

Министерство образования Республики Беларусь

Учреждение образования
«Белорусский государственный университет информатики и
радиоэлектроники»

Кафедра ЭВМ

Лабораторная работа № 4
«Аналоговый ввод-вывод. АЦП. Компаратор. Потенциометр. Сенсорная
клавиатура»
Вариант №7

Выполнил:

Проверил:

МИНСК 2024

1 ЦЕЛЬ РАБОТЫ

Изучить принципы работы с аналоговыми сигналами на базе микроконтроллера MSP430F5529.

2 ИСХОДНЫЕ ДАННЫЕ

1. В соответствии с вариантом задания, показанном в таблице 2.1, написать программу, которая непрерывно сравнивает сигнал на указанных выводах и в зависимости от того, где уровень выше, включает тот или иной светодиод. Для тач-панели использовать соответствующие кнопкам светодиоды, для потенциометра — LED3, для иных сигналов — LED1. Не допускается использовать иные заголовочные файлы, кроме `msp430`, а также использовать высокоуровневые библиотеки.

2. В отчет по выполнению работы включить исходный текст программы с обязательными комментариями. Комментарии в тексте программы обязательны, они должны пояснять что именно делает данный фрагмент. Описать подробнее используемый метод сравнения сигналов, изобразить соответствующую схему соединения и диаграммы работы. Привести объяснение полученным результатам. Задание по варианту:

7	термодатчик АЦП	опорное напряжение АЦП V_{ref+}	АЦП
---	-----------------	-----------------------------------	-----

3 ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Аналого-цифровой преобразователь

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, V_{сс} и внешних опорных напряжений;
- независимый для каждого канала опорный источник, как положительного, так и отрицательного напряжения;
- конфигурируемый источник тактового сигнала;
- четыре режима преобразования: одноканальный, повторноодноканальный, 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+ источники опорного напряжения.

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

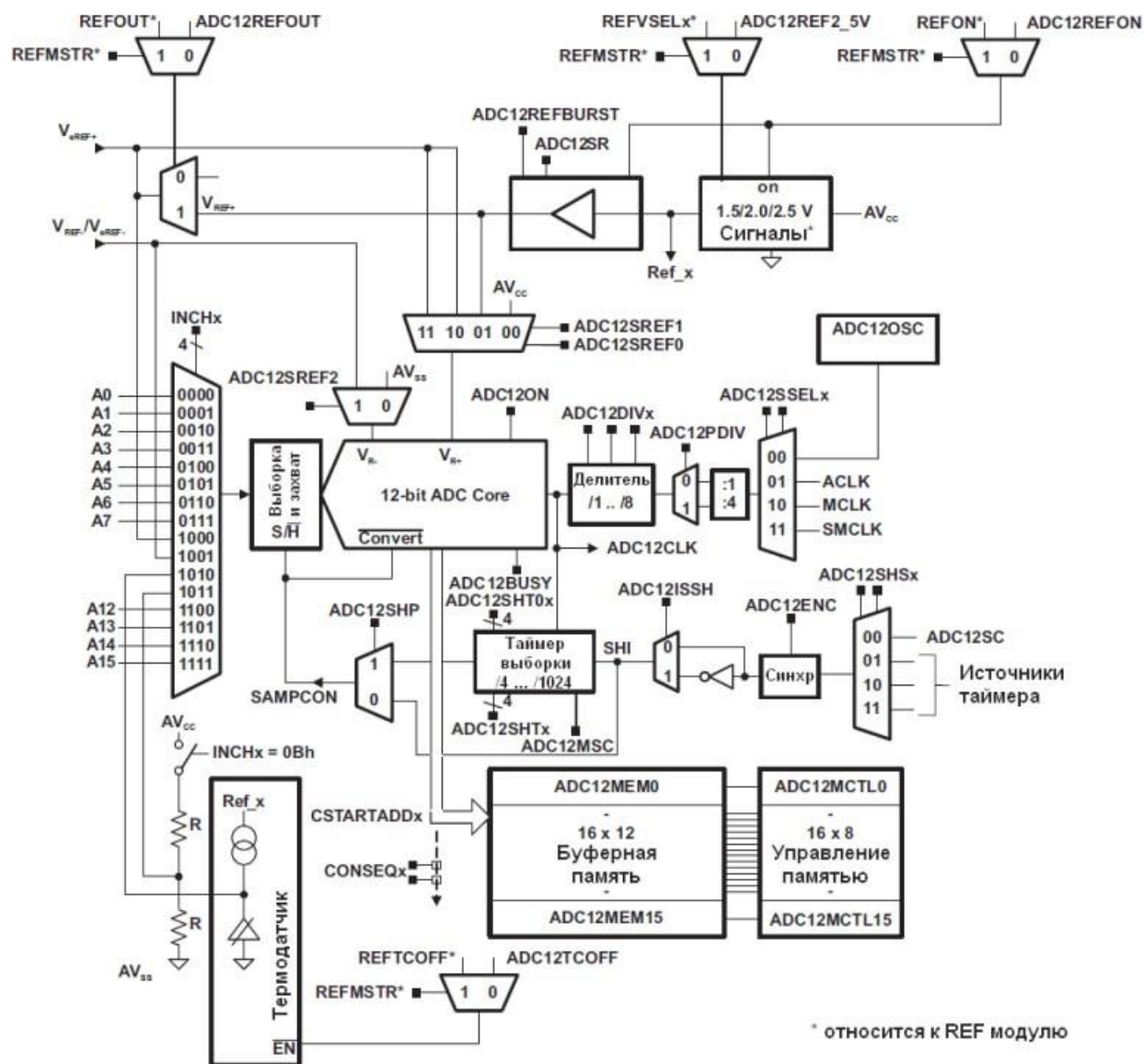


Рисунок 3.1 - Структура АЦП ADC12A

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 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 $ADC12SHP = 0$, $SAMPCON = SHI$.

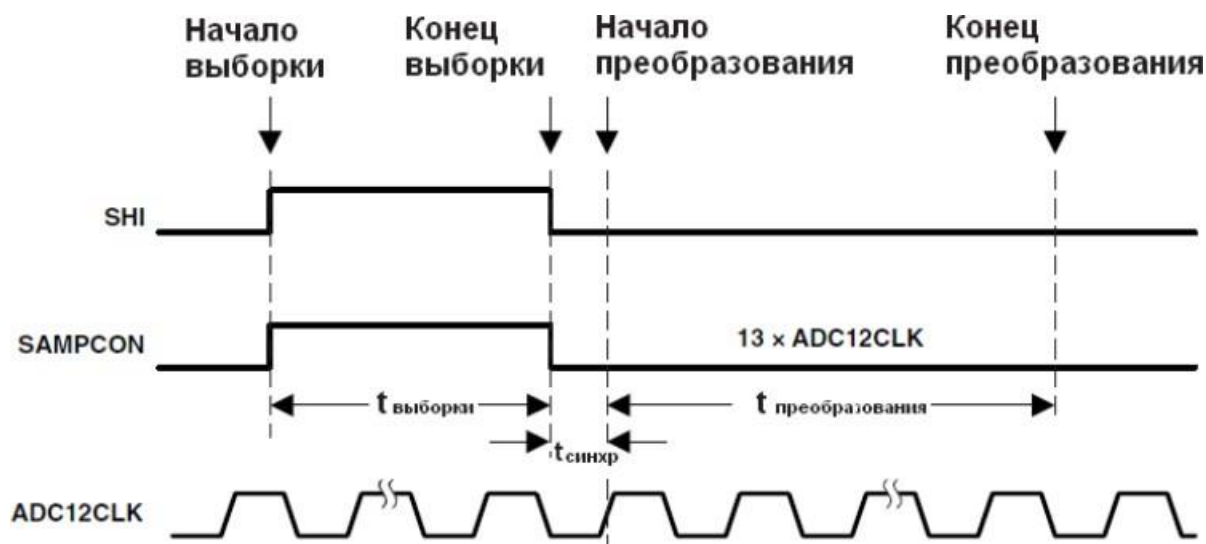


Рисунок 3.2 - Расширенный режим

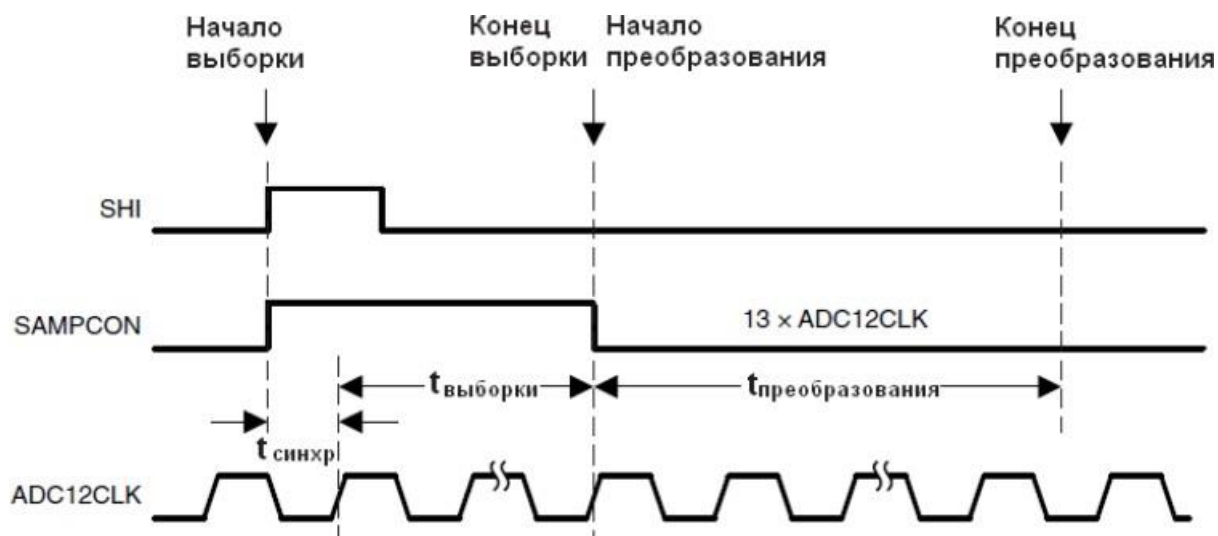


Рисунок 3.3 - Импульсный режим

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

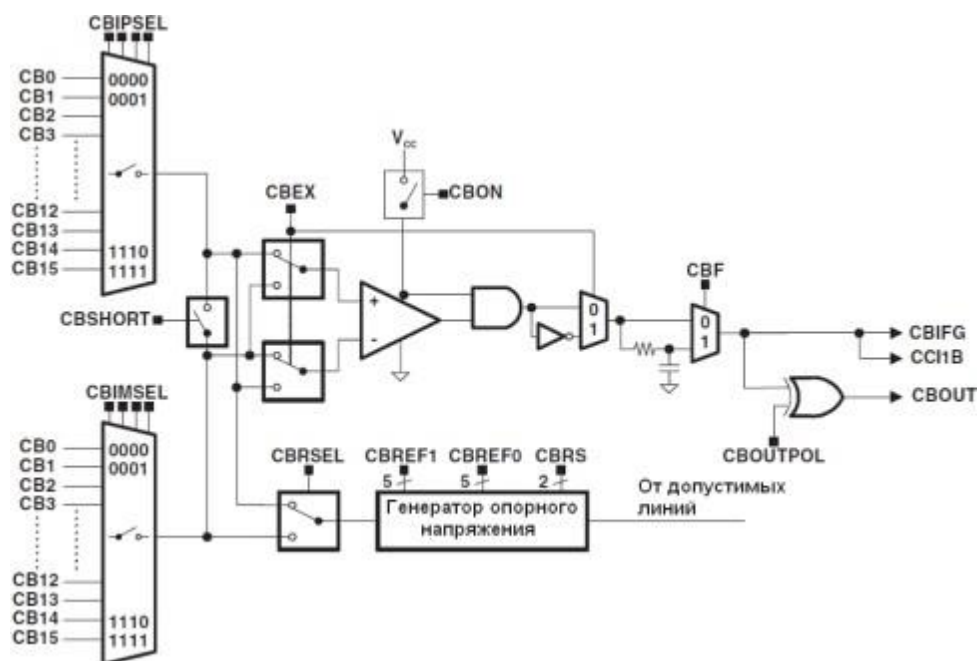


Рисунок 3.4 - Структура компаратора

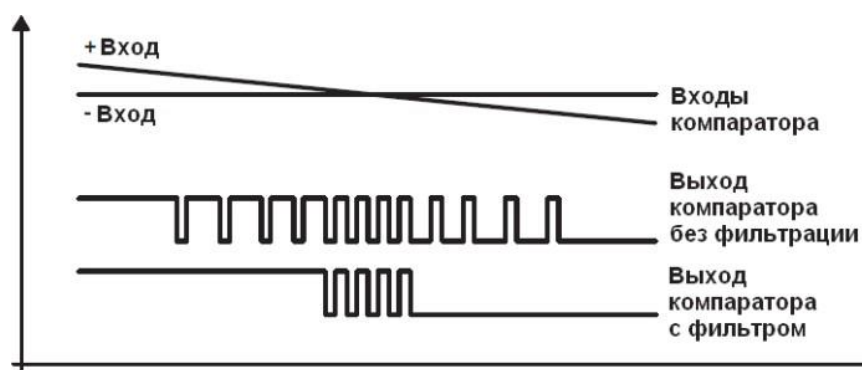


Рисунок 3.5 - Использование фильтра на выходе компаратора

Потенциометр

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

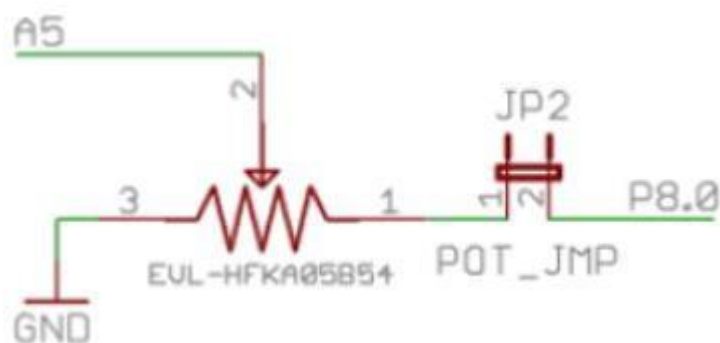


Рисунок 3.6 - Схема подключения потенциометра

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. Это показано на рисунке 3.7.

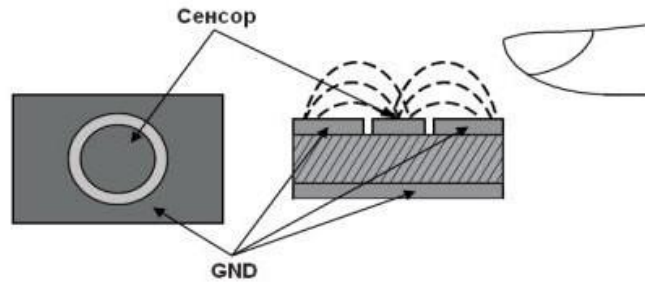


Рисунок 3.7 - Принцип действия сенсорного элемента

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.

4 ВЫПОЛНЕНИЕ РАБОТЫ

Ниже представлен текст программы:

```
#include <msp430.h>

void configureADC();           // Прототип функции настройки
АЦП
unsigned int readADC(unsigned int channel); // Прототип
функции для чтения с канала АЦП

#define LED1 BIT0             // Светодиод 1 подключен к P1.0
#define LED3 BIT2             // Светодиод 3 подключен к P8.2

void main(void)
{
    WDTCTL = WDTPW | WDTHOLD; // Останавливаем watchdog
таймер

    // Настройка портов для светодиодов
    P1DIR |= LED1;             // Устанавливаем LED1 как
ВЫХОД
    P1OUT &= ~LED1;            // Изначально LED1 выключен

    P8DIR |= LED3;             // Устанавливаем LED3 как
ВЫХОД
    P8OUT &= ~LED3;            // Изначально LED3 выключен

    configureADC();            // Настраиваем АЦП

    while (1)
    {
        // Чтение термодатчика (канал 10) и опорного
        напряжения (канал 11)
        unsigned int tempValue = readADC(10); // Чтение
        термодатчика
        // tempValue += 50; // Добавляем 50 для тестирования
        (например)

        unsigned int vrefValue = readADC(11); // Читаем
        опорное напряжение

        // Сравниваем значения термодатчика и опорного
        напряжения
        if (tempValue > vrefValue) {
            P1OUT |= LED1;      // Включаем LED1, если
            температура выше опорного напряжения
            P8OUT &= ~LED3;      // Выключаем LED3
        } else {
            P1OUT &= ~LED1;      // Выключаем LED1
```

```

        P8OUT |= LED3;          // Включаем LED3, если опорное
напряжение выше или равно
    }
}

// Настройка АЦП
void configureADC()
{
    ADC12CTL0 = ADC12SHT0_2 | ADC12ON;          //
Устанавливаем время выборки и включаем АЦП
    ADC12CTL1 = ADC12SHP;                      //
Используем источник от таймера
    ADC12CTL2 = ADC12RES_2;                    //
Устанавливаем разрешение 12 бит
    ADC12MCTL0 = ADC12INCH_10 | ADC12SREF_1;    // Канал
10 (термодатчик), опорное напряжение
    ADC12MCTL1 = ADC12INCH_11 | ADC12SREF_1;    // Канал
11 (опорное напряжение Vref+)
    ADC12CTL0 |= ADC12ENC;                      // Разрешаем
измерение
}

// Чтение с канала АЦП
unsigned int readADC(unsigned int channel)
{
    ADC12CTL0 &= ~ADC12SC;                      // Сброс
флага старта
    ADC12CTL0 |= ADC12SC;                      // Старт
нового преобразования
    while (ADC12CTL1 & ADC12BUSY);              // Ожидание
завершения
    return ADC12MEM0;                          //
Возвращаем результат
}

```

5 ВЫВОДЫ

В ходе выполнения лабораторной работы была в соответствии с вариантом задания написана программа, которая непрерывно сравнивает сигнал на указанных выводах и в зависимости от того, где уровень выше, включает тот или иной светодиод