

- ***ADC 12А. Аналого-цифровой преобразователь, 16 каналов***
- ***COMP\_B. Аналоговый компаратор, 16 каналов***

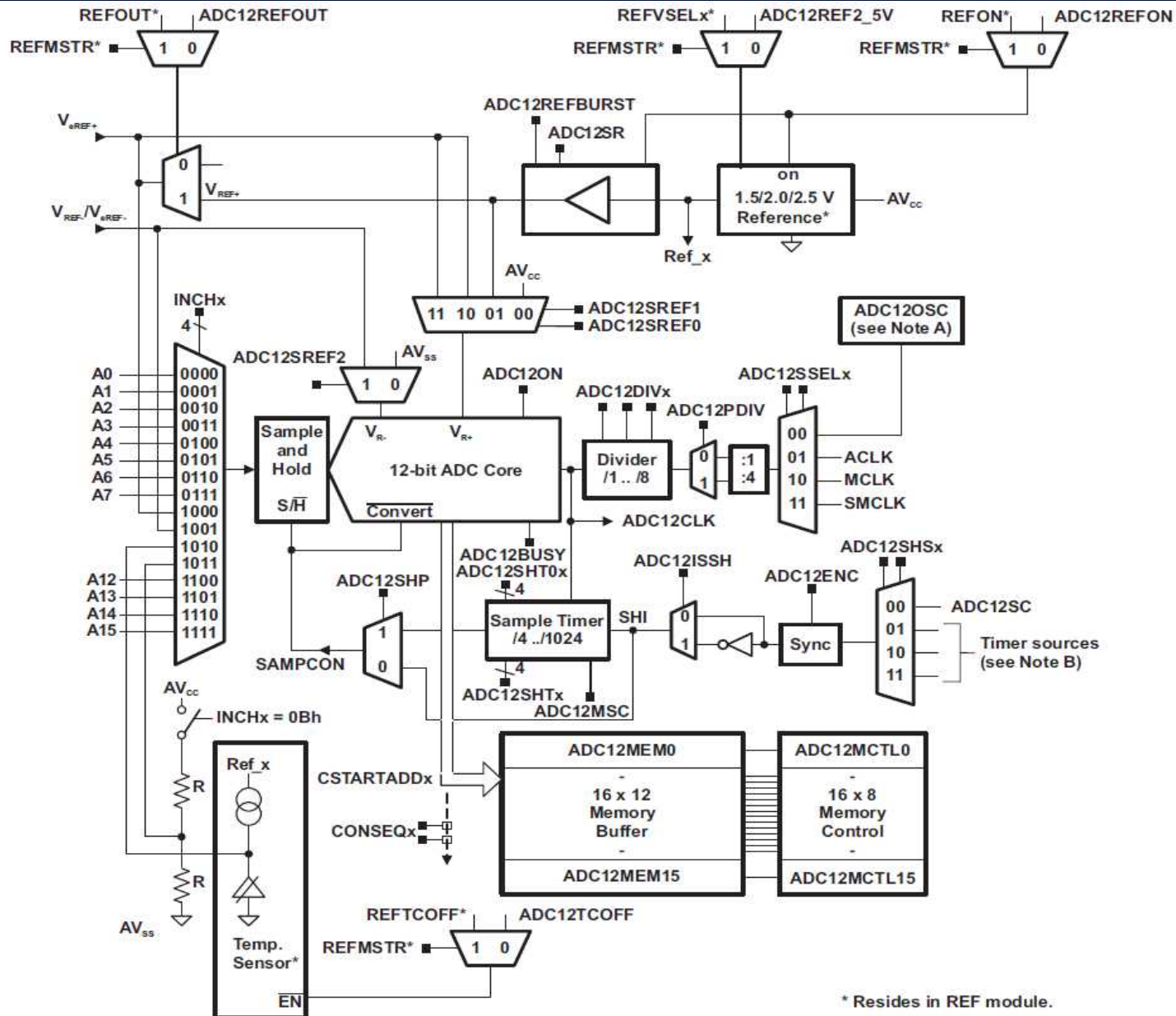
# **MSP430F5529. ADC\_12A**

- **максимальная скорость преобразования — более 200 тыс. 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 V)**
- **программный выбор внешнего или внутреннего источника опорного напряжения**
- **12 отдельно конфигурируемых внешних входных каналов**
- **каналы для внутреннего датчика температуры,  $V_{ss}$  и внешних опорных напряжений**

# **MSP430F5529. ADC\_12A**

- **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 прерываний АЦП**
- **Сохранение 16 результатов**
- **Ядро АЦП автоматически отключается, если не идет цикл измерения**
- **$N_{ADC} = 4095 * (V_{IN} - V_{R-}) / (V_{R+} - V_{R-})$**

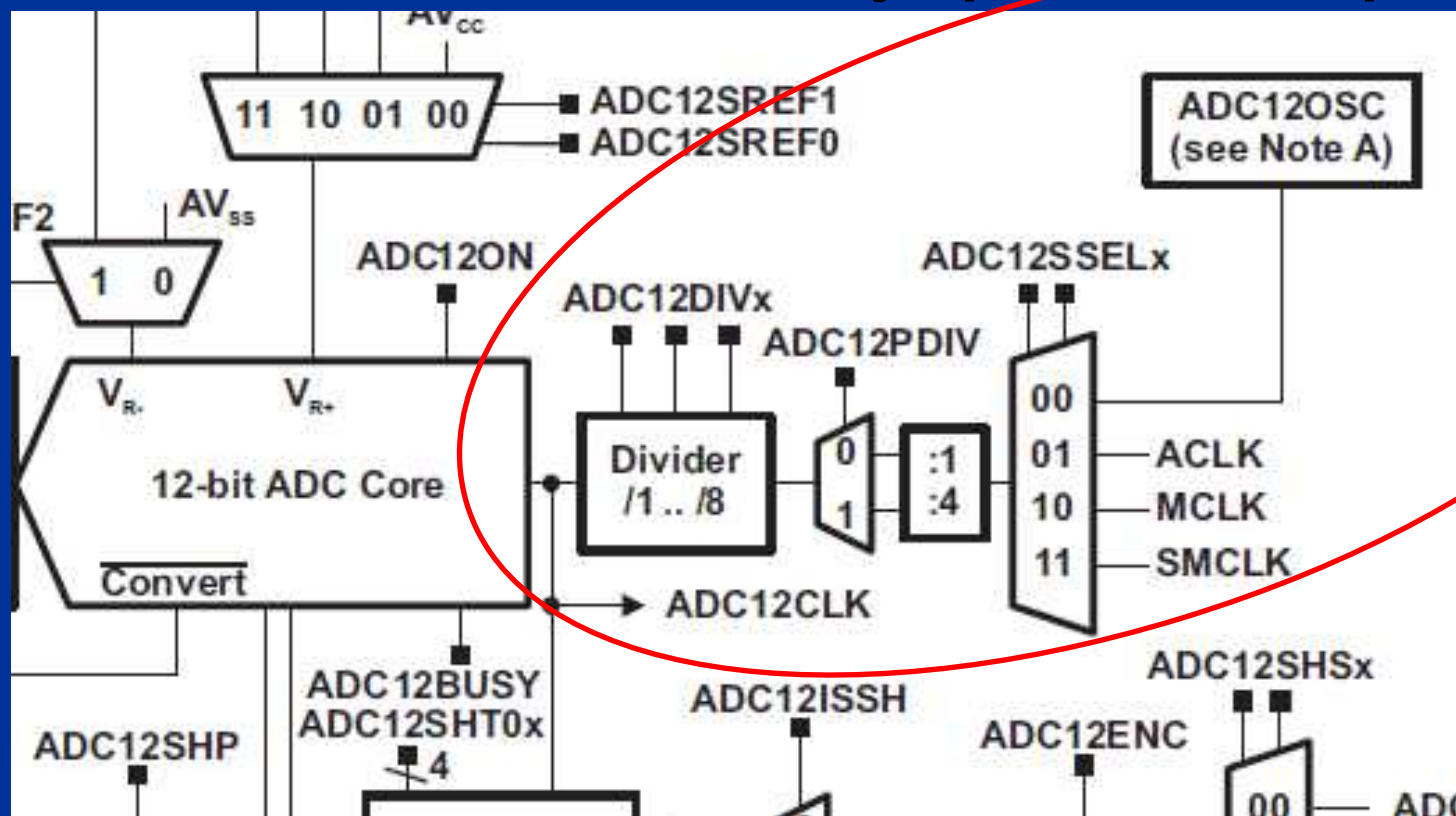
# MSP430F5529. ADC\_12A





# MSP430F5529. ADC\_12A

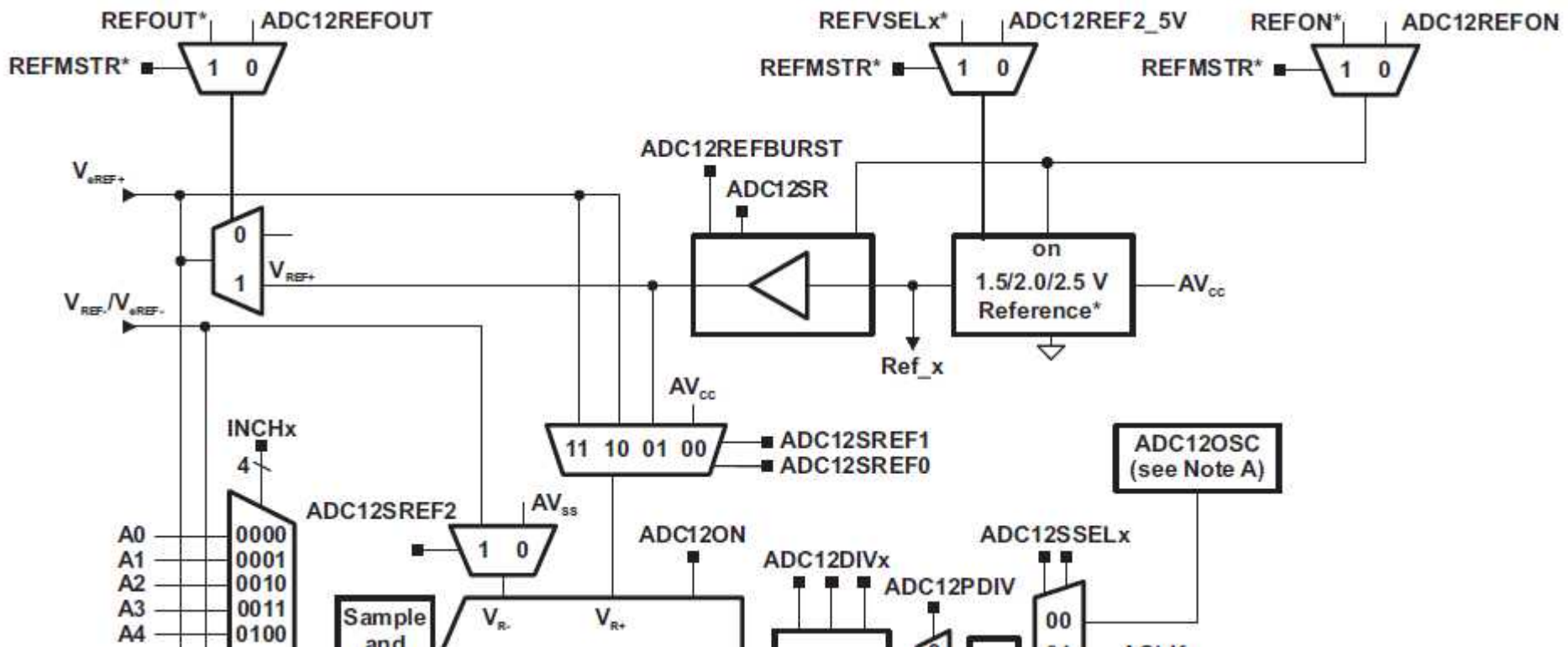
- Выбор тактирования из 4 источников: MCLK, SMCLK, ACLK, ADC12OSC
- Частота может делиться /1, /2, ... /8, /12, ... /32 раза при использовании двух делителей: ADC12DIVx, ADC12PDIV
- ADC12OSC связан со внутренним генератором 5 МГц





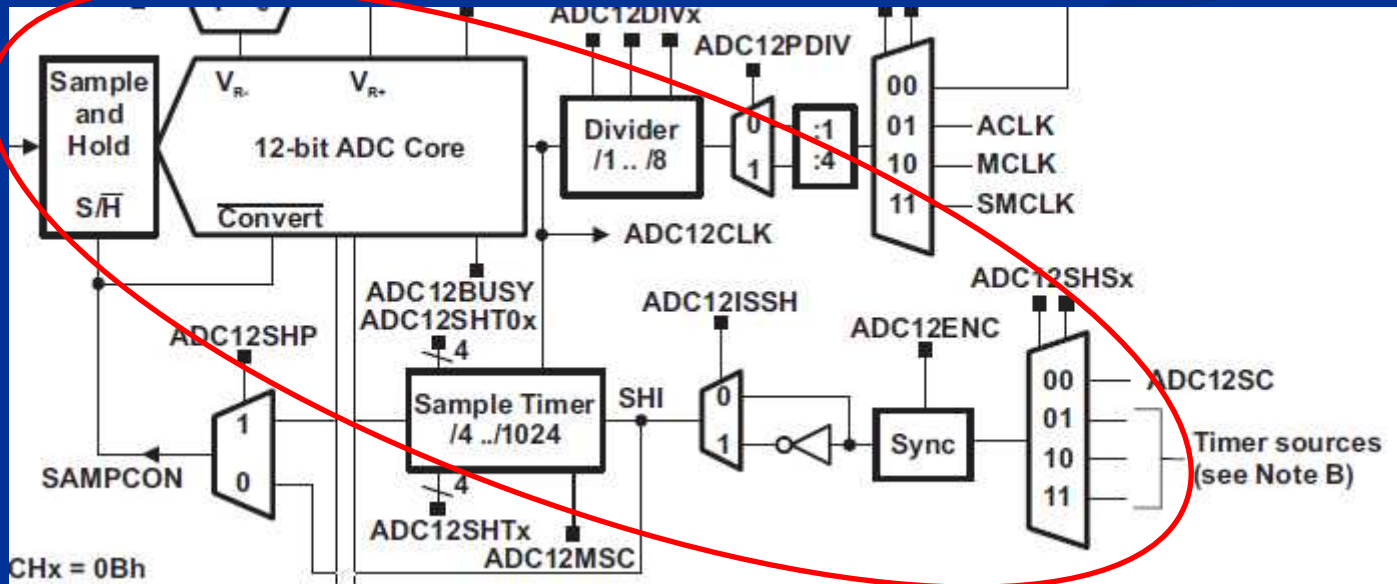
# MSP430F5529. ADC\_12A

- **Используется независимый генератор опорного напряжения (1.5, 2.0, 2.5 V), который может быть доступен как внутри АЦП, так и снаружи (выход  $V_{REF+}$ )**
- **REF ON включает/отключает генератор**
- **REFBURST = 1 включает буферный усилитель автоматически только когда используется АЦП; = 0 - постоянно**



# MSP430F5529. ADC\_12A

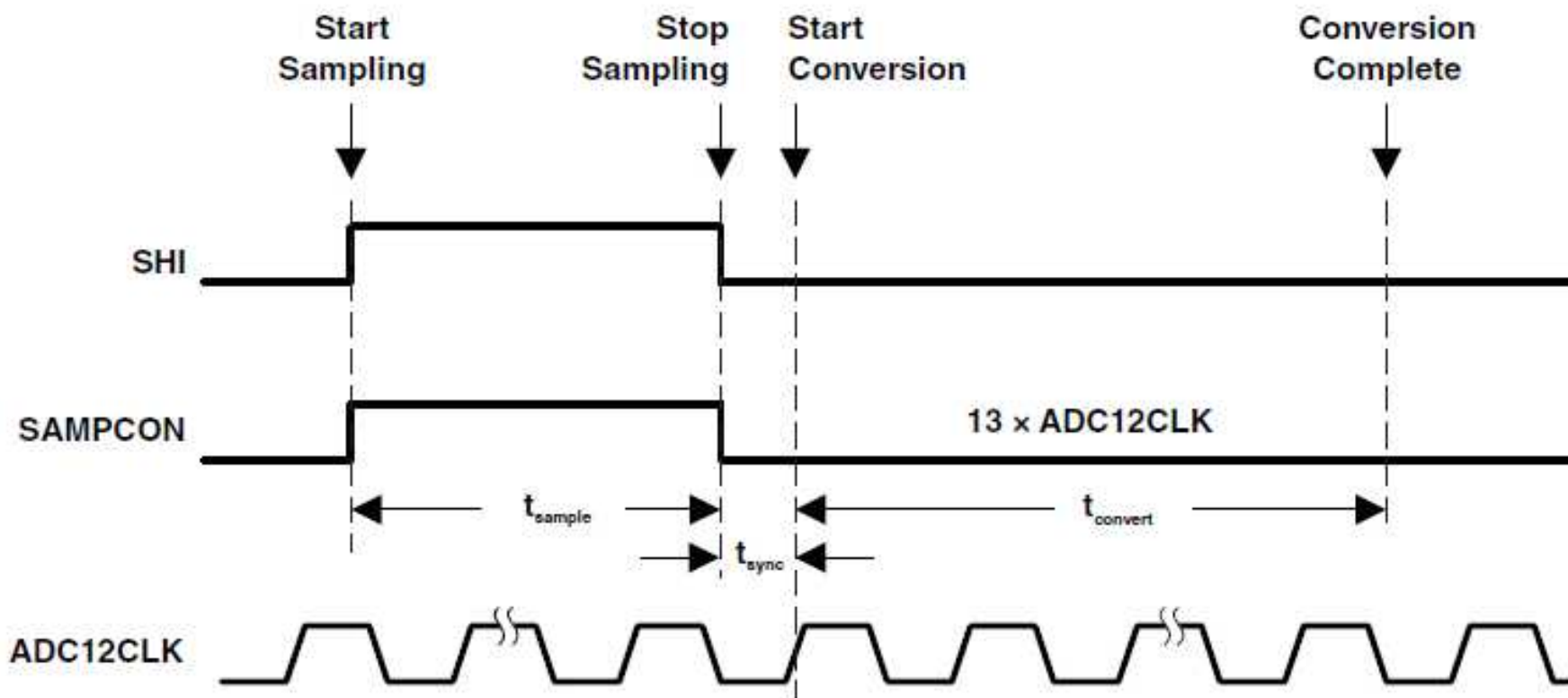
- Преобразование начинается по фронту входного сигнала *SHI* (выбирается из *ADC12SC* и сигналов от таймера)
- *ADC12RES* выбирает точность (8, 10 или 12 бит), что требует 9, 11 или 13 тактов генератора
- *ADC12DF* определяет знаковое или беззнаковое хранение результата
- *SAMPCON* управляет периодом выборки и началом преобразования. Преобразование начинается по спаду сигнала
- *ADC12SHP* – 2 режима преобразования (расширенный и импульсный)


$$CH_x = 0Bh$$



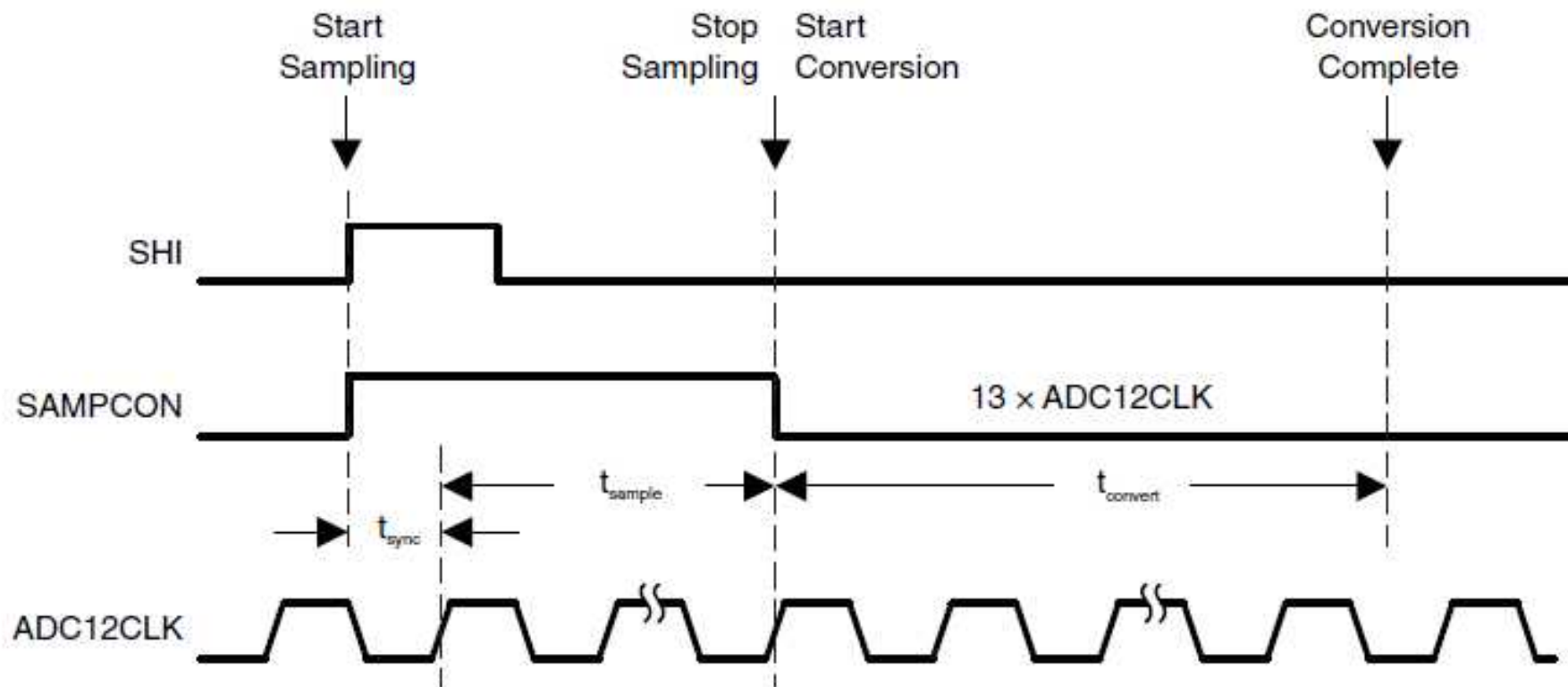
# MSP430F5529. ADC\_12A

- **Расширенный режим преобразования:**  
 **$ADC12SHP = 0$ ,**  
 **$SAMPCON == SHI$**



# MSP430F5529. ADC\_12A

- **импульсный режим преобразования:**  
 **$ADC12SHP = 1$ ,  $SHI$  сигнал определяет начало выборки, биты  $ADC12SHT0x$  и  $ADC12SHT1x$  регистра  $ADC12CTL0$  задают длительность времени выборки**



## *MSP430F5529. ADC\_12A*

- *CONSEQx* выбирает режим
- *Одноканальный режим: однократное измерение одного канала. Результат записывается в ADC12MEMx, определяемый битами CSTARTADDx*
- *Повторно-одноканальный: непрерывные измерения одного канала. Результат записывается в ADC12MEMx, определяемый битами CSTARTADDx. Необходимо читать результат после каждого измерения*

# *MSP430F5529. ADC\_12A*

- *Последовательный режим:  
однократное измерение  
последовательности каналов*
- *Запись начинается с регистра  
ADC12MEMx, определяемого битами  
CSTARTADDx. После каждой записи  
происходит инкремент номера  
регистра*
- *Измерение прекращается, когда  
происходит измерение канала с  
установленным битом ADC12EOS*

## ***MSP430F5529. ADC\_12A***

- ***Повторно-последовательный режим: постоянное измерение последовательности каналов***
- ***Запись начинается с регистра ADC12MEMx, определяемого битами CSTARTADDx. После каждой записи происходит инкремент номера регистра***
- ***Последовательность прекращается, когда происходит измерение канала с установленным битом ADC12EOS***
- ***Новый цикл начинается после установки сигнала запуска измерения***



## ***MSP430F5529. ADC\_12A***

- ***Во всех режимах: если источниками запуска выбраны таймеры, нужно переключать ADC12ENC для следующего измерения***
- ***Максимальная частота измерения:***
- ***Используется бит ADC12MSC = 1 (Multiple Sample and Convert) при режиме CONSEQx > 0 и использовании таймера выборок***
- ***Тогда 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*
  - *ADC12REFON = 1 (при REFMSTR = 0)*
  - *REFON = 1 (при REFMSTR = 1)*
- *Необходимо использовать период выборки больше 30 мкс*

# **MSP430F5529. ADC\_12A**

- **18 источников прерывания:**
- **ADC12IFG0 ... ADC12IFG15**
- **ADC12OV (Переполнение ADC12MEMx – запись нового результата до прочтения старого)**
- **ADC12TOV (Превышение времени преобразования – вызывается новый цикл измерения до окончания предыдущего)**
- **Флаги прерывания ADC12IFGx**  
устанавливаются, когда происходит запись в соответствующий регистр ADC12MEMx
- **Запрос на прерывание возникает, если установлен флаг разрешения соответствующего ADC12IEx и установлен глобальный флаг разрешения GIE**

## ***MSP430F5529. ADC\_12A***

- ***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***

# ***MSP430F5529. Регистры ADC\_12A***

Регистр	Адрес	Назначение
ADC12CTL0	0700h	Регистры управления
ADC12CTL1	0702h	
ADC12CTL2	0704h	
ADC12IFG	070Ah	Флаги прерываний
ADC12IE	070Ch	Разрешение прерываний
ADC12IV	070Eh	Вектор прерываний
ADC12MCTL0	0710h	Управление памятью
ADC12MCTL1	0711h	
...	...	
ADC12MCTL15	071Fh	



# ***MSP430F5529. Регистры ADC\_12A***

Регистр	Адрес	Назначение
ADC12MEM0	0720h	Память
ADC12MEM1	0722h	
...	...	
ADC12MEM15	073Dh	

# MSP430F5529. Регистры ADC\_12A

Регистр	Биты	Поле	Назначение
ADC12CTL0	12-15	ADC12SHT1x	Количество циклов ADC12CLK (длительность выборки) для ADC12MEM8 ... ADC12MEM15
	8-11	ADC12SHT0x	Количество циклов ADC12CLK (длительность выборки) для ADC12MEM0 ... ADC12MEM7  4 / 8 / 16 / 32 / 64 / 96 / 128 / 192 / 256 / 384 / 512 / 768 / 1024
	7	ADC12MSC	Для режима последовательности и непрерывного: 0 – для запуска требуется фронт SHI сигнала, 1 – первый по фронту SHI, потом автоматически по окончании цикла начинается новый

# ***MSP430F5529. Регистры ADC\_12A***

Регистр	Биты	Поле	Назначение
ADC12CTL0	6	ADC12REF2_5V	Напряжение опорного генератора (ADC12REFON должен быть установлен) 0 – 1.5, 1 – 2.5 В
	5	ADC12REFON	Включение опорного генератора
	4	ADC12ON	Включение АЦП
	3	ADC12OVIE	Разрешение прерывания OV
	2	ADC12TOVIE	Разрешение прерывания TOV
	1	ADC12ENC	Разрешение измерения
	0	ADC12SC	Программный запуск выборки и преобразования. Сбрасывается автоматически

# ***MSP430F5529. Регистры ADC 12A***

Регистр	Биты	Поле	Назначение
ADC12CTL1	12-15	ADC12CSTARTADDx	Стартовый адрес записи результата
	10-11	ADC12SHSx	Источник сигнала запуска
	9	ADC12SHP	Выбор источника сигнала SAMPCON (расширенный или импульсный режим)
	8	ADC12ISSH	Инвертирование сигнала запуска
	5-7	ADC12DIVx	Делитель тактовой частоты
	3-4	ADC12SSELx	Выбор тактового сигнала
	1-2	ADC12CONSEQx	Режим
	0	ADC12BUSY	Индикатор активного режима

# ***MSP430F5529. Регистры ADC\_12A***

Регистр	Биты	Поле	Назначение
ADC12CTL2	8	ADC12PDIV	Деление тактовой частоты на 4
	7	ADC12TCOFF	Отключение температурного датчика
	4-5	ADC12RES	Точность (8, 10, 12 бит)
	3	ADC12DF	Результат - знаковый
	2	ADC12SR	Буфер работает на частоте 0: 200К , 1: 50К выборок / с
	1	ADC12REFOUT	Выход опорного напряжения
	0	ADC12REFBURST	Режим работы буферного усилителя опорного напряжения: постоянный (0) или автоматический (1)



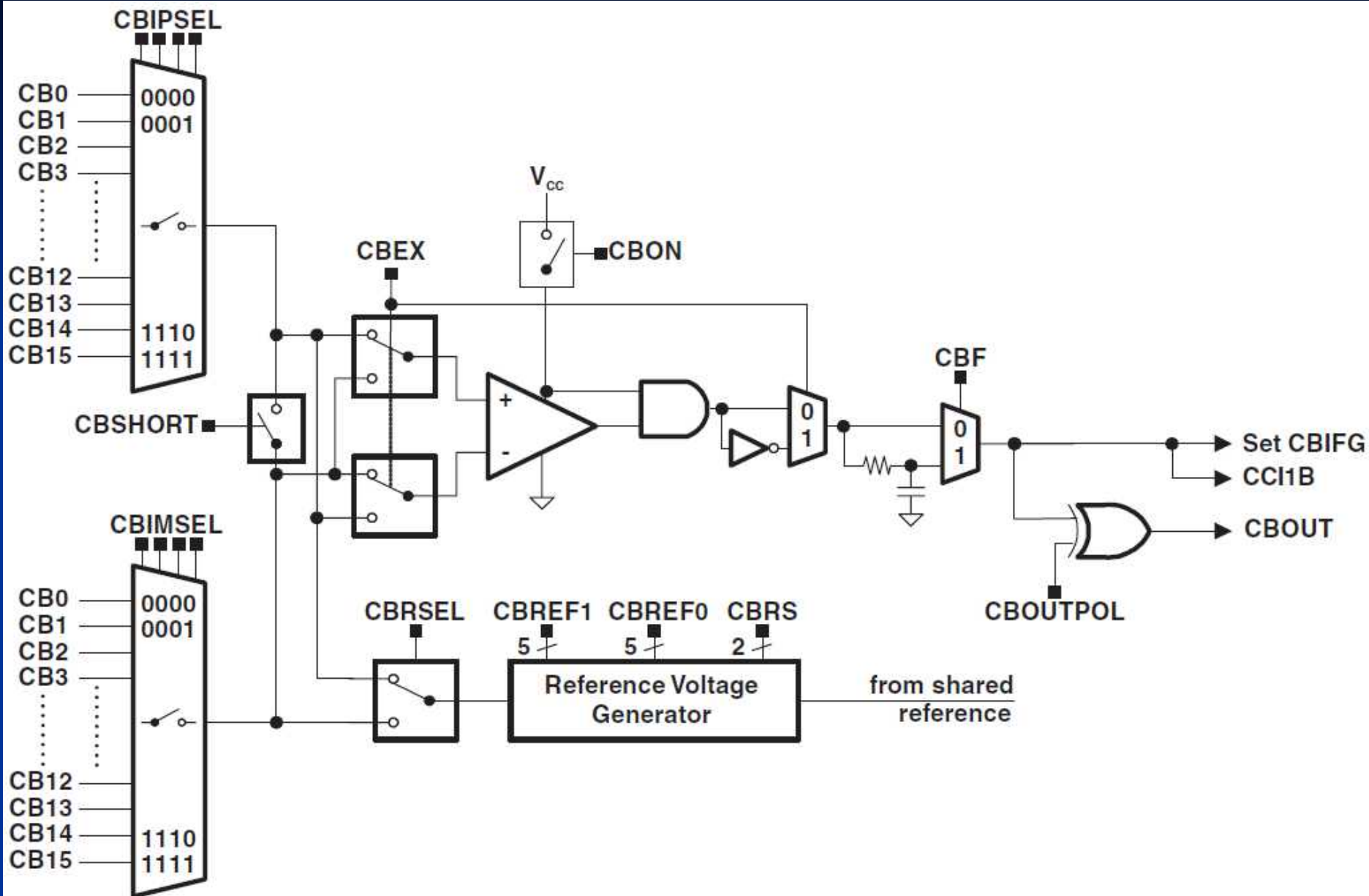
# ***MSP430F5529. Регистры ADC 12A***

Регистр	Биты	Поле	Назначение
ADC12MEMx	0-15		Результат измерения
ADC12MCTLx	7	ADC12EOS	Маркер конца последовательности
	4-6	ADC12SREFx	Выбор пары опорных напряжений VR+, VR-
	0-3	ADC12INCHx	Выбор входного канала
ADC12IE			Разрешение прерывания по соответствующему флагу
ADC12IFG			Флаг запроса на прерывание
ADC12IV			Вектор запросов на прерывания

# *MSP430F5529. Компаратор*

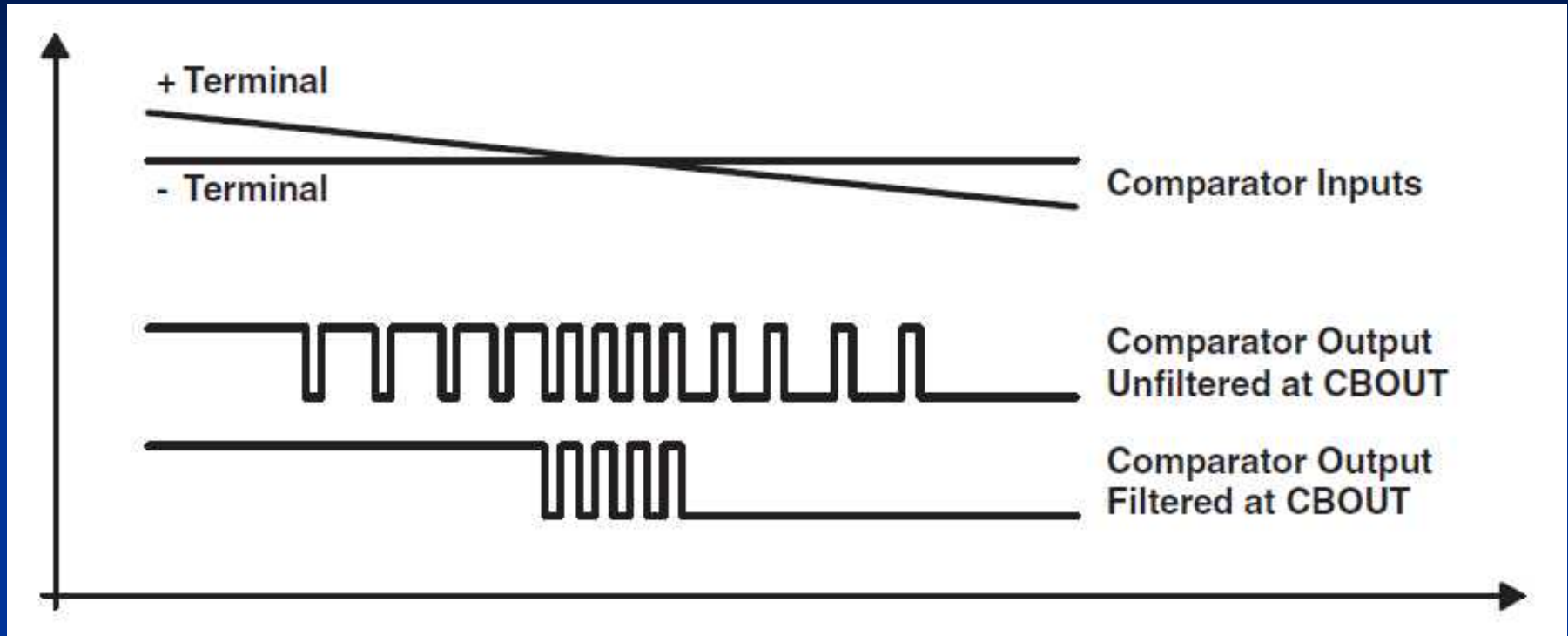
- *Прямое и инверсное сравнение*
- *Программное подключение RC-фильтра на выходе*
- *Выход подключается ко входу таймера A*
- *Программный выбор каналов (из 16 возможных)*
- *Использование прерываний*
- *Программируемый генератор опорного напряжения*

# MSP430F5529. Компаратор



# MSP430F5529. Компаратор

## ■ RC-фильтр



# ***MSP430F5529. Компаратор***

Регистр	Адрес	Назначение
CBCTL0	08C0h	Регистры управления
CBCTL1	0802h	
CBCTL2	0804h	
CBCTL3	0806h	
CBINT	080Ch	Управление прерываниями
CBIV	080Eh	Вектор прерываний



# ***MSP430F5529. Компаратор***

Регистр	Биты	Поле	Назначение
CBCTL0	15	CBIMEN	Разрешение входного канала на V-
	8-11	CBIMSEL	Выбор входного канала V-
	7	CBIPEN	Разрешение входного канала на V+
	0-3	CBIPSEL	Выбор входного канала V+
CBCTL1	12	CBMRVS	0 – выход компаратора управляет выбором между VREF0 и VREF1 1 – управляет CBMRVL
	11	CBMRVL	Выбор VREFx
	10	CBON	Включение компаратора

# ***MSP430F5529. Компаратор***

Регистр	Биты	Поле	Назначение
CBCTL1	8-9	CBPWRMD	Режим питания
	6-7	CBFDLY	Величина задержки фильтра (0.6, 1.0, 1.8, 3.4 мкс)
	5	CBEX	Инверсный режим
	4	CBSHORT	Закорачивание входных каналов
	3	CBIES	Выбор прерывания по фронту или спаду
	2	CBF	Выходной фильтр
	1	CBOUTPOL	Полярность выхода
	0	CBOUT	Выход компаратора

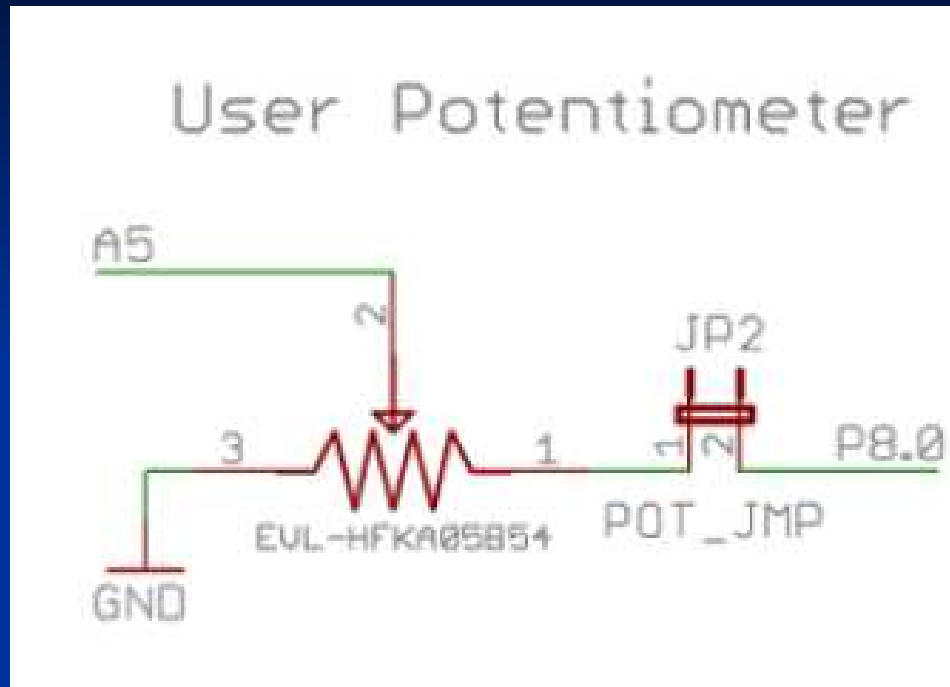
# ***MSP430F5529. Компаратор***

Регистр	Биты	Поле	Назначение
CBCTL2	15	CBREFACC	Точность
	13-14	CBREFL	Опорное напряжение (отключено, 1.5, 2, 2.5 В)
	8-12	CBREF1	Устанавливает напряжение на выходе резисторного делителя
	6-7	CBRS	Источник опорного напряжения
	5	CBRSEL	Коммутация опорного напряжения к + и - входам
	0-4	CBREF0	Как и CBREF1

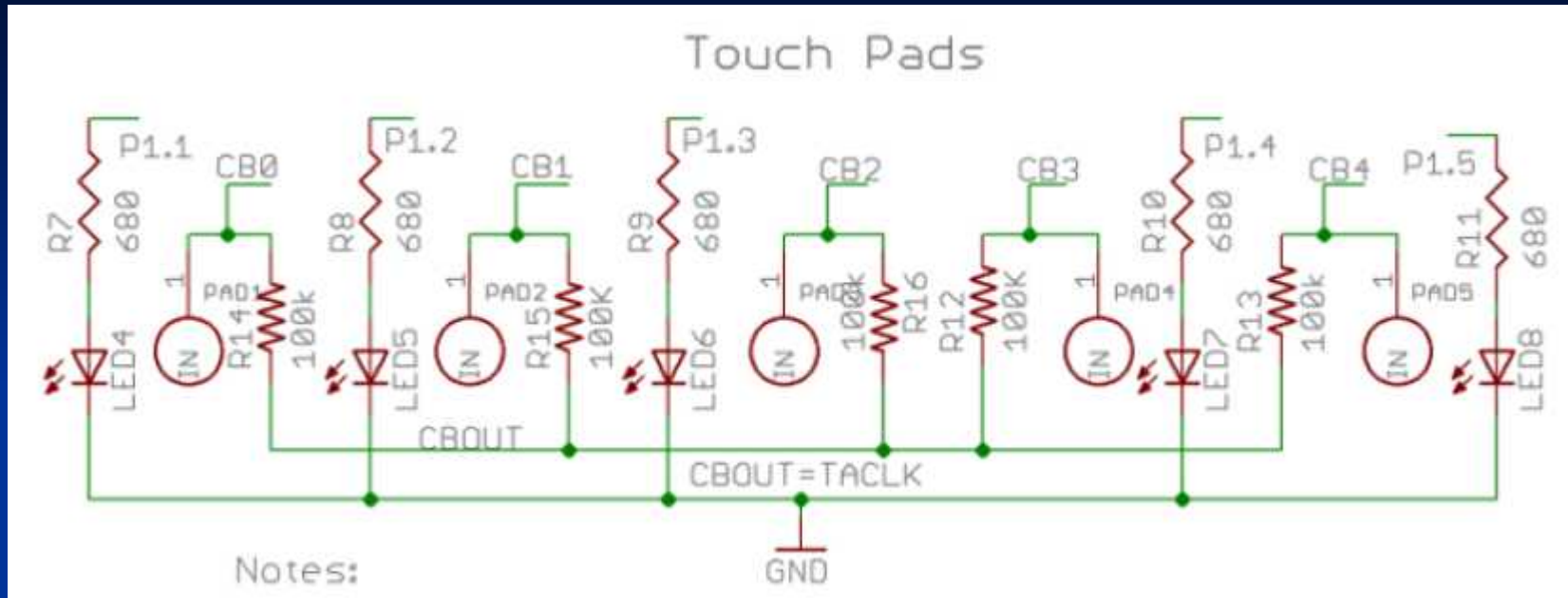
# ***MSP430F5529. Компаратор***

Регистр	Биты	Поле	Назначение
CBCTL3		CBPD	Отключение входного буфера для каждого вывода
CBINT	9	CBIE	Выход разрешения прерывания по инверсному выходу
	8	CBIE	Выход разрешения прерывания
	1	CBIFG	Флаг инверсного прерывания
	0	CBIFG	Флаг прерывания
CBIV			Вектор прерываний

# Аналоговый ввод-вывод MSP430F5529. ЦАП, АЦП, компаратор

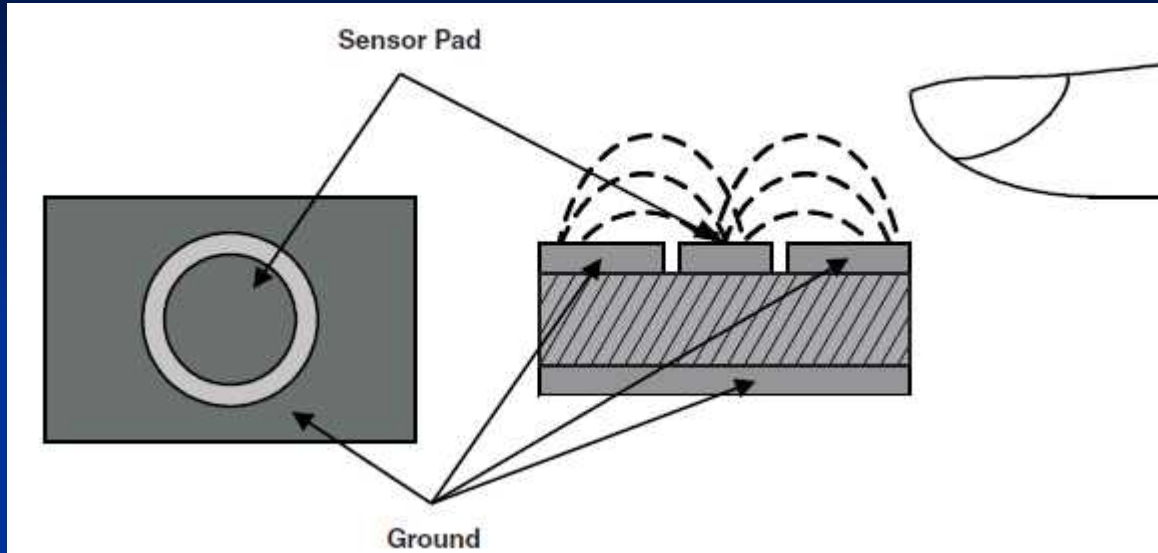


# Аналоговый ввод-вывод MSP430F5529. ЦАП, АЦП, компаратор



# Сенсорный элемент

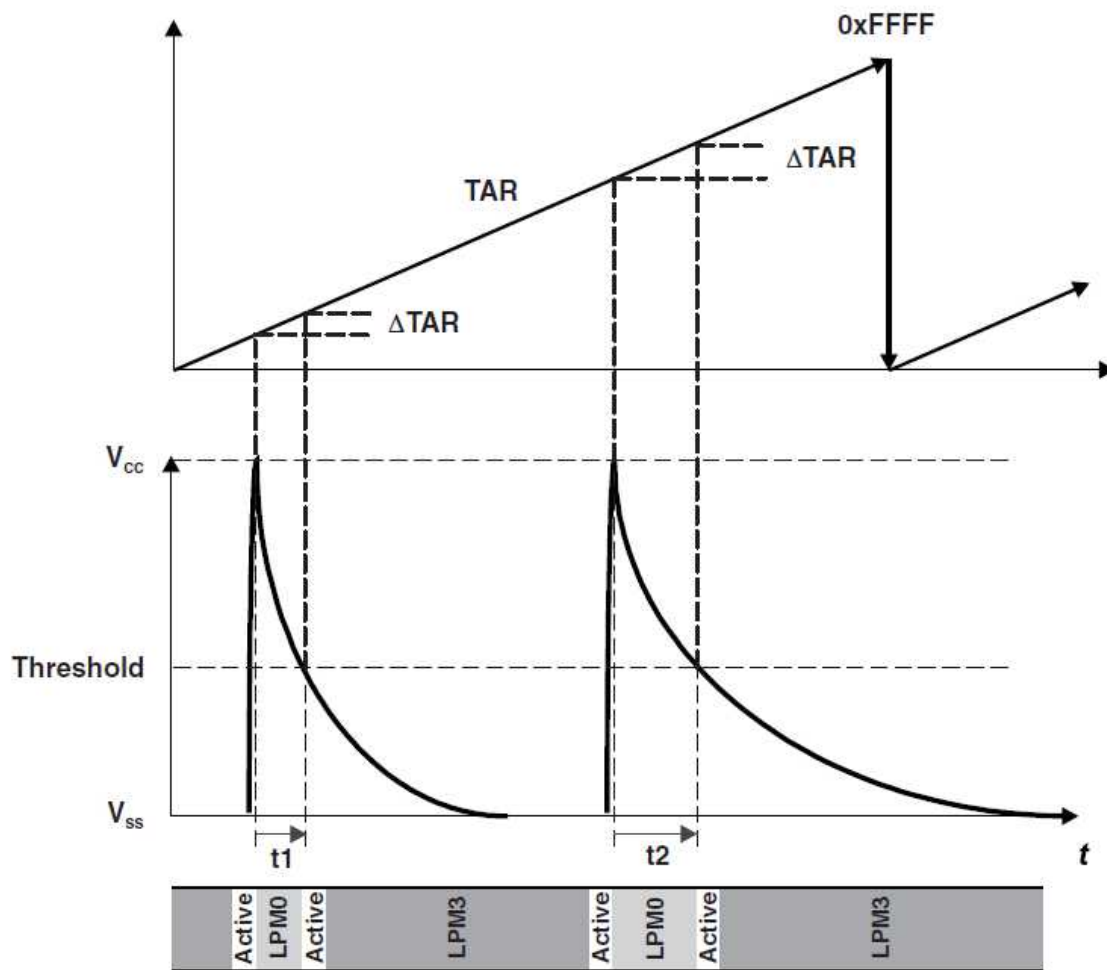
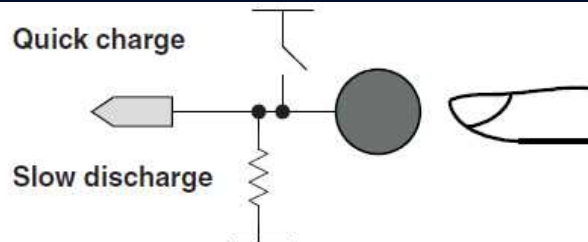
## ■ Конденсатор



- *При приближении к сенсору меняется магнитное поле и, следовательно, емкость*
- *Зависимость от условий внешней среды требует отслеживания базового уровня*

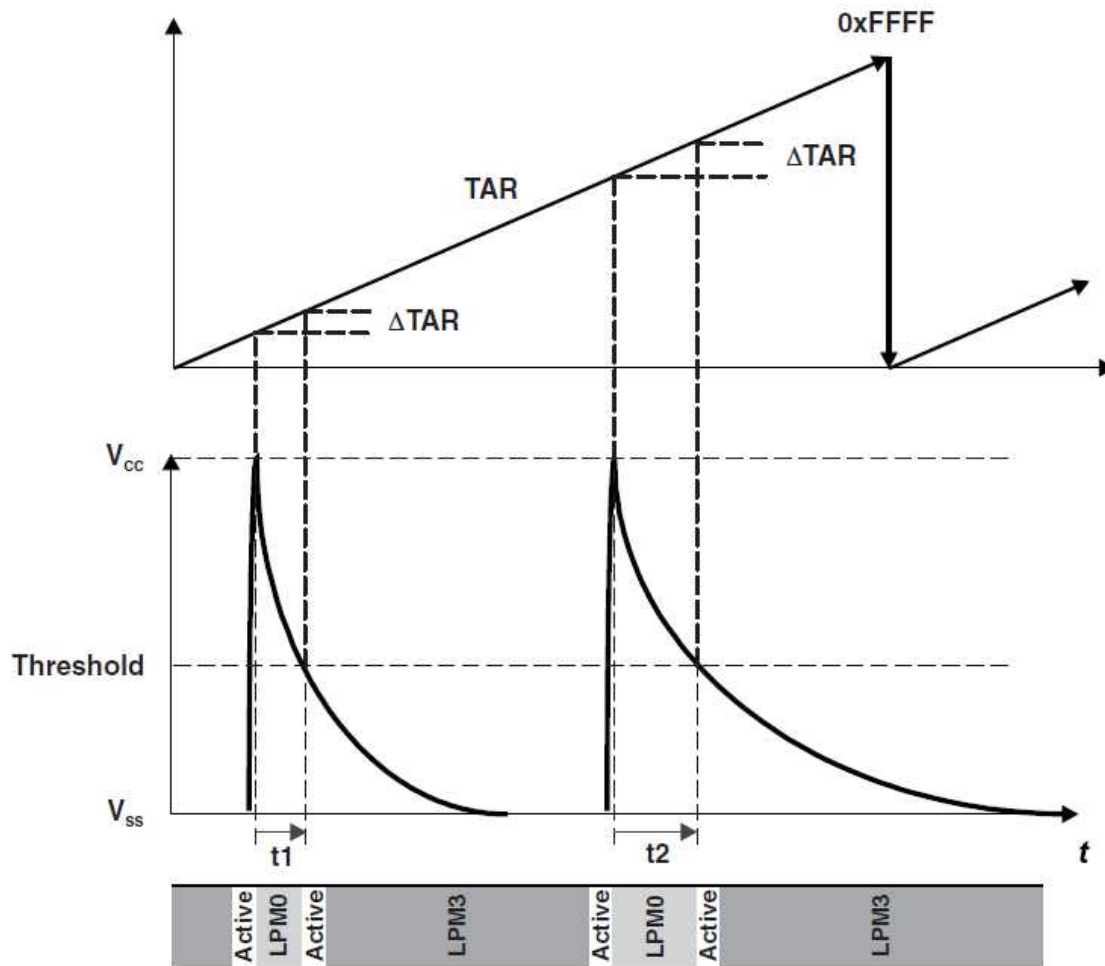
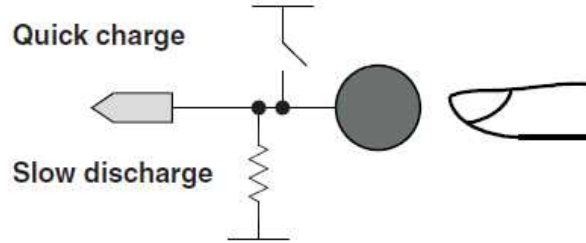


# Сенсорный элемент



- **Время разряда RC цепи используется для измерения емкости сенсора**
  - Резистор включается между землей и сенсором, сенсор подключается к I/O
  - На I/O подается «1» и конденсатор быстро заряжается. Читается время таймера

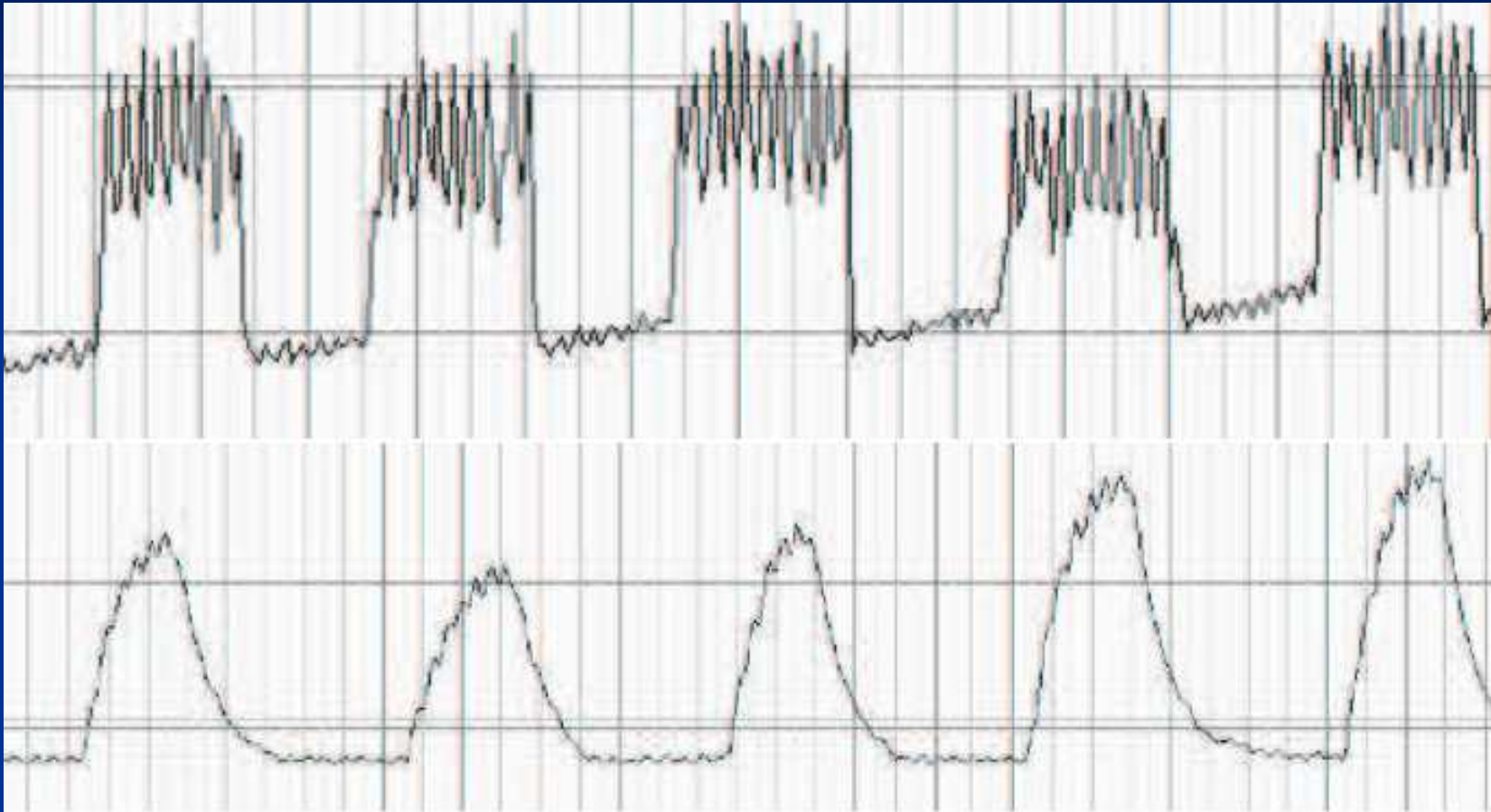
# Сенсорный элемент



- 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
- Микроконтроллер выводится из LPM0

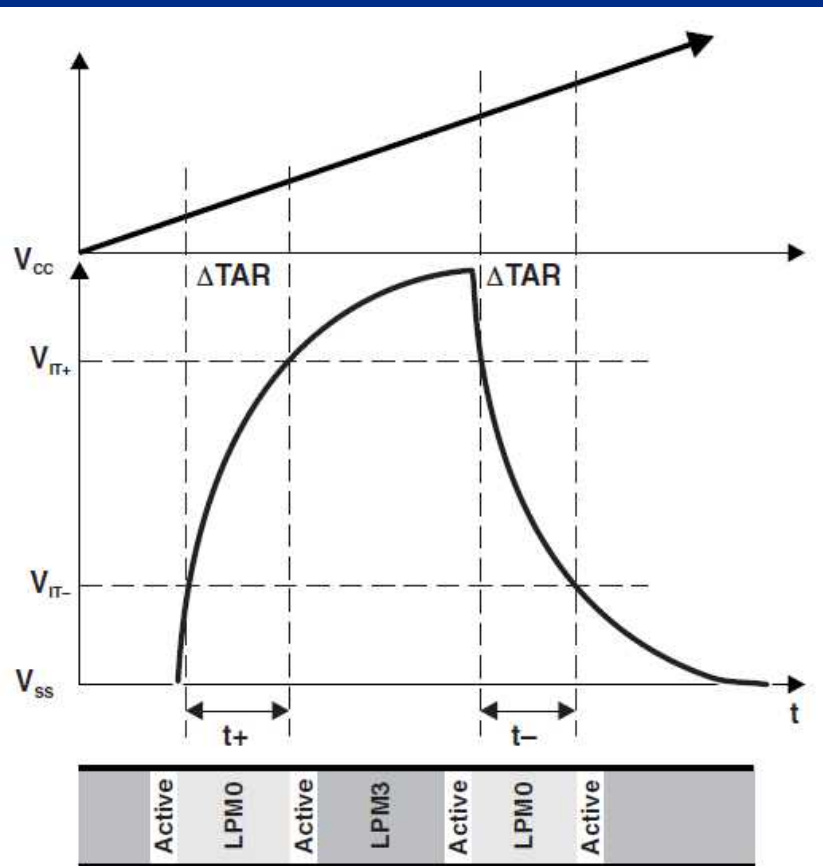
# Сенсорный элемент

- **Фильтрация сигнала (IIR – БИХ фильтр)**

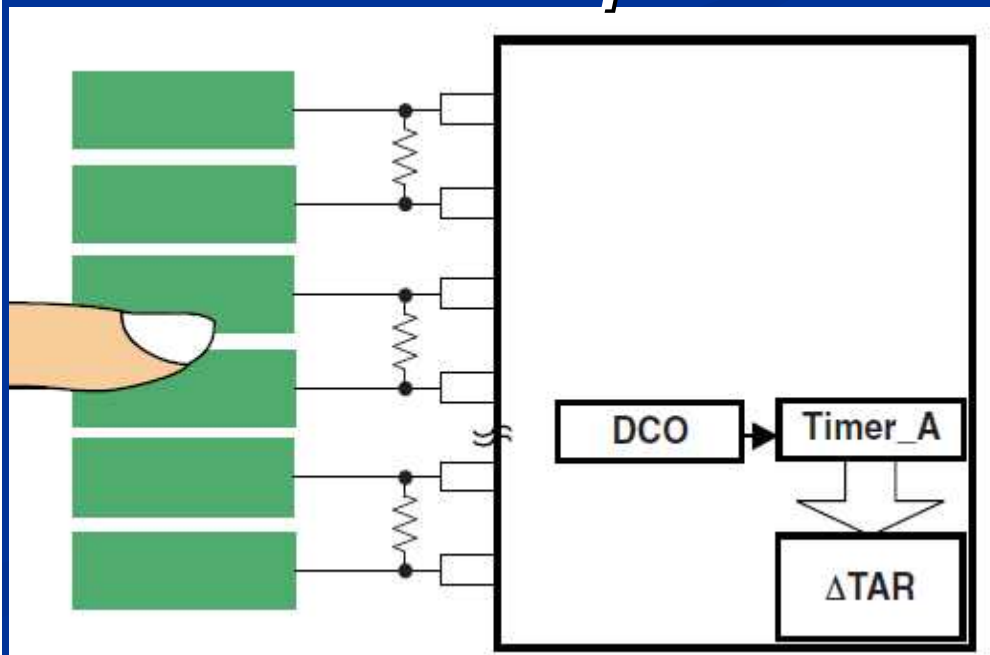


# Сенсорный элемент

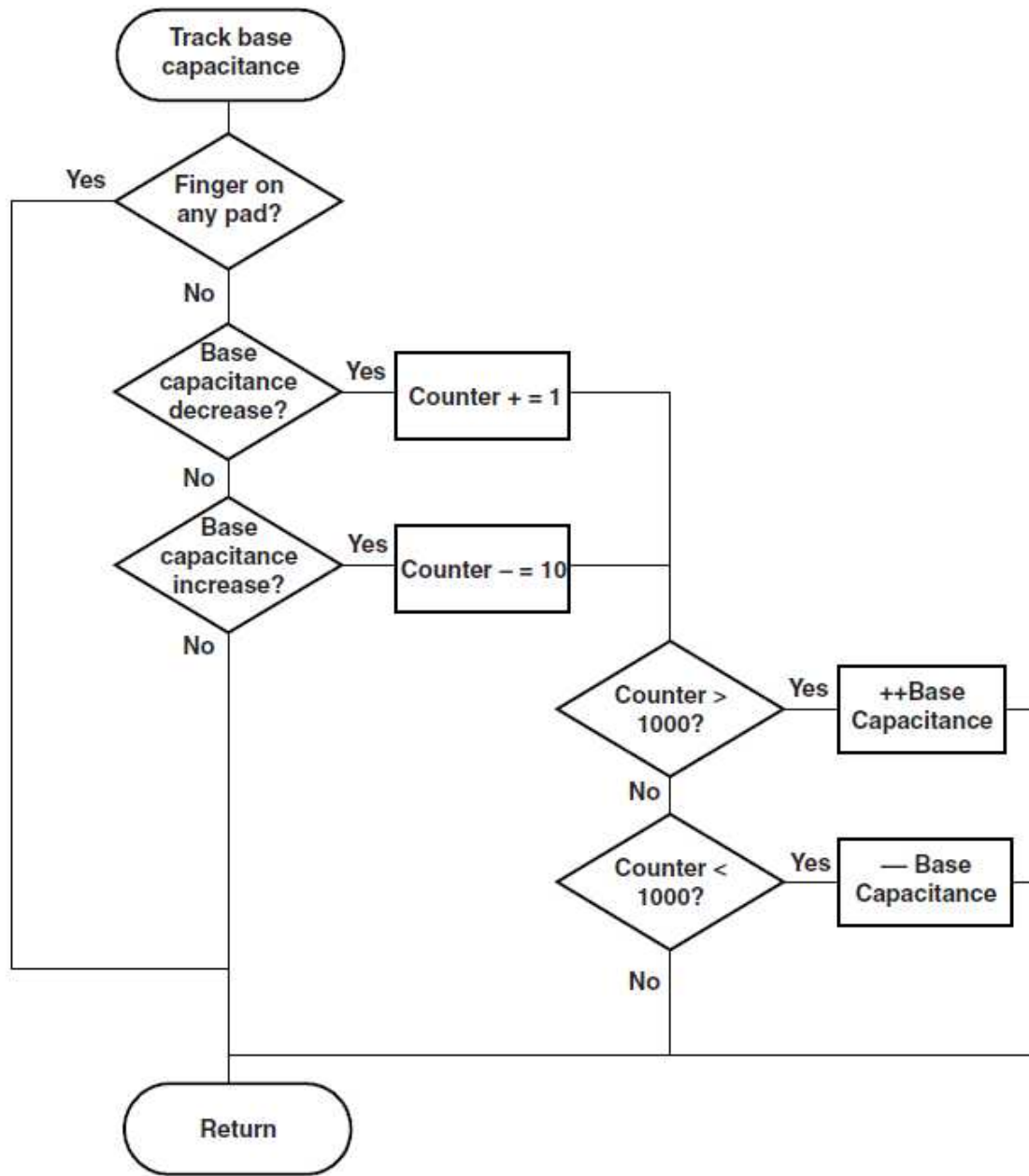
- **Снижение чувствительности к шуму: усреднение измерений заряда и разряда**



- **Уменьшение в 2х количества резисторов: один сенсор заряжается, второй измеряется, потом наоборот**

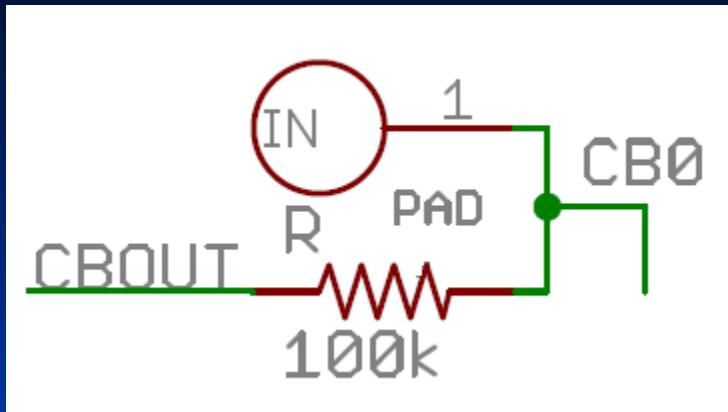


# Сенсорный элемент



- **Отслеживание дрейфа базовой емкости**

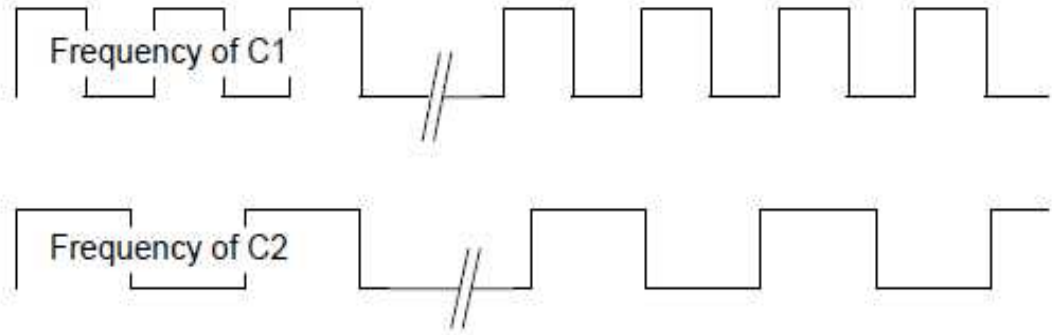
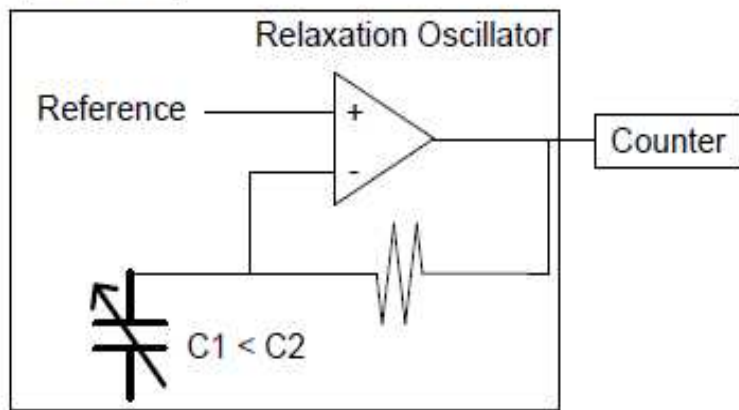
# Сенсорный элемент



- ***P1.6 / TA1CLK / CBOUT***
- ***P6.0 / CB0 / A0***

# Сенсорный элемент

TimeBase 1



TimeBase 2

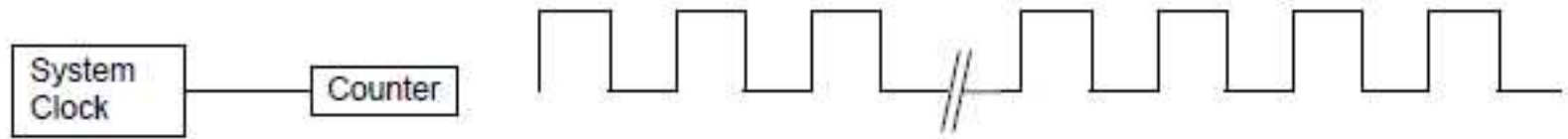


- ***RO – метод (Relaxation Oscillator).  
Измерение тактов релаксационного  
генератора за промежуток времени***

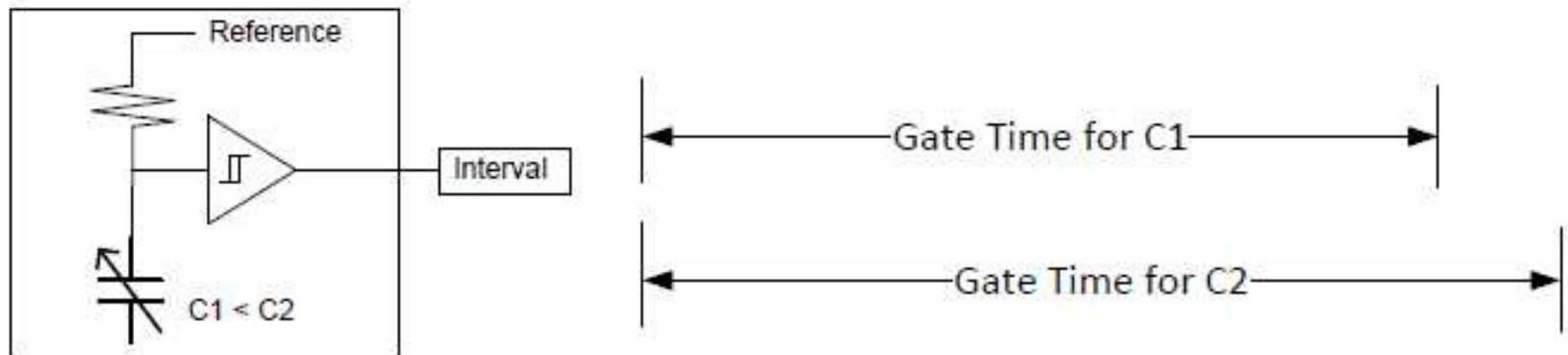


# Сенсорный элемент

TimeBase 1



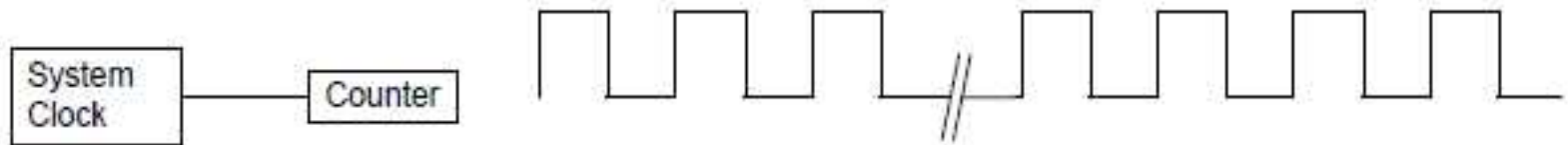
TimeBase 2



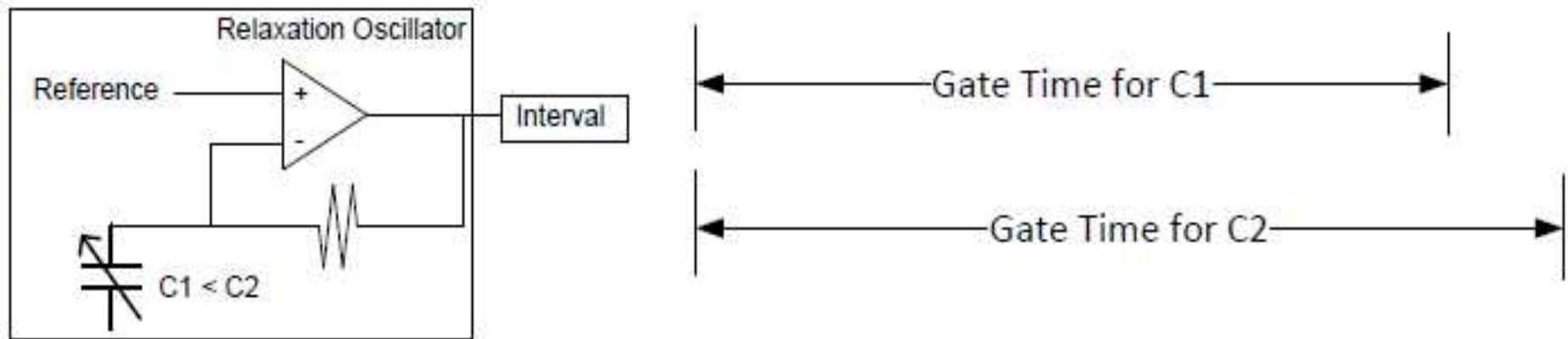
- ***RC – метод. Измерение времени разряда RC-цепи***

# Сенсорный элемент

TimeBase 1



TimeBase 2



- ***fRO – метод. Измерение периода колебаний релаксационного генератора***

## Пример. Компаратор. Сенсоры. Потенциометр

- *На входы компаратора подаются сигналы с потенциометра и сенсорного элемента*
- *Если напряжение на кнопке выше, то включается светодиод*
- *2 режим:*
- *На входы компаратора подаются сигналы с двух сенсорных кнопок*
- *На какой из них напряжение больше, у той и загорается светодиод*

# Сенсор, компаратор, потенциометр

```
#include <msp430.h>
int poten_mode; // Global potentiometer mode
int main(void) {
    WDTCTL = WDTPW | WDTHOLD; // Stop watchdog timer

    P1DIR &= ~(BIT7); // P1.7 (S1) set as input
    P1OUT |= BIT7;    // P1.7 (S1) pull-up resistor
    P1REN |= BIT7;    // P1.7 (S1) enable resistor

    // P6.0 (PAD1) & P6.1 (PAD2) & P6.5 (POT) set as input
    P6DIR &= ~(BIT0 | BIT1 | BIT5);
    // device mode: P6.0 is PAD1, P6.1 is PAD2, P6.5 is POT
    P6SEL |= (BIT0 | BIT1 | BIT5);

    P1DIR |= BIT6;    // P1.6 (CBOUT) set as output
    P1SEL |= BIT6;    // device mode: P1.6 is CBOUT

    P8DIR |= BIT0;    // P8.0 (Vcc POT) set as output
    P8SEL &= ~BIT0;   // P8.0 is digital I/O
    P8OUT |= BIT0;    // Set Vcc to POT
```

# Сенсор, компаратор, потенциометр

```
// P1.1 (LED PAD1) & P1.2 (LED PAD2) set as output
P1DIR |= BIT1 | BIT2;
P1SEL &= ~(BIT1 | BIT2); // LED is digital I/O

// Enable both comparator inputs,
// CBIP(+) = A5 (POT), CBIM(-) = A0 (PAD1)
CBCTL0 = CBIMEN | CBIPEN | CBIPSEL_5 | CBIMSEL_0;
// Enable Comparator, Enable out filter 3600 ns delay
CBCTL1 = CBON | CBF | CBFDLY_3;
poten_mode = 1;

P1IE |= BIT7; // P1.7 (S1) interrupt enable
// P1.7 (S1) edge for interrupt: low-to-high
P1IES &= ~(BIT7);
TA0CTL0 = CCIE; // CCR0 interrupt enabled
// SMCLK, no divide, contmode, clear TAR
TA0CTL = TASSEL_2 | ID_0 | MC_2 | TACLK ;
```

# Сенсор, компаратор, потенциометр

```
__bis_SR_register(LPM0_bits + GIE);  
__no_operation();  
return 0; }
```

```
#pragma vector=PORT1_VECTOR  
__interrupt void PORT1_ISR(void) {  
    poten_mode ^= BIT0; // change mode: POT or PAD2  
    if(poten_mode)  
    {  
        CBCTL0 |= CBIPSEL_5; // set CBIP(+) = A5 (POT)  
    }  
    else  
    {  
        CBCTL0 |= CBIPSEL_1; // set CBIP(+) = A1 (PAD2)  
    }  
    P1IFG = 0; // reset interrupt flag  
}
```

# Сенсор, компаратор, потенциометр

```
// Timer0 interrupt service routine
#pragma vector=TIMER0_A0_VECTOR
__interrupt void TIMER0_A0_ISR(void) {
    P1OUT &= ~(BIT1 | BIT2); // LED off
    if ( (CBCTL1 & CBOUT) == 0)
    {
        P1OUT |= BIT1; // LED PAD1 on
    }
    else if (!poten_mode)
        P1OUT |= BIT2; // LED PAD2 on
}
```



***Видео***  
***03. Analog IO 2***

## Домашнее задание № 3

---

3.1. Почему оба режима реагируют на потенциометр?

3.2. Какой из методов измерения емкости сенсора использован?

# Сенсор, компаратор, потенциометр

```
__bis_SR_register(LPM0_bits + GIE);  
__no_operation();  
return 0; }
```

```
#pragma vector=PORT1_VECTOR  
__interrupt void PORT1_ISR(void) {  
    poten_mode ^= BIT0;    // change mode: POT or PAD2  
    CBCTL0 ^= CBIPSEL2;    // ??????  
    P1IFG = 0;             // reset interrupt flag  
}
```

***Видео  
04. Analog IO 2***

# Домашнее задание № 3

3.3. Что такого сделала строчка

```
CBCTL0 ^= CBIPSEL2; // ??????
```

3.4. Почему после запуска включается 2 режим (PAD)?

3.5. Почему PAD2 оказывает влияние в режиме ROT ?

3.6. Почему при отпущенных кнопках сигнал с PAD1 стабильно выше?

3.7. Почему при нажатом PAD2 он не становится стабильно выше?