `AN10`, АЦП передает результат аналого-цифрового преобразования в регистр `ADCR` и генерирует конец прерывания.

4. После освобождения из режима `HALT` в ответ на конец преобразования, программа читает результат аналого-цифрового преобразования из регистра `ADCR`, сдвигает результат на 6 бит вправо, и сохраняет сдвинутые данные во внутреннем ОЗУ в `RL78/G13`.

5. Микроконтроллер возвращается в режим `HALT` и ждет конца А/Ц преобразования.

Алгоритм настройки АЦП `R_ADC_CREATE` приведен на рис.3.

При настройке АЦП выполняются следующие шаги:

1. Установкой бита `ADCEN = 1` осуществляется запуск сигнала тактовой частоты.

2. Записью в регистр `ADM0` значения `00H` осуществляется остановка АЦП.

3. Установкой бита `ADIF` в 0 осуществляется очистка флага запроса прерывания.

4. Установкой бита `ADMK` в 1 отключается окончание прерывания аналого-цифрового преобразования.

5. Устанавливается уровень приоритета прерывания в 3 для АЦП. Для этого биты `ADPR0` и `ADPR1` устанавливаются в 1.

6. Устанавливается время преобразования 33 нс, для этого в регистр `ADM0` заносится значение `28H`, биты `FR2 ... FR0` устанавливаются в значение `101B`: `fCLK/5`

(fCLK= 32 МГц). Выбирается режим выбора, для этого бит ADMD устанавливается в значение 0, в регистр ADM1 заносится значение 00H. Выбирается режим триггера программного обеспечения, для этого биты ADTMD1 и ADTMD0 = 00B.

7. Производится установка эталонного источника напряжения – в данном блоке осуществляется выбор регистра результата преобразования АЦП и выбор эталонного источника напряжения это VDD и VSS. Выбор осуществляется установкой значений FFH в регистр ADM2. Разрешение в 10 бит выбирается установкой бита ADTYP в 1, биты ADREFP1 и ADREF0 = 00B для питания от VDD, бит ADREFM =0 для питания от VSS.

8. Производится установка верхнего/нижнего пределов сравнения результата преобразования: на данном этапе результат преобразования сравнивается с верхним пределом, который задается в регистре ADUL = FFH и с нижним пределом, который задается в регистре ADLL =00H.

9. Задается канал аналогового ввода, при этом регистр ADS устанавливается в 00H биты ADS4 ... ADS0 = 00000B, бит ADISS =0.

После инициализации код запуска вызывает функцию main(). Функция main() вызывает функцию AD_Start() для старта операции аналого-цифрового преобразования и устанавливает режим отображения в 10-ом виде. Функция main() затем отображает сообщение на ЖКИ.

Программа читает аналого-цифровое преобразование внутри переменной результата преобразования. Результат преобразования лежит между 0 и 0xFF (1023), когда преобразование осуществляется с разрешением 10 бит. [3]

Результат измерения отображается на экране в десятичном виде (0 - 1023).

При выполнении главной программы выполняются следующие шаги:

- переменная result_buffer, которая сохраняет результаты аналого-цифрового преобразования устанавливается в 00h;
- запрет прерываний IE =0;
- включение компаратора напряжений аналого-цифрового преобразования;
- устанавливается время стабилизации компаратора, необходимое для того, чтобы прошло определенное количество времени для корректной работы компаратора иначе результаты преобразования будут ошибочными;
- запуск функции аналого-цифрового преобразования;
- переключение в режим HALT;
- установка флага запроса конца прерывания аналого-цифрового преобразования, ADIF = 1;
- сохранение результатов аналого-цифрового преобразования в ОЗУ;

- установка флага запроса прерывания окончания аналого-цифрового преобразования, ADIF = 0.

Алгоритм программы main представлен на рис.4.

Подпрограмма AD_Read(USHORT*buffer) считывает результат преобразования АЦП, который запоминается в определенном месте – в параметрическом буфере. Данные относятся к классу “unsigned short” – 16-разрядная ячейка памяти хранит данные в диапазоне 0- 65535. Подпрограмма AD_Read получает текущее значение регистра ADCR, который сохраняет результат последнего преобразования АЦП. Биты регистра от 15 до 6 содержат 10-разрядный результат. Подпрограмма выравнивает результат, сдвигая его на шесть бит вправо, 9 бит вплоть до 0. Этот результат сдвига величиной от 0 до 0x3FF (1023) запоминается в параметрическом буфере. [5] АЦП работает непрерывно, обновляя содержимое регистра ADCR результатом последнего преобразования. Этот результат остается в регистре ADCR до тех пор, пока не закончится следующее преобразование, пользователь может прочитать регистр ADCR каждый раз после получения последнего преобразования.



Рис.4. Главная программа

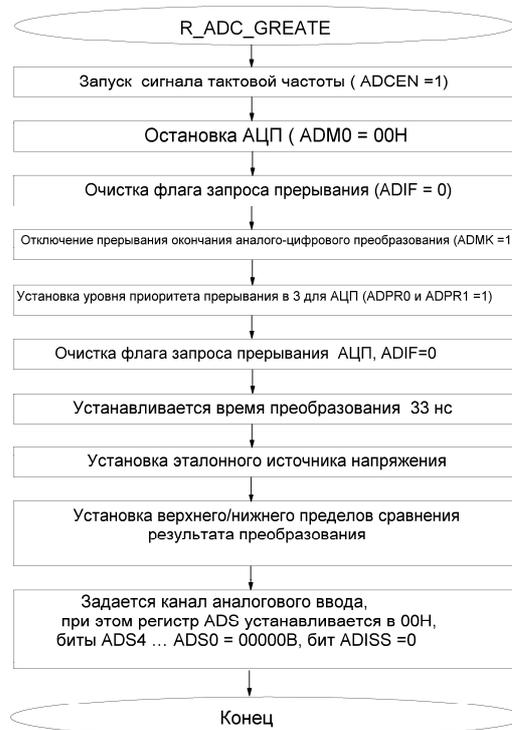


Рис.3. Алгоритм настройки АЦП

Выводы

Результатом работы является разработанное аппаратное и программное обеспечение схемы слежения за точкой максимальной мощности солнечной батареи для контроллера солнечной батареи. Изготовлен макетный образец устройства (рис.1). Опытная эксплуатация показала, что применение фототранзистора в качестве датчика контроллера солнечной батареи обеспечивает возможность слежения за точкой максимальной мощности солнечной батареи и позволяет повысить ее КПД.

Возможным направлением улучшения качественных характеристик солнечной батареи является применение других датчиков и других алгоритмов слежения за точкой максимальной мощности солнечного излучения.

Список литературы.

1. В.Б. Топильский. Схемотехника измерительных устройств. Москва. Биномю Лаборатория знаний, 2010.
2. Лабейш В.Г. Нетрадиционные и возобновляемые источники энергии: Учебное пособие. – СПб.:СЗТУ, 2003.-79 с.
3. Стюарт Болл Р. Аналоговые интерфейсы микроконтроллеров. Москва. Издательский дом “Додэка-XXI”, 2007.
4. Quick Start Guide Renesas Promotion Board for RL78/G13
5. RL78/G13. User’s Manual: Hardware 16-Bit Single-Chip Microcontrollers.