БГУИР

Кафедра ЭВМ

Отчет по лабораторной работе № 5

Тема: «Аналоговый ввод-вывод. АЦП. Компаратор. Потенциометр. Сенсорная клавиатура»

Вариант 5

Выполнил: студент группы 950505

Киреев Ю.В.

Проверил: ассистент каф ЭВМ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Богдан Е.В.

Минск 2022

**1 Цели работы**

В ходе выполнения лабораторной работы необходимо изучить принципы работы с аналоговыми сигналами на базе микроконтроллера MSP430F5529.

**2 Исходные данные к работе**

Для выполнения лабораторной работы используется плата MSP-EXP430F5529 с использованием среды разработки Code Composer Studio. В процессе выполнения работы требуется написать программу, которая непрерывно сравнивает сигнал потенциометра и термодатчика. Если уровень сигнала потенциометра включать LED3, иначе LED1.

Не допускается использовать иные заголовочные файлы, кроме msp430, а также использовать высокоуровневые библиотеки.

**3 Теоретические сведения**

Микроконтроллер MSP430F5529 содержит 16-канальный АЦП ADC12A, обладающий следующими возможностями:

- максимальная скорость преобразования — более 200 тысяч выборок/с; - выборка и сохранение с программируемым периодом выборки;

- запуск преобразования программно или от таймера;

- программно конфигурируемый внутренний генератор опорного напряжения (1.5, 2.0, 2.5 В);

- программный выбор внешнего или внутреннего источника опорного напряжения;

- 12 отдельно конфигурируемых внешних входных каналов;

- каналы для внутреннего датчика температуры, Vсс и внешних опорных напряжений;

- независимый для каждого канала опорный источник, как положительного, так и отрицательного напряжения;

- конфигурируемый источник тактового сигнала;

- четыре режима преобразования: одноканальный, повторно-одноканальный, последовательный и повторно-последовательный;

- ядро АЦП и ИОН могут выключаться независимо друг от друга; - быстрое декодирование 18 источников прерываний АЦП;

- сохранение 16 результатов;

- ядро АЦП автоматически отключается, если не идет цикл измерения. Цифровое значение измеряемой величины вычисляется по формуле:

NADC = 4095 \* (VIN – VR-) / (VR+ – VR- ),

где VIN–измеряемый(входной)сигнал, VR- и VR+ источникиопорного напряжения.

2

Структура АЦП представлена на рис. 3.1, описание полей регистров приведено в таблицах 3.1 — 3.4.

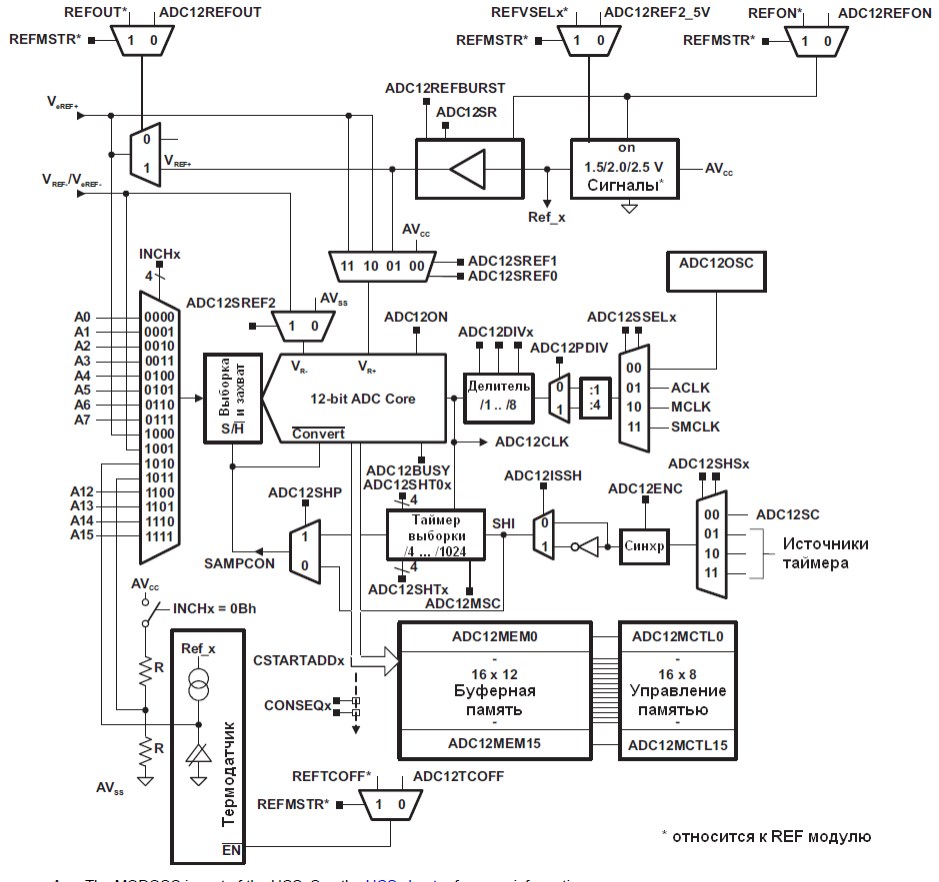


Рисунок 3.1 – Структура АЦП ADC12A

Выбор тактирования возможен из 4 источников: MCLK, SMCLK, ACLK, ADC12OSC. Тактовая частота может делиться /1, /2, … /8, /12, … /32 раза при использовании двух делителей: ADC12DIVx, ADC12PDIV. Модуль ADC12OSC связан со внутренним генератором 5 МГц. В АЦП используется независимый генератор опорного напряжения (1.5, 2.0, 2.5 В), который может быть доступен как внутри АЦП, так и снаружи (выход VREF+). Сигнал REF ON включает/отключает генератор, REFBURST = 1 включает буферный усилитель автоматически только когда используется АЦП; = 0 — постоянно.

Цикл преобразования начинается по фронту входного сигнала SHI (выбирается из ADC12SC и сигналов от таймера). Сигналы ADC12RES задают

3

точность (8, 10 или 12 бит), что требует 9, 11 или 13 тактов генератора, ADC12DF определяет знаковое или беззнаковое хранение результата, SAMPCON управляет периодом выборки и началом преобразования. Преобразование начинается по спаду сигнала ADC12SHP – 2 режима преобразования (расширенный и импульсный).

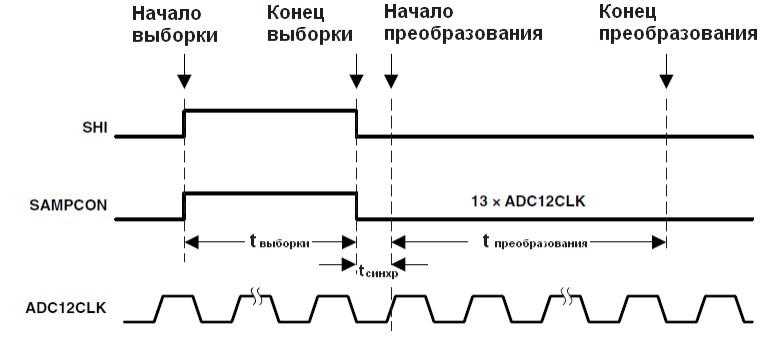
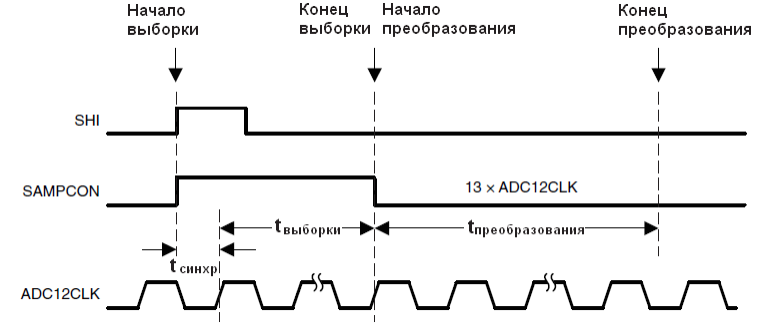


Диаграмма работы в расширенном режиме представлена на рисунке 3.2. Режим задается сигналами ADC12SHP = 0, SAMPCON = SHI.

Рисунок 3.2 – Расширенный режим

В импульсном режиме преобразования (ADC12SHP = 1) сигнал SHI определяет начало выборки, биты ADC12SHT0x и ADC12SHT1x регистра ADC12CTL0 задают длительность времени выборки:

Рисунок 3.3 – Импульсный режим

4

Таблица 3.1 – Регистры АЦП

Регистр

ADC12CTL0

ADC12CTL1

ADC12CTL2

ADC12IFG

ADC12IE

ADC12IV

ADC12MCTL0… ADC12MCTL15

ADC12MEM0...

ADC12MEM15

Адрес Назначение

0700h Регистры управления

0702h

0704h

070Ah Флаги прерываний

070Ch Разрешение прерываний

070Eh Вектор прерываний

0710h… Управление памятью 071Fh

0720h Память

073Dh

Таблица 3.2 – Поля регистра управления АЦП ADC12CTL0

Определение Биты Поле Назначение флагов в

msp430f5529.h

12-15 ADC12SHT1x Количество циклов ADC12CLK (длительность ADC12SHT1\_0 ... выборки) для ADC12MEM8 … ADC12MEM15 ADC12SHT1\_15

8-11 ADC12SHT0x

7 ADC12MSC

6 ADC12REF2\_5V

5 ADC12REFON 4 ADC12ON

3 ADC12OVIE 2 ADC12TOVIE 1 ADC12ENC

0 ADC12SC

Количество циклов ADC12CLK (длительность выборки) для ADC12MEM0 … ADC12MEM7 - 4 / 8 / 16 / 32 / 64 / 96 / 128 / 192 / 256 / 384 / 512 / 768 / 1024

Для режима последовательности и непрерывного: 0 – для запуска требуется фронт SHI сигнала, 1 – первый по фронту SHI, потом автоматически по окончании цикла начинается новый

Напряжение опорного генератора (ADC12REFON должен быть установлен) 0 – 1.5, 1 – 2.5 В

Включение опорного генератора Включение АЦП

Разрешение прерывания OV Разрешение прерывания TOV Разрешение измерения

Программный запуск выборки и преобразования. Сбрасывается автоматически

ADC12SHT0\_0 ... ADC12SHT0\_15

ADC12MSC

ADC12REF2\_5V

ADC12REFON ADC12ON ADC12OVIE ADC12TOVIE ADC12ENC

ADC12SC

5

Таблица 3.3 – Поля регистров управления АЦП ADC12CTL1-2

**Регистр**

**Биты** **Поле** **Назначение**

**Определение флагов в msp430f5529.h**

12-15 ADC12CSTARTADDx

10-11 ADC12SHSx

9 ADC12SHP

8 ADC12ISSH

ADC12CTL1

5-7 ADC12DIVx

3-4 ADC12SSELx

1-2 ADC12CONSEQx

0 ADC12BUSY

Стартовый адрес записи результата

Источник сигнала запуска

Выбор источника сигнала SAMPCON (расширенный или импульсный режим)

Инвертирование сигнала запуска

Делитель тактовой частоты

Выбор тактового сигнала

Режим

Индикатор активного режима

ADC12CSTARTADD\_0 ... ADC12CSTARTADD\_15

ADC12SHS\_0 ... ADC12SHS\_3

ADC12SHP

ADC12ISSH

ADC12DIV\_0 ... ADC12DIV\_7

ADC12SSEL\_0 ... ADC12SSEL\_3

ADC12CONSEQ\_0 ... ADC12CONSEQ\_3

ADC12BUSY

8 ADC12PDIV Деление тактовой частоты на 4 ADC12PDIV

7 ADC12TCOFF

4-5 ADC12RES

3 ADC12DF 2 ADC12SR

ADC12CTL2

1 ADC12REFOUT

0 ADC12REFBURST

Отключение температурного датчика

Точность (8, 10, 12 бит)

Результат - знаковый

Буфер работает на частоте 0: 200К , 1: 50К выборок / с

Выход опорного напряжения

Режим работы буферного усилителя опорного напряжения: постоянный (0) или автоматический (1)

ADC12TCOFF

ADC12RES\_0 ... ADC12RES\_3

ADC12DF ADC12SR

ADC12REFOUT

ADC12REFBURST

Битами CONSEQx выбирается режим. Одноканальный режим: однократное измерение одного канала. Результат записывается в регистр ADC12MEMx, определяемый битами CSTARTADDx. Повторно-одноканальный: непрерывные измерения одного канала. Результат записывается в ADC12MEMx, определяемый битами CSTARTADDx. Необходимо читать результат после каждого измерения, иначе данные потеряются. Последовательный режим: однократное измерение последовательности каналов. Запись начинается с регистра ADC12MEMx,

6

определяемого битами CSTARTADDx. После каждой записи происходит инкремент номера регистра. Измерение прекращается, когда происходит измерение канала с установленным битом ADC12EOS. Повторно-последовательный режим: постоянное измерение последовательности каналов. Запись начинается с регистра ADC12MEMx, определяемого битами CSTARTADDx. После каждой записи происходит инкрементномера регистра. Последовательность прекращается, когда происходит измерение канала с установленным битом ADC12EOS. Новый цикл начинается после установки сигнала запуска измерения. Независимо от режима, если источниками запуска выбраны таймеры, то нужно переключать ADC12ENC для следующего измерения.

Таблица 3.4 – Поля иных регистров АЦП ADC12

Регистр Биты

ADC12MEMx 0-15

ADC12MCTLx 7

Поле

ADC12EOS

Назначение

Результат измерения

Маркер конца последовательности

Определение флагов в msp430f5529.h

ADC12EOS

4-6 ADC12SREFx Выбор пары опорных напряжений VR+, VR-

0-3 ADC12INCHx Выбор входного канала

ADC12SREF\_0 ... ADC12SREF\_7

ADC12INCH\_0 ... ADC12INCH\_15

ADC12IE Разрешение прерывания ADC12IE0 ... по соответствующему ADC12IE15 флагу

ADC12IFG

ADC12IV

Флаг запроса на прерывание

Вектор запросов на прерывания

ADC12IFG0 ... ADC12IFG15

ADC12IV\_NONE, ADC12IV\_ADC12OVIFG, ADC12IV\_ADC12TOVIFG, ADC12IV\_ADC12IFG0 ... ADC12IV\_ADC12IFG15

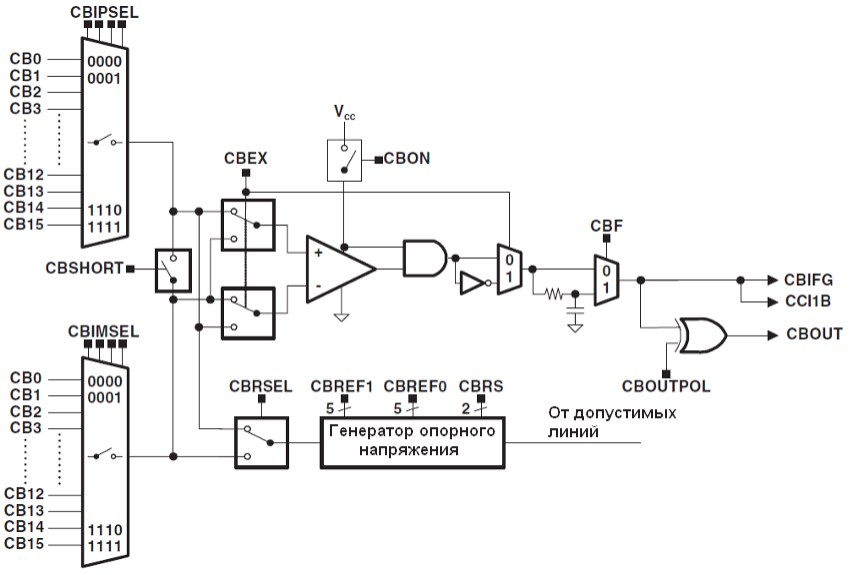
Для получения максимальной частоты измерений используется бит ADC12MSC = 1 (Multiple Sample and Convert) при режиме CONSEQx > 0 и использовании таймера выборок. В этом случае первый фронт SHI запускает первое измерение, после измерения автоматически запускается следующее. «Лишние» фронты SHI игнорируются, пока не окончится цикл измерения.

Использование встроенного температурного датчика: помимо выбора номера канала (1010), устанавливается либо ADC12REFON = 1 (при REFMSTR = 0), либо REFON = 1 (при REFMSTR = 1), при этом необходимо использовать период выборки больше 30 мкс.

Доступно 18 источников прерывания. ADC12IFG0 … ADC12IFG15 (эти флаги устанавливаются, когда происходит запись в соответствующий регистр ADC12MEMx), ADC12OV (переполнение ADC12MEMx – запись нового

7

результата до прочтения старого), ADC12TOV (превышение времени преобразования – вызывается новый цикл измерения до окончания предыдущего). Запрос на прерывание возникает, если установлен флаг разрешения соответствующего ADC12IEx и установлен глобальный флаг разрешения GIE. Любой доступ к ADC12IV автоматически сбрасывает флаги ADC12OV, ADC12TOV (если запрос от них был с максимальным приоритетом). ADC12IFGx сбрасывается программно либо автоматически при доступе к соответствующему регистру ADC12MEMx.



**3.2 Компаратор**

Компаратор в составе MSP430F5529 обладает следующими возможностями: прямое и инверсное сравнение; программное подключение RC-фильтра на выходе; выход подключается ко входу таймера А; программный выбор каналов (из 16 возможных); использование прерываний; программируемый генератор опорного напряжения.

Структура компаратор представлена на рисунке 3.4, а принцип работы фильтрации на выходе — на рисунке 3.5.

Рисунок 3.4 – Структура компаратора

8

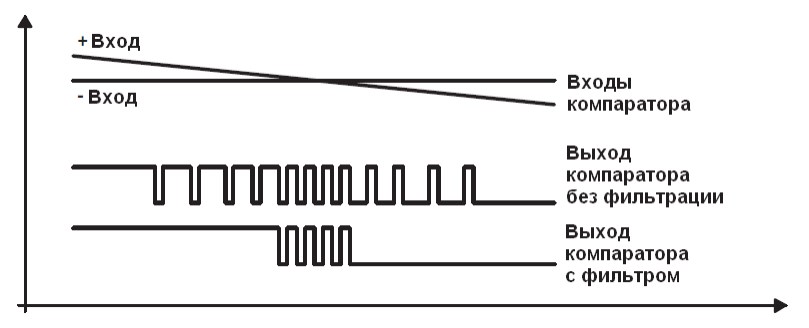


Рисунок 3.5 – Использование фильтра на выходе компаратора

Регистры управления компаратором CBCTL0 … CBCTL3 имеют адреса 08C0h, 0802h, 0804h, 0806h, регистр управления прерываниями CBINT – 080Ch, регистр флагов прерываний CBIV – 080Eh. Состав и назначение полей регистров компаратора приведены в таблице 3.5.

Таблица 3.5 – Поля регистров компаратора

Регистр

CBCTL0

CBCTL1

Биты Поле

15 CBIMEN 8-11 CBIMSEL

7 CBIPEN 0-3 CBIPSEL

12 CBMRVS

11 CBMRVL 10 CBON

8-9 CBPWRMD

6-7 CBFDLY

5 CBEX

4 CBSHORT

3 CBIES

Назначение

Разрешение входного канала на V-Выбор входного канала V-

Разрешение входного канала на V+ Выбор входного канала V+

0 – выход компаратора управляет выбором между VREF0 и VREF1 1 – управляет CBMRVL

Выбор VREFx Включение компаратора Режим питания

Величина задержки фильтра (0.6, 1.0, 1.8, 3.4 мкс)

Инверсный режим Закорачивание входных каналов

Выбор прерывания по фронту или спаду

Определение флагов в msp430f5529.h

CBIMEN

CBIMSEL\_0 ... CBIMSEL\_15

CBIPEN

CBIPSEL\_0 ... CBIPSEL\_15

CBMRVS

CBMRVL CBON

CBPWRMD\_0 ... CBPWRMD\_3

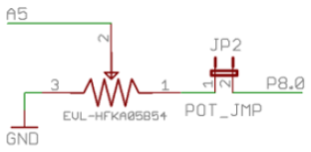
CBFDLY\_0 ... CBFDLY\_3

CBEX CBSHORT

CBIES

9

Окончание таблицы 3.5



Регистр

CBCTL1

CBCTL2

CBCTL3

CBINT

CBIV

Биты Поле

2 CBF

1 СBOUTPOL 0 CBOUT

15 CBREFACC 13-14 CBREFL

8-12 CBREF1

6-7 CBRS

5 CBRSEL

0-4 CBREF0

0-15 CBPD

9 CBIIE

8 CBIE

1 CBIIFG

0 CBIFG

Назначение

Выходной фильтр Полярность выхода Выход компаратора Точность

Опорное напряжение (отключено, 1.5, 2, 2.5 В)

Устанавливает напряжение на выходе резисторного делителя

Источник опорного напряжения

Коммутация опорного напряжения к + и - входам

Как и CBREF1

Отключение входного буфера для каждого вывода

Выход разрешения прерывания по инверсному выходу

Выход разрешения прерывания Флаг инверсного прерывания Флаг прерывания

Вектор прерываний

Определение флагов в msp430f5529.h

CBF CBOUTPOL CBOUT CBREFACC

CBREFL\_0 ... CBREFL\_3

CBREF1\_0 ... CBREF1\_31

CBRS\_0 ... CBRS\_3

CBREF0\_0 ... CBREF0\_31

CBPD0 ... CBPD15

CBIIE

CBIE CBIIFG CBIFG

CBIV\_NONE, CBIV\_CBIFG, CBIV\_CBIIFG

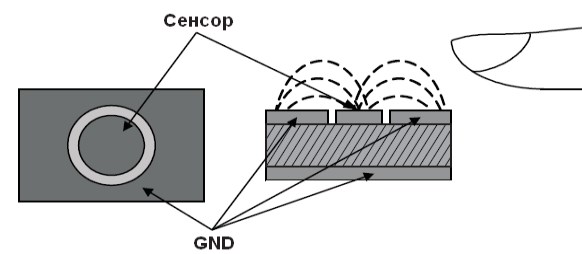
**3.3 Потенциометр**

Схема подключения потенциометра на макете MSP-EXP430F5529 приведена на рисунке 3.6.

Рисунок 3.6 – Схема подключения потенциометра

10

Cигнал с потенциометра подан на вывод A5 контроллера. Он соединен с соответствующими каналами (CB5, A5) на входах компаратора и АЦП. Поскольку данный вывод разделен с цифровым I/O (P6.5), его необходимо переключить в режим периферийных устройств на ввод данных. Порт P8.0 необходимо использовать в цифровом режиме на выход, подав на него высокий уровень, чтобы подать на резистор разность потенциалов.



**3.4 Сенсорная клавиатура**

На плате расположены пять площадок, совмещенных со светодиодными индикаторами. Каждая из таких площадок представляет собой емкостный сенсорный элемент. Сенсор сконструирован таким образом, что его электрическое поле и емкость могут быть изменены внешним токопроводящим объектом, например пальцем (см. рисунок 3.7).

Рисунок 3.7 – Принцип действия сенсорного элемента

При приближении к сенсору меняется магнитное поле и, следовательно, емкость. Количественная характеристика нажатия получается путем измерения емкости тач-сенсора. Очевидно, что магнитное поле сильно зависит от условий внешней среды, поэтому требуется отслеживание фонового уровня.

Один из методов измерения — RC-метод. Он основан на измерении времени разряда RC-цепи и пояснен на рисунке 3.8. Резистор включается между землей и сенсором, сенсор подключается к цифровому I/O. На I/O подается «1» и конденсатор быстро заряжается. Читается время таймера. После этого I/O устанавливается на вход с прерыванием по спаду, контроллер переводится в режим LPM0. Конденсатор разряжается через резистор, когда напряжение станет меньше порога, произойдет прерывание. Обработчик прерывания вновь читает таймер, вычисляет время разряда, после чего микроконтроллер выводится из LPM0.

11

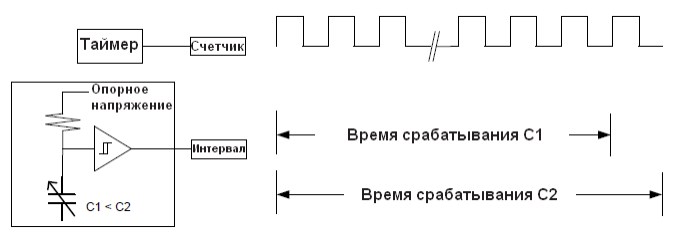
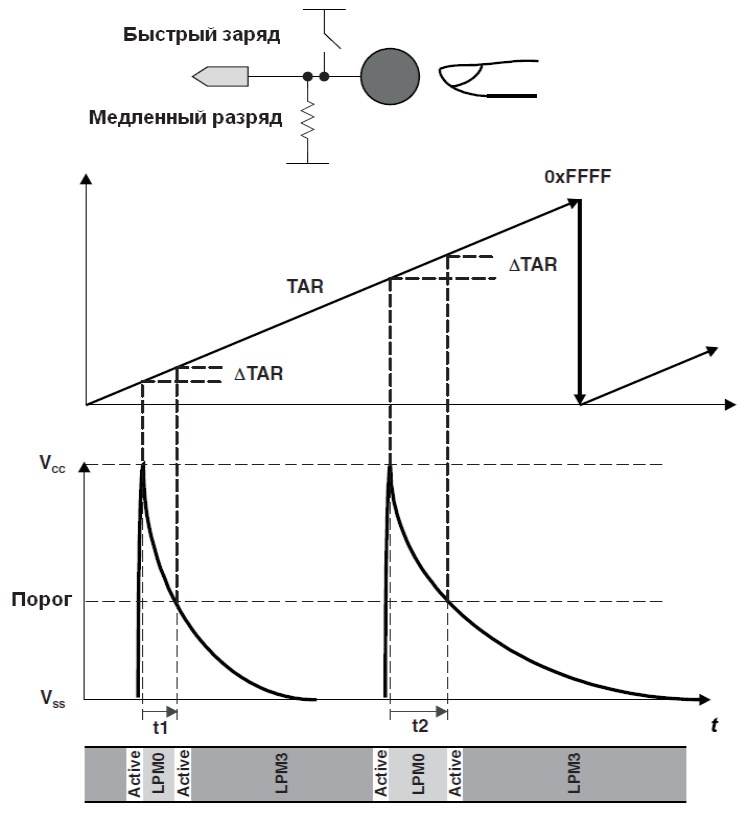
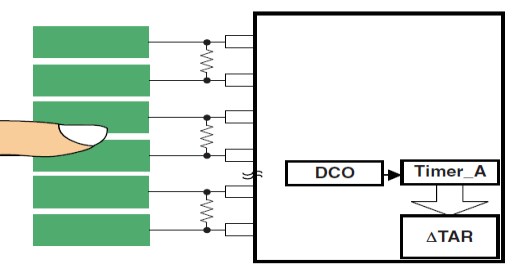
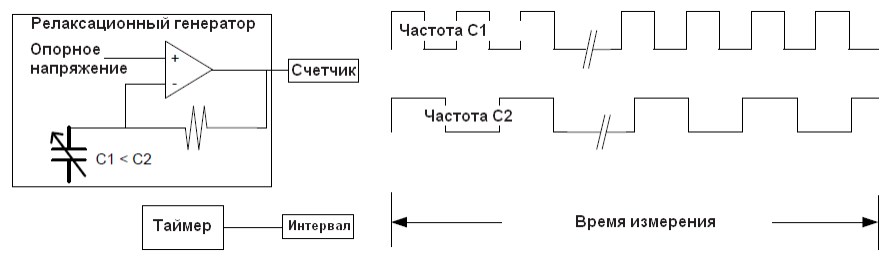


Рисунок 3.8 – Измерение емкости сенсора. RC-метод

12

Для снижения чувствительности сенсора к шуму используют разные подходы. Например, подавление шума с помощью БИХ-фильтра, либо усреднение времени заряда и времени разряда.



Иногда сенсоры объединяют попарно, что позволяет вдвое снизить количество используемых резисторов. В этом случае, когда один сенсор измеряется, второй заряжается и наоборот (см. рисунок 3.9).

Рисунок 3.9 – Попарное объединение сенсоров

Другими методами, используемыми для измерений, являются RO-метод (Relaxation Oscillator) и fRO (fast scan Relaxation Oscillator). В случае RO-метода считается количество тактов релаксационного генератора за некоторое время. Генератор включает в себя сенсор и компаратор (см. рисунок 3.10).

Рисунок 3.10 – Измерение емкости сенсора. RО-метод

В случае fRO-метода измеряется длительность периода одного такта релаксационного генератора. Он подобен RC-методу, но использует релаксационный генератор, а не RC-цепочку (см. рисунок 3.11).

Схема включения сенсорныхэлементов и связанныхс нимисветодиодов представлена на рисунке (см. рисунок 3.12).

13

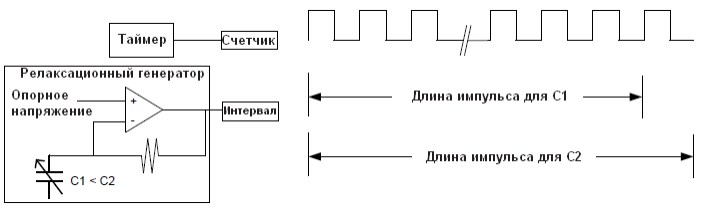
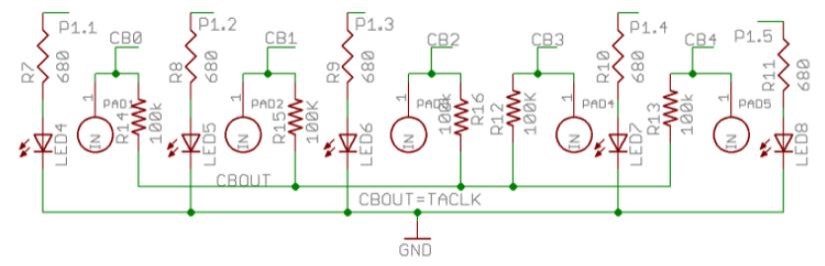


Рисунок 3.11 – Измерение емкости сенсора. fRО-метод

Рисунок 3.12 – Схема включения сенсорных элементов

Сигналы с сенсорных элементов поданы на выводы CB0 – CB4 контроллера и разделяются с цифровыми I/O P6.0 – P6.4 соответственно, а также подключены к каналам компаратора и АЦП (CB0 – CB4 и A0 – A4 соответственно). Резистор каждого из элементов подключен к выводу CBOUT контроллера, который также разделяется c цифровым I/O (P1.6), и связан с входом TA1CLK таймера TA1.

**4 Выполнение работы**

**4.1 Описание программы**

Используется два канала АЦП в повторно-многоканальном режиме. По прерыванию каждого канала данные заносятся в переменные и происходит выход из LPM0. В main происходит сравнение переменных со значениями и вход в режим LPM0.

**4.2 Схемы соединения и диаграммы работы**

14

*P8.0*

*TEMP*

*P6.5*

*SENSOR VOUT* *Vcc*

*CH5* *ADC12* *MEM0 CH10 MEM1 Vref+*

*Vref-*

*temp\_adc*

*pot\_adc*

Рисунок 4.1 – Схема соединения

Температурный сенсор подключается к каналу 10 АЦП, выход потенциометра – к каналу 5. В MEM0 заносится значения температурного сенсора, в MEM1 заносится значения потенциометра (см. рисунок 4.1).

*temp\_adc*



*pot\_adc* t t1 t2 t3 t4

Рисунок 4.2 – Диаграмма работы

Пример работы приведен на рисунке 4.2. В момент времени t1 загорается LED1, в t2 – LED3, в t3 – LED1, в t4 – LED3.

**4.3 Листинг кода**

#include <msp430.h> #include <stdint.h>

/\* Функция для взаимодействия с GPIO \*/

#define GPIO\_DIR\_OUTPUT(...) GPIO\_DIR\_OUTPUT\_SUB(\_\_VA\_ARGS\_\_) #define GPIO\_DIR\_OUTPUT\_SUB(port, pin) (P##port##DIR |= (1 << (pin)))

#define GPIO\_DIR\_INPUT(...) GPIO\_DIR\_INPUT\_SUB(\_\_VA\_ARGS\_\_)

#define GPIO\_DIR\_INPUT\_SUB(port, pin) (P##port##DIR &= ~(1 << (pin)))

15

#define GPIO\_PULLUP(...) GPIO\_PULLUP\_SUB(\_\_VA\_ARGS\_\_)

#define GPIO\_PULLUP\_SUB(port, pin) P##port##REN |= (1 << (pin)); \ P##port##OUT |= (1 << (pin))

#define GPIO\_PULLDOWN(...) GPIO\_PULLDOWN\_SUB(\_\_VA\_ARGS\_\_)

#define GPIO\_PULLDOWN\_SUB(port, pin) P##port##REN |= (1 << (pin)); \ P##port##OUT &= ~(1 << (pin))

#define GPIO\_NOPULL(...) GPIO\_NOPULL\_SUB(\_\_VA\_ARGS\_\_)

#define GPIO\_NOPULL\_SUB(port, pin) (P##port##REN &= ~(1 << (pin)))

#define GPIO\_READ\_PIN(...) GPIO\_READ\_PIN\_SUB(\_\_VA\_ARGS\_\_)

#define GPIO\_READ\_PIN\_SUB(port, pin) ((P##port##IN & (1 << (pin))) ? 1 : 0)

#define GPIO\_WRITE\_PIN(...) GPIO\_WRITE\_PIN\_SUB(\_\_VA\_ARGS\_\_)

#define GPIO\_WRITE\_PIN\_SUB(port, pin, value) (P##port##OUT = (P##port##OUT & ~(1 << (pin))) | (value << (pin)))

#define GPIO\_TOGGLE\_PIN(...) GPIO\_TOGGLE\_PIN\_SUB(\_\_VA\_ARGS\_\_)

#define GPIO\_TOGGLE\_PIN\_SUB(port, pin) (P##port##OUT ^= (1 << (pin)))

#define GPIO\_TRIG\_EDGE\_FALLING(...) GPIO\_TRIG\_EDGE\_FALLING\_SUB(\_\_VA\_ARGS\_\_) #define GPIO\_TRIG\_EDGE\_FALLING\_SUB(port, pin) (P##port##IES |= (1 << (pin)))

#define GPIO\_TRIG\_EDGE\_RISING(...) GPIO\_TRIG\_EDGE\_RISING\_SUB(\_\_VA\_ARGS\_\_) #define GPIO\_TRIG\_EDGE\_RISING\_SUB(port, pin) (P##port##IES &= ~(1 << (pin)))

#define GPIO\_INTERRUPT\_ENABLE(...) GPIO\_INTERRUPT\_ENABLE\_SUB(\_\_VA\_ARGS\_\_) #define GPIO\_INTERRUPT\_ENABLE\_SUB(port, pin) P##port##IFG &= ~(1 << (pin)); \

P##port##IE |= (1 << (pin))

#define GPIO\_INTERRUPT\_DISABLE(...) GPIO\_INTERRUPT\_DISABLE\_SUB(\_\_VA\_ARGS\_\_) #define GPIO\_INTERRUPT\_DISABLE\_SUB(port, pin) (P##port##IE &= ~(1 << (pin)))

#define GPIO\_PERIPHERAL(...) GPIO\_PERIPHERAL\_SUB(\_\_VA\_ARGS\_\_)

#define GPIO\_PERIPHERAL\_SUB(port, pin) (P##port##SEL |= (1 << (pin)))

#define GPIO\_CLEAR\_IT\_FLAG(...) GPIO\_CLEAR\_IT\_FLAG\_SUB(\_\_VA\_ARGS\_\_) #define GPIO\_CLEAR\_IT\_FLAG\_SUB(port, pin) (P##port##IFG &= ~(1 << (pin)))

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define LED\_ON 1 #define LED\_OFF 0

/\* Порты и пины используемых выводов \*/

#define LED1\_PORT 1 #define LED1\_PIN 0

#define LED3\_PORT 8 #define LED3\_PIN 2

#define POT\_VCC\_PORT 8 #define POT\_VCC\_PIN 0

#define POT\_OUT\_PORT 6 #define POT\_OUT\_PIN 5

// Значение потенциометра volatile uint16\_t temp\_adc = 0;

// Значение температурного датчика volatile uint16\_t pot\_adc = 0;

int main(void)

16

{

// Остановка сторожевого таймера WDTCTL = WDTPW + WDTHOLD;

// Подаем напряжение на ногу потенциометра GPIO\_DIR\_OUTPUT(POT\_VCC\_PORT, POT\_VCC\_PIN); GPIO\_WRITE\_PIN(POT\_VCC\_PORT, POT\_VCC\_PIN, 1);

GPIO\_DIR\_OUTPUT(LED3\_PORT, LED3\_PIN); GPIO\_WRITE\_PIN(LED3\_PORT, LED3\_PIN, LED\_OFF);

GPIO\_DIR\_OUTPUT(LED1\_PORT, LED1\_PIN); GPIO\_WRITE\_PIN(LED1\_PORT, LED1\_PIN, LED\_OFF);

// Включаем функцию АЦП на ножке msp430 GPIO\_PERIPHERAL(POT\_OUT\_PORT, POT\_OUT\_PIN);

// Дает возможность управлять опорным напряжениям с помощью регистров АЦП REFCTL0 &= ~REFMSTR;

// время выборки АЦП = 256 ADC12CLK циклов // включение опорного генератора

// включение АЦП, непрерывный режим измерения

ADC12CTL0 = ADC12SHT0\_8 + ADC12REFON + ADC12ON + ADC12MSC;

// повторно-многоканальный, импульсный режим, ADC12CTL1 = ADC12SHP + ADC12CONSEQ\_3;

// 10 Канал - температурный сенсор, опорные VCC и GND ADC12MCTL0 = ADC12SREF\_0 + ADC12INCH\_10;

// 5 Канал - потенциометр, последний канал, опорные VCC и GND ADC12MCTL1 = ADC12SREF\_0 + ADC12INCH\_5 + ADC12EOS;

// Разрешение прерываний для ADCMEM0, ADCMEM1 ADC12IE = 0x003;

// Разрешение измерения ADC12CTL0 |= ADC12ENC; // Начало измерения ADC12CTL0 |= ADC12SC; while(1)

{

// Вход в режим LPMO с разрешением прерываний \_\_bis\_SR\_register(LPM0\_bits + GIE);

if (temp\_adc > pot\_adc) {

GPIO\_WRITE\_PIN(LED3\_PORT, LED3\_PIN, LED\_OFF); GPIO\_WRITE\_PIN(LED1\_PORT, LED1\_PIN, LED\_ON);

} else {

GPIO\_WRITE\_PIN(LED3\_PORT, LED3\_PIN, LED\_ON); GPIO\_WRITE\_PIN(LED1\_PORT, LED1\_PIN, LED\_OFF);

} }

}

#pragma vector=ADC12\_VECTOR \_\_interrupt void ADC12ISR (void) {

switch(\_\_even\_in\_range(ADC12IV,34)) {

// Источник прерывания ADC12IFG0 case 6:

17

temp\_adc = ADC12MEM0; break;

// Источник прерывания ADC12IFG1 case 8:

pot\_adc = ADC12MEM1; break;

default: break; }

// Выход из LPM0 \_\_bic\_SR\_register\_on\_exit(LPM0\_bits);

}

**5 Выводы**

В ходе лабораторной работы были изучены принципы работы с аналоговыми сигналами на базе микроконтроллера MSP430F5529. В результате выполнения работы была написана программа, сравнивающая напряжения на выходах температурного датчика и потенциометра с помощью АЦП.

18