



Применение смешанной аналого-цифровой периферии.

Приемы и трюки



Для чего этот класс...

- | ... Понимания основ связи МК с аналоговым миром
- | ... Использования некоторых аналоговых модулей контроллеров PIC[®]
- | ... Обзор некоторых решений

План

- | **Преобразование Аналоговой величины в Цифровую**
- | **Периферия PIC[®] Микроконтроллеров**
- | **Преобразование Аналоговой величины в Цифровую с помощью периферии PIC контроллера**
- | **Применение аналоговой периферии PIC микроконтроллеров**
- | **Итоги**

Преобразование Аналоговой величины в Цифровую

Преобразование Аналога в Цифру

| Аналоговые датчики изменяют

| Сопротивление

| Термистор



| Емкость

| Датчик влажности

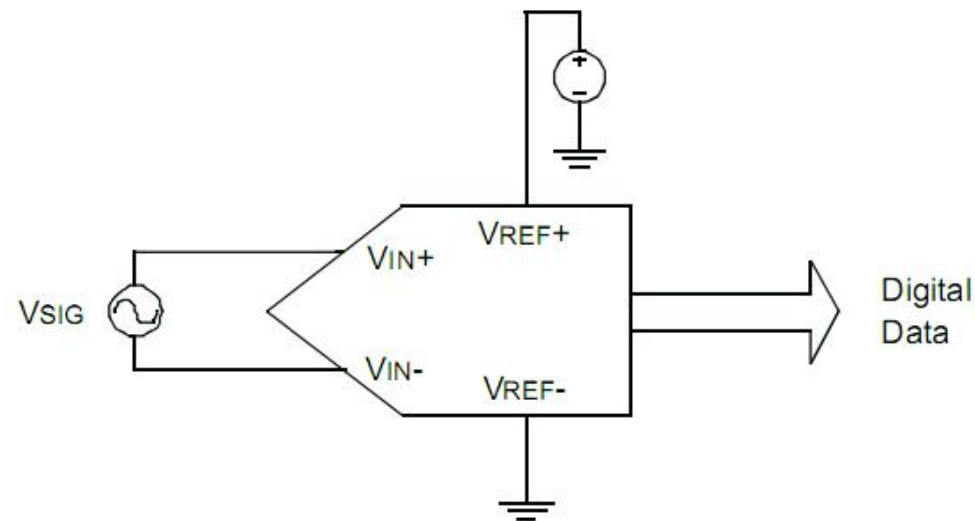
| Индуктивность

| Датчики расстояния



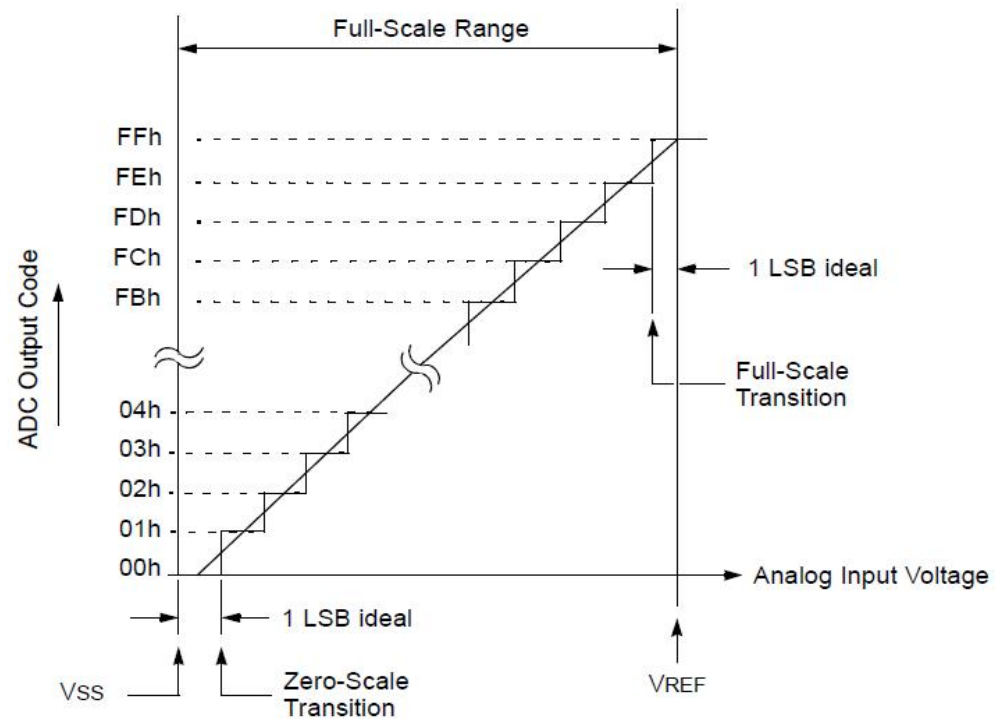
Преобразование Аналога в Цифру

- | Традиционно применяется АЦП
- | Выход датчика → Напряжение → Усилитель → АЦП (преобразующий в 8-, 10-, 12 бит значение)



Что важно в АЦП?

Основное: преобразование аналогового сигнала и полный диапазон входного сигнала

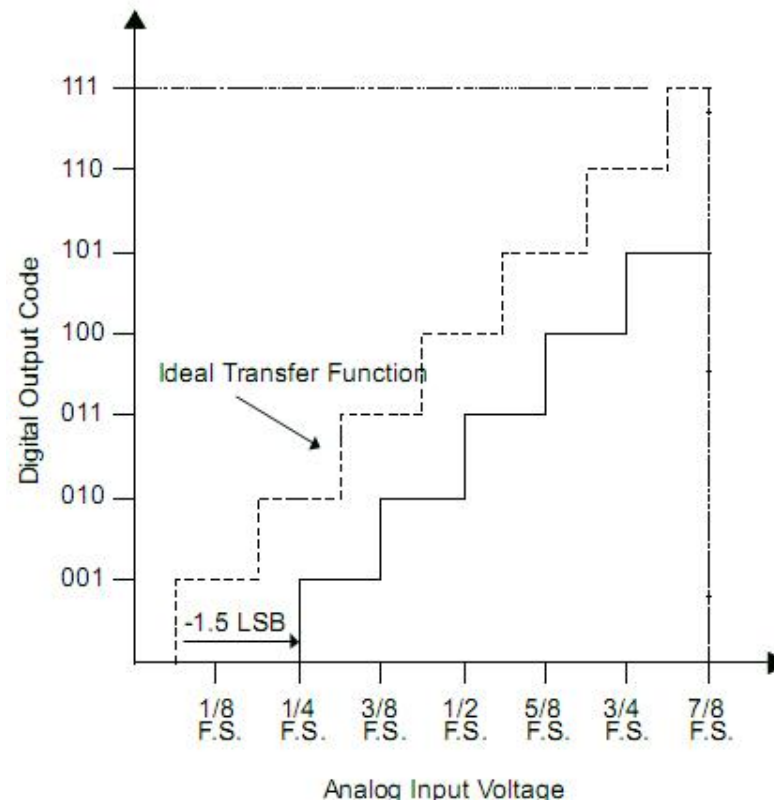


Что важно в АЦП?

- | Разрешение: Число выходных разрядов после преобразователя
- | Повторяемость:
 - | SNR
- | Точность:
 - | Ошибка сдвига: Разница между идеальной и реальной характеристиками преобразования
 - | Ошибка усиления: Разница между идеальным и реальным наклоном характеристики преобразования

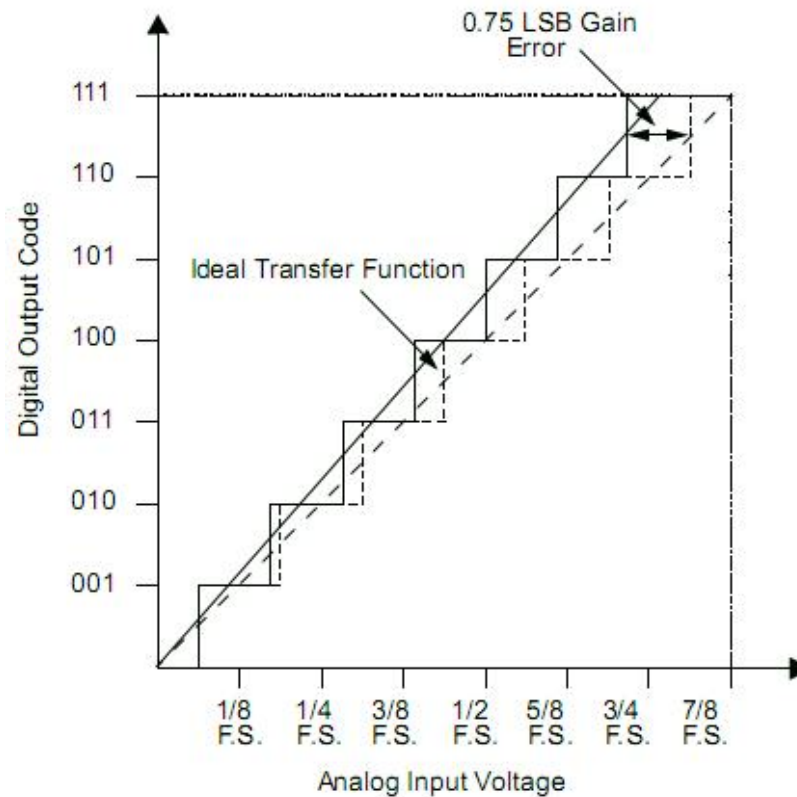
Что важно в АЦП?

Ошибка сдвига :



Что важно в АЦП?

Ошибка усиления :



Преобразование Аналога в Цифру

Характеристики:

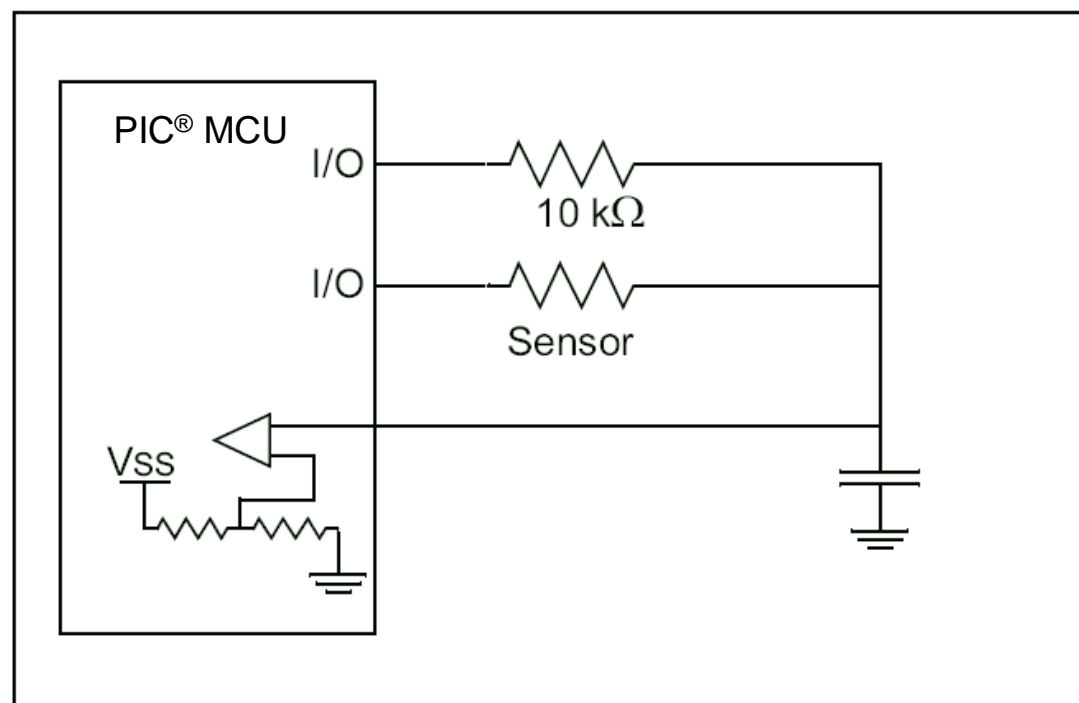
TABLE 28-8: PIC16F193X/LF193X A/D CONVERTER (ADC) CHARACTERISTICS:

Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$							
Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
AD01	NR	Resolution	—	—	10	bit	
AD02	EIL	Integral Error	—	—	± 1	LSb	$V_{REF} = 3.0\text{V}$
AD03	EDL	Differential Error	—	—	± 1	LSb	No missing codes $V_{REF} = 3.0\text{V}$
AD04	EOFF	Offset Error	—	—	± 3	LSb	$V_{REF} = 3.0\text{V}$
AD05	EGN	Gain Error	—	—	± 3	LSb	$V_{REF} = 3.0\text{V}$
AD06	V_{REF}	Reference Voltage ⁽³⁾	1.8	—	V_{DD}	V	
AD07	V_{AIN}	Full-Scale Range	V_{SS}	—	V_{REF}	V	
AD08	Z_{AIN}	Recommended Impedance of Analog Voltage Source	—	—	50	$k\Omega$	Can go higher if external $0.01\mu\text{F}$ capacitor is present on input pin.
AD09*	I_{REF}	V_{REF} Input Current ⁽³⁾	10	—	1000	μA	During V_{AIN} acquisition.

Преобразование Аналога в Цифру

| Типы:

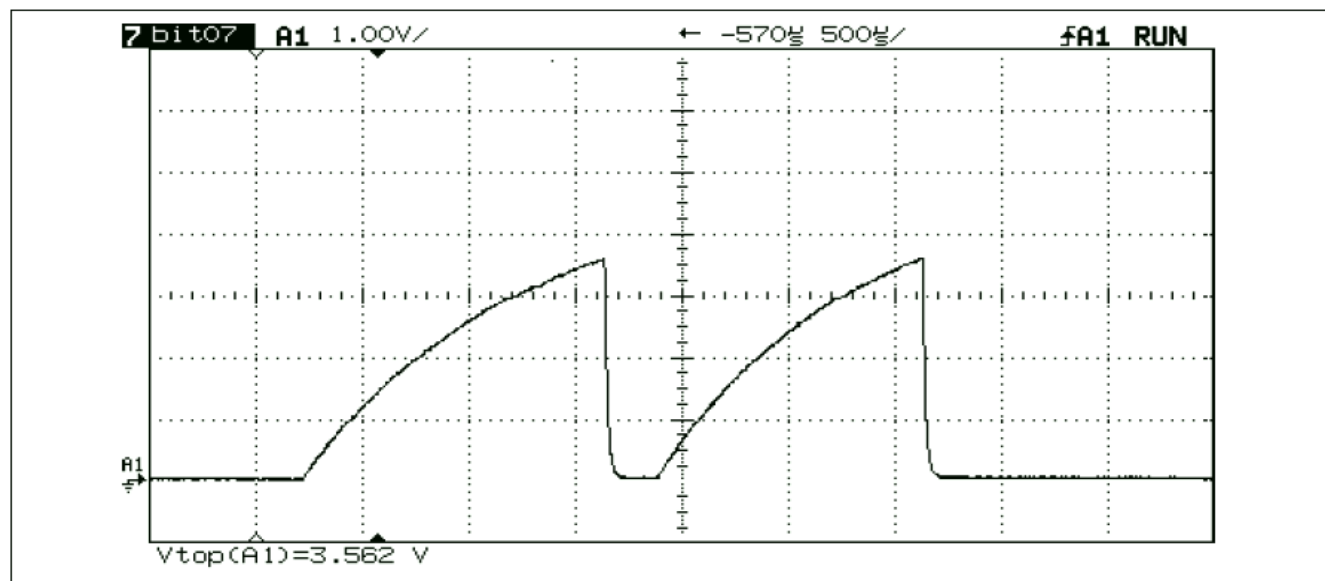
| Интегрирующий АЦП



Преобразование Аналога в Цифру

| Типы:

| Интегрирующий АЦП



Преобразование Аналога в Цифру

| Преобразование

| Передаточная характеристика

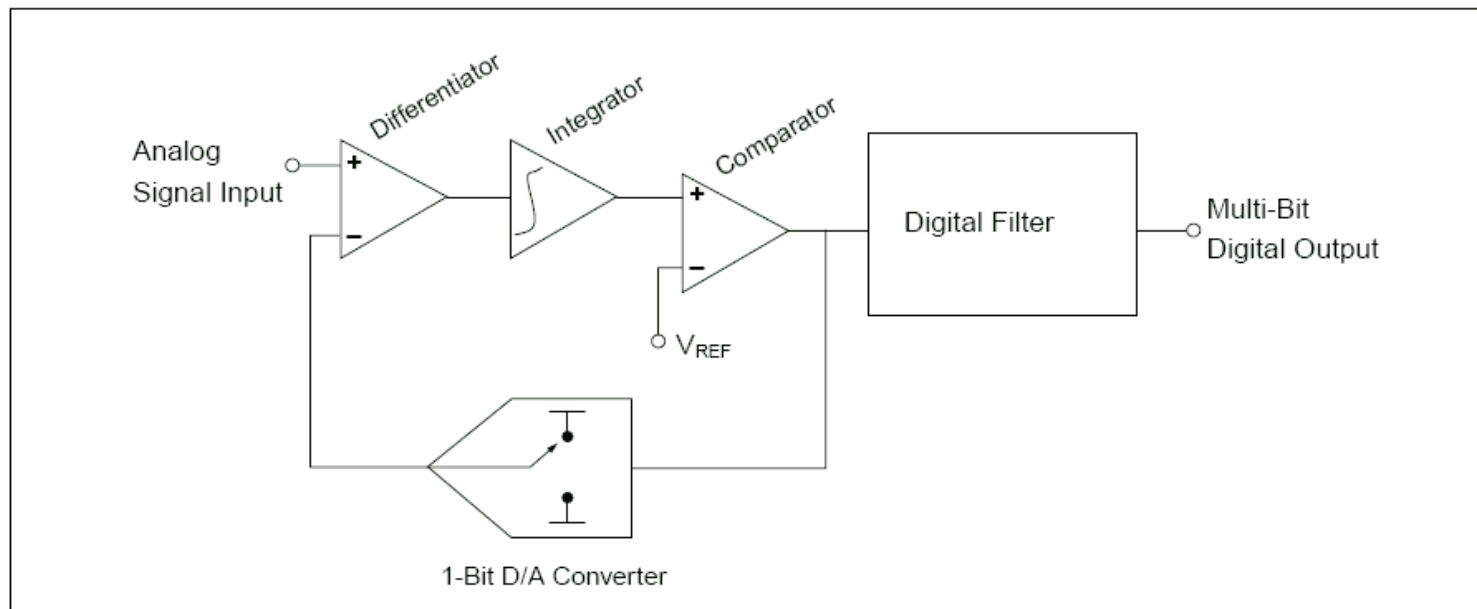
$$| R_{NTC} = R_{REF} \times t_{NTC}/t_{REF}$$

- | Время преобразования не зависит от Vdd
- | Transfer function independent of cap or MCU osc.
- | Разрешение зависит от RC и таймера
- | Точность зависит от R_{REF}

Преобразование Аналога в Цифру

| Тип:

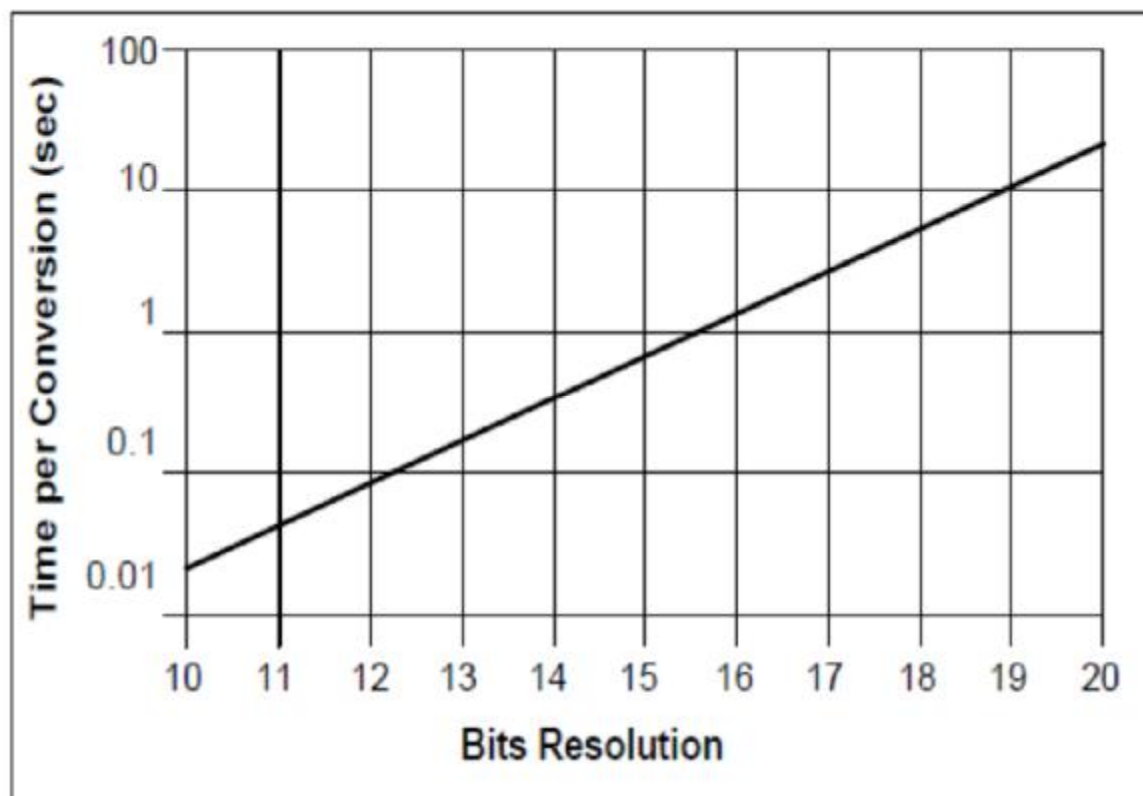
| Сигма-Дельта АЦП



Преобразование Аналога в Цифру

| Тип:

| Сигма-Дельта АЦП



Сигма-Дельта АЦП

| Сигма-Дельта АЦП

| Ошибка вычисления:

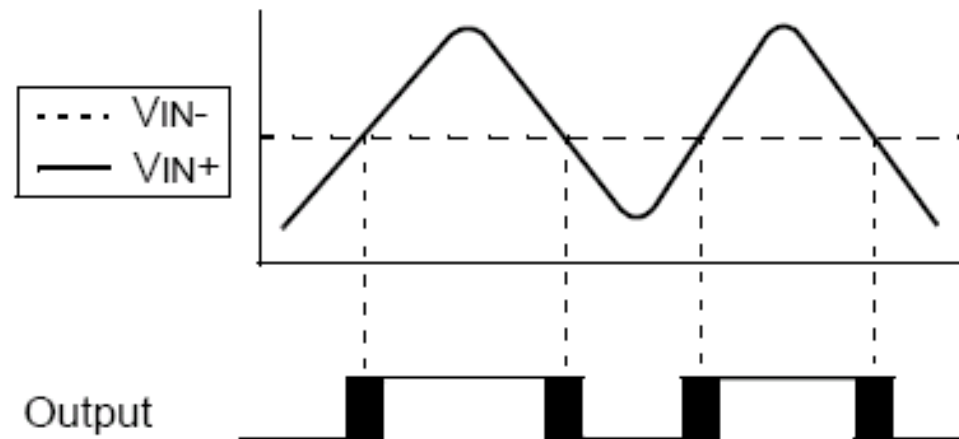
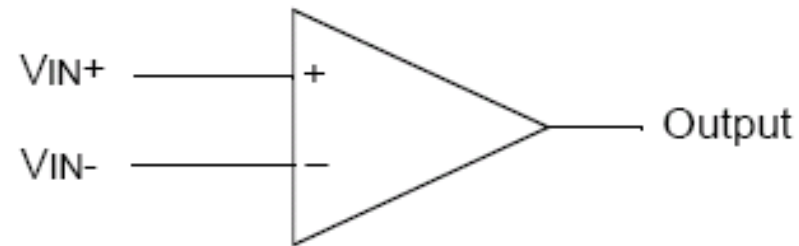
- | Токи утечки портов

- | Точность источника опорного напряжения

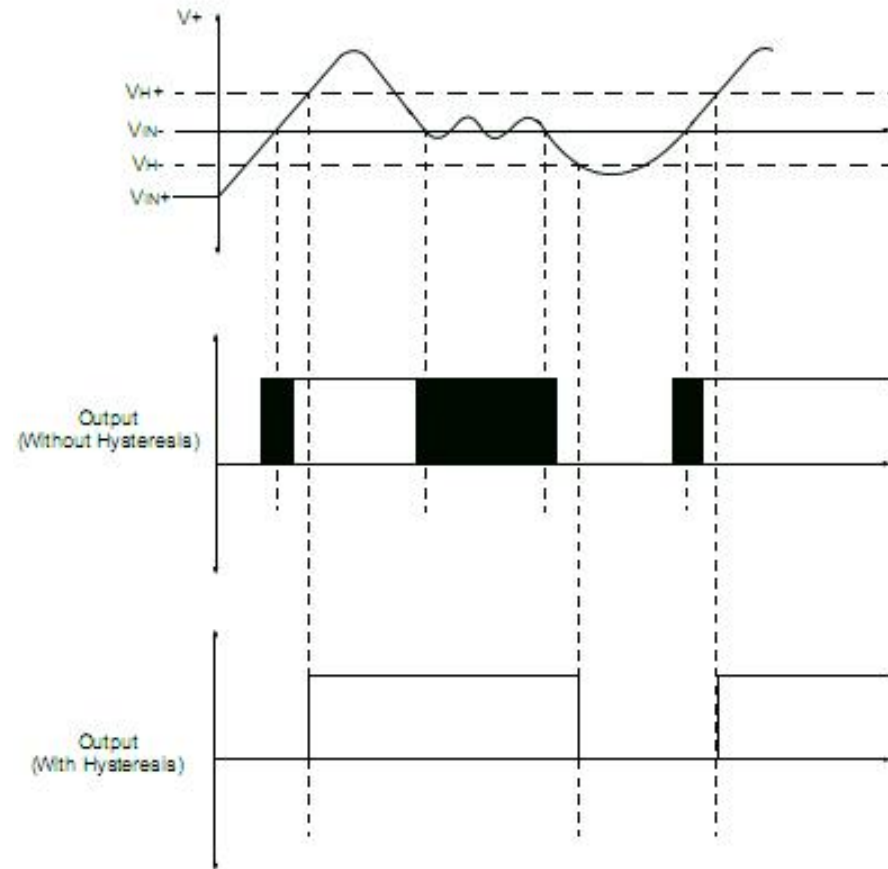
Периферия PIC[®] микроконтроллеров

Аналоговые компоненты

Компаратор:

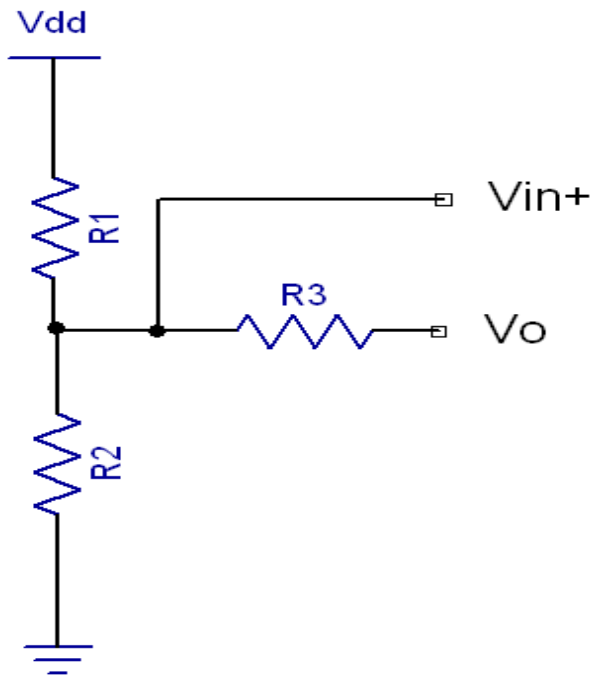


Компаратор, гистерезис:



Аналоговые компоненты

Компаратор: Увеличение гистерезиса



$$\text{if } V_o = V_{dd}$$

$$V_{in} = \left[\frac{R_2}{R_2 + R_1 \parallel R_f} \right] \cdot V_{dd}$$

$$= \frac{2}{3} \cdot V_{dd} = V_{th}$$

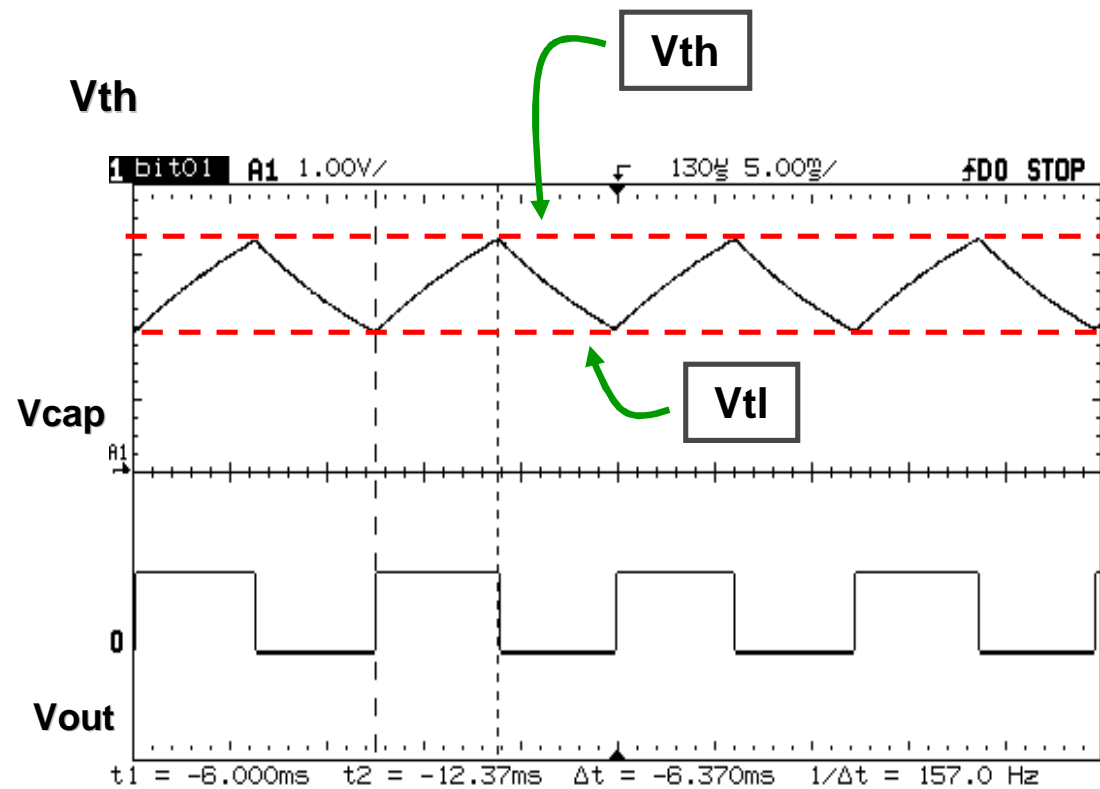
$$\text{if } V_o = 0V$$

$$V_{in} = \left[\frac{R_2 \parallel R_3}{R_1 + R_2 \parallel R_f} \right] \cdot V_{dd}$$

$$= \frac{1}{3} \cdot V_{dd} = V_{tl}$$

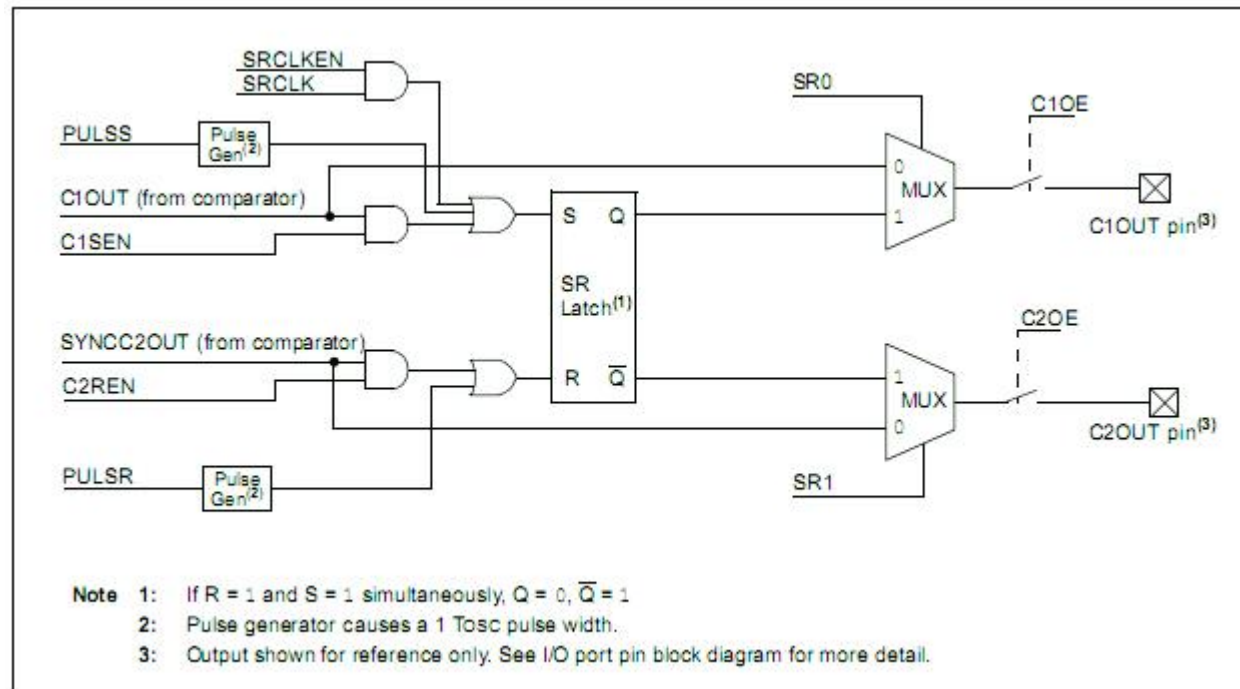
Аналоговые компоненты

Компаратор:

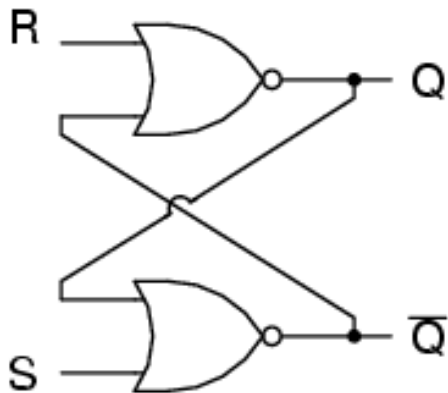


Аналоговые компоненты

S-R Триггер:



Что такое SR-триггер?



S	R	Q	!Q
0	0	Latched	Latched
0	1	0	1
1	0	1	0
1	1	0	0

NOR Gates give a R dominant latch

| “Наш” SR триггер:

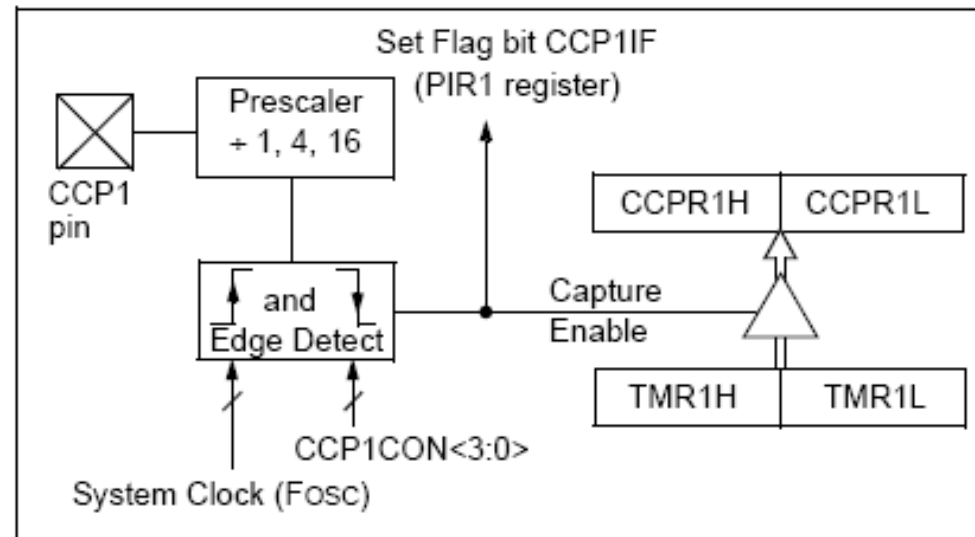
| NOR based

| !Q это не выход триггера, это инверсный выход Q:

S	R	Q	!Q
0	0	Latched	Latched
0	1	0	1
1	0	1	0
1	1	0	1

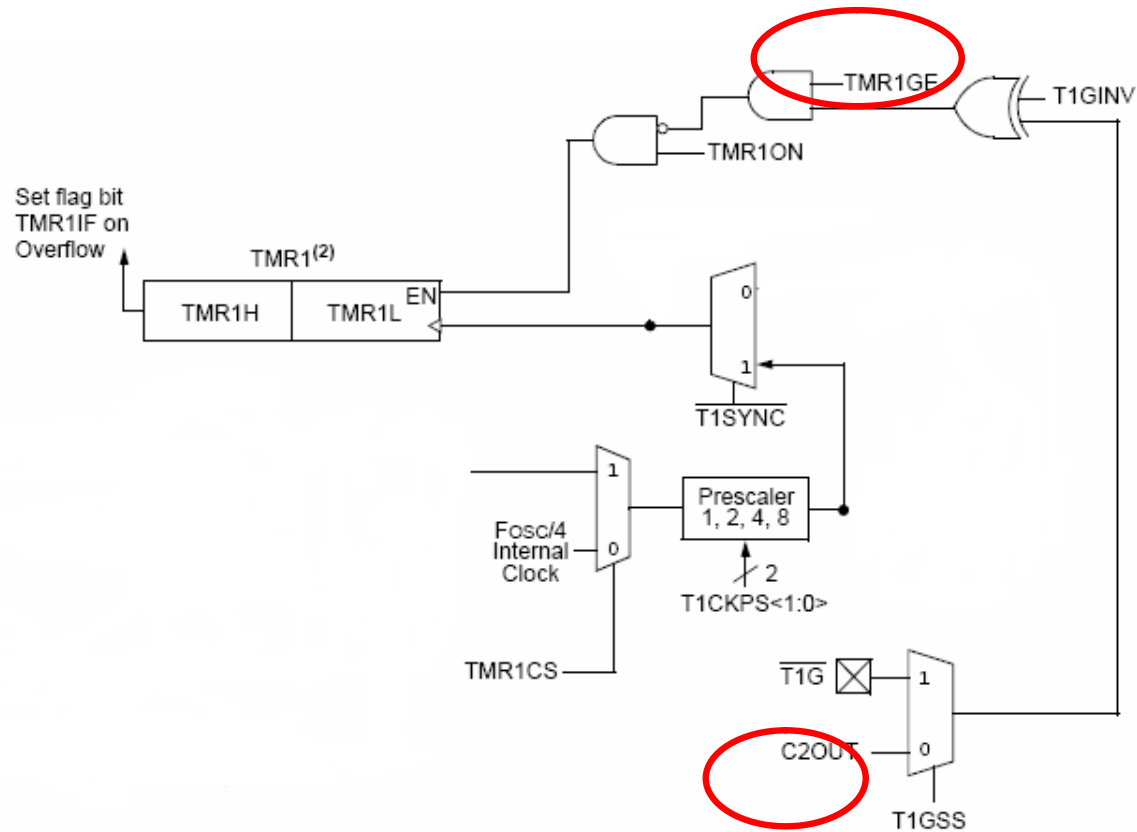
ССР модуль

- Модуль Захвата для аналоговых приложений



Аналоговые компоненты

TIMER 1 GATE



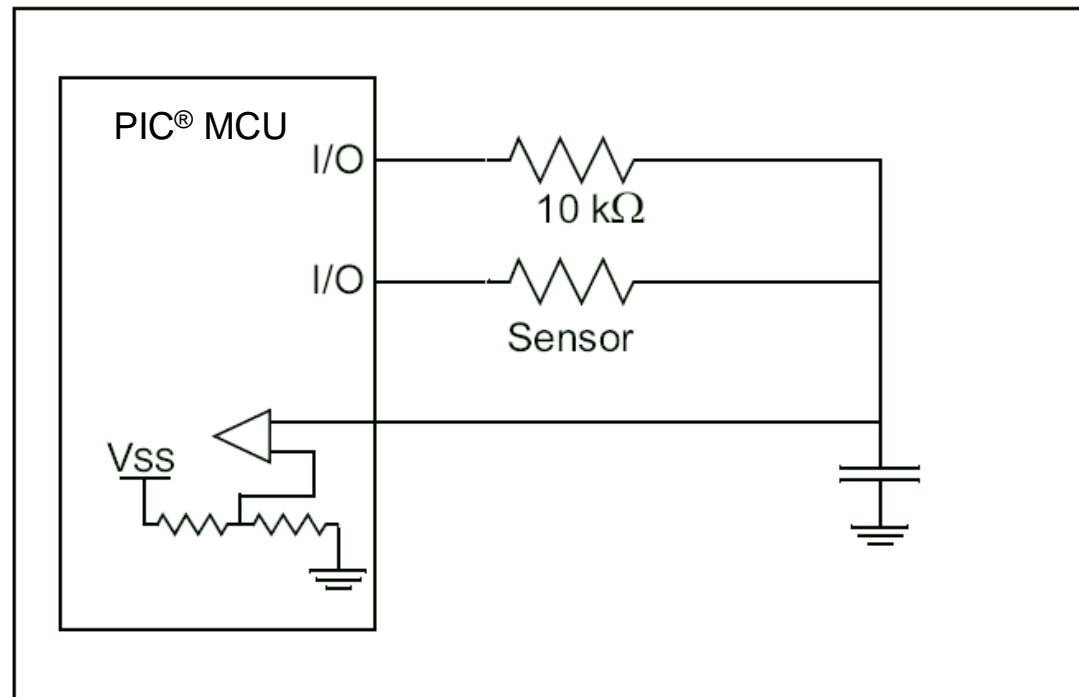
АЦП с использованием периферии PIC[®] микроконтроллеров

Преобразование Аналога в Цифру

- | Традиционно используется модуль АЦП
- | Выход датчика → Напряжение → Усилитель → АЦП
(преобразующий в 8-, 10-, 12 бит значение)

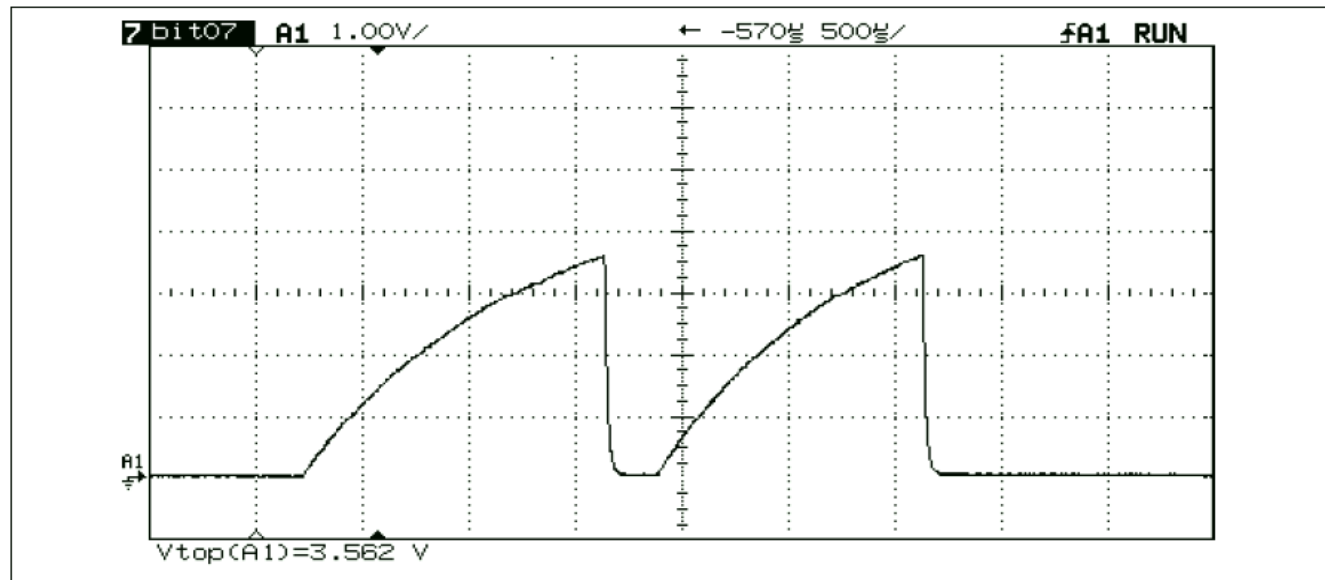
АЦП на базе компаратора

Интегрирующий АЦП:



Преобразование Аналога в Цифру

Интегрирующий АЦП:



Преобразование Аналога в Цифру

Интегрирующий АЦП :

```

// Step 1
TRISC0 = 1;      // Set inputs as high
                  // impedance

TRISC1 = 1;
PORTC = 0;

// Step 2
TRISA1 = 0;      // Discharge Cap.
RA1 = 0;
DelayMs(1);

// Step 3
TRISA1 = 1;      // Cap discharged.
TRISC0 = 0;      // Unknown resistance pin
RC0 = 1;
TMR0 = 0;        // Reset Timer cap begins
to               // charge

// Step 4
while(C1OUT);   // Wait for comparator to
trip

Data[1]= TMR0;

```

```

// Step 5
TRISC0 = 1;      // Set inputs as high
                  // impedance

TRISC1 = 1;
PORTC = 0;

// Step 6
TRISA1 = 0;      // Discharge Cap.
RA1 = 0;
DelayMs(1);

// Step 7
TRISA1 = 1;      // Cap discharged.
TRISC1 = 0;      // Ref resistance pin
RC1 = 1;
TMR0 = 0;

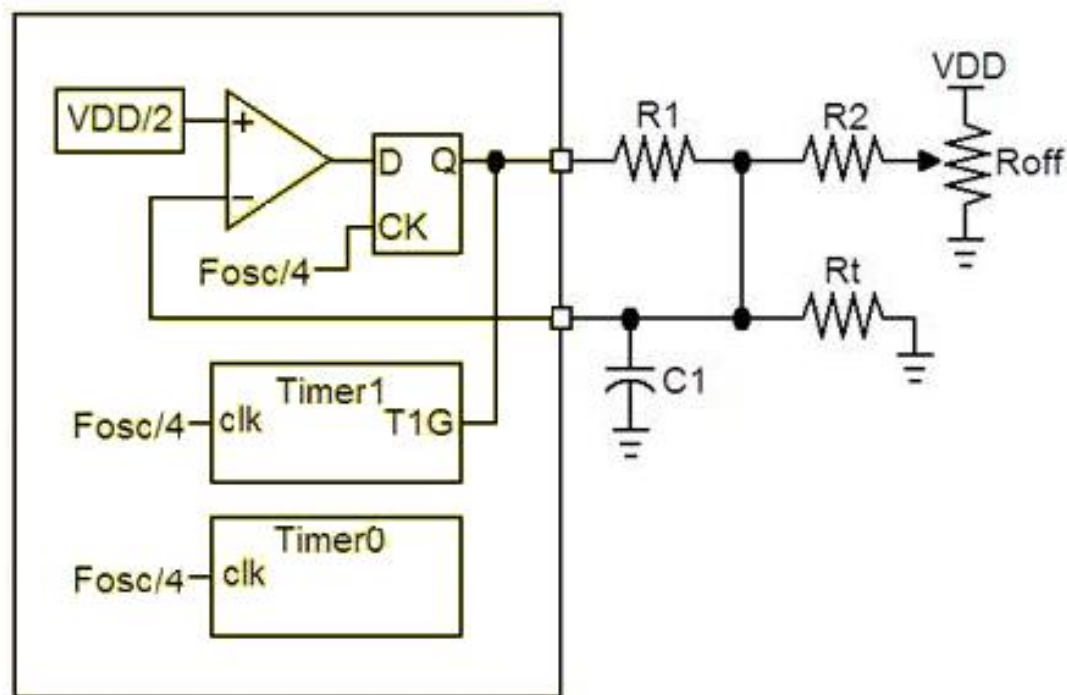
// Step 8
while(C1OUT);   // Wait for comparator to trip

Data [2]= TMR0;
// To calculate the unknown resistance multiply
// Reference resistance * Step4 time/Step8 time
// = Rref*i2c_readBuff_loc[1]/i2c_readBuff_loc[2]

```

Преобразование Аналога в Цифру

| Sigma – Delta:



Преобразование Аналога в Цифру

- | **Sigma – Delta:**
 - | 16 bit A/D
 - | Gain = $R1/Rt$
 - | $Vref = Vdd/2$
 - | На 8МГц время преобразования = 33мсек.

Преобразование Аналога в Цифру

| Sigma – Delta:

Только несколько строк кода

```
TMR1ON = 0;           // stop timer1
Value   = (TMR1H * 256) + TMR1L; // get the 16 bit
                                           //value

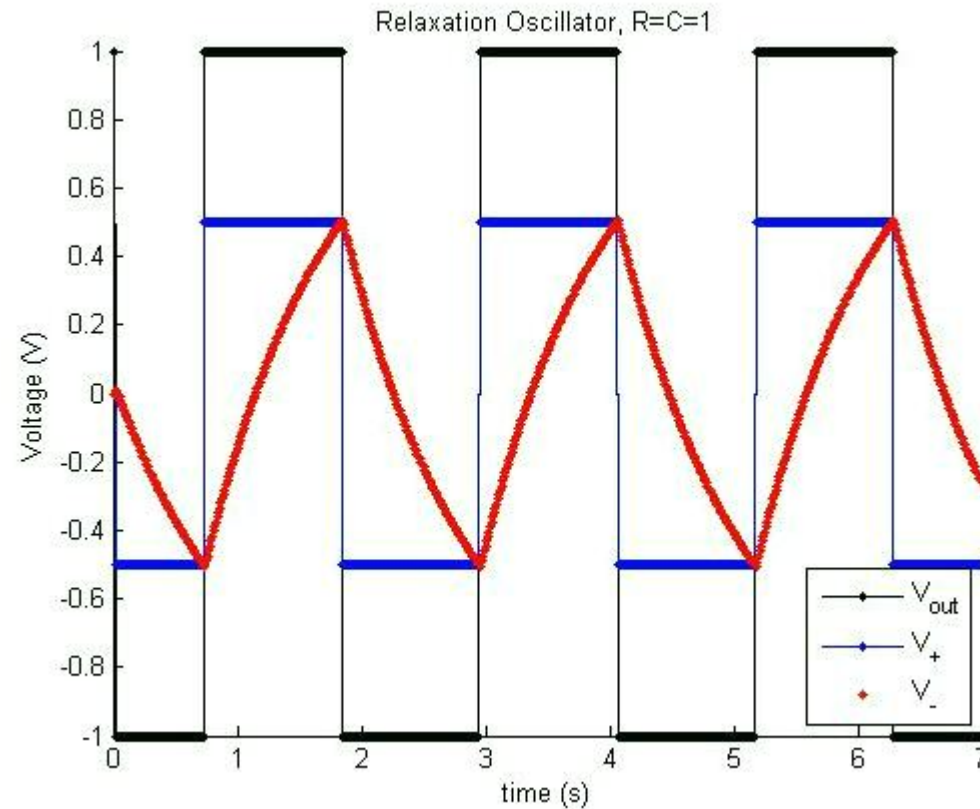
TMR1L   = 0;           // clear timers
TMR1H   = 0;
TMR0    = 0;
TMR1ON  = 1;           // start next conversion
```

Преобразование Аналога в Цифру

- | **Простой способ преобразования значения R, C или L в цифровой сигнал**
- | **Использование релаксационного генератора на основе компаратора**

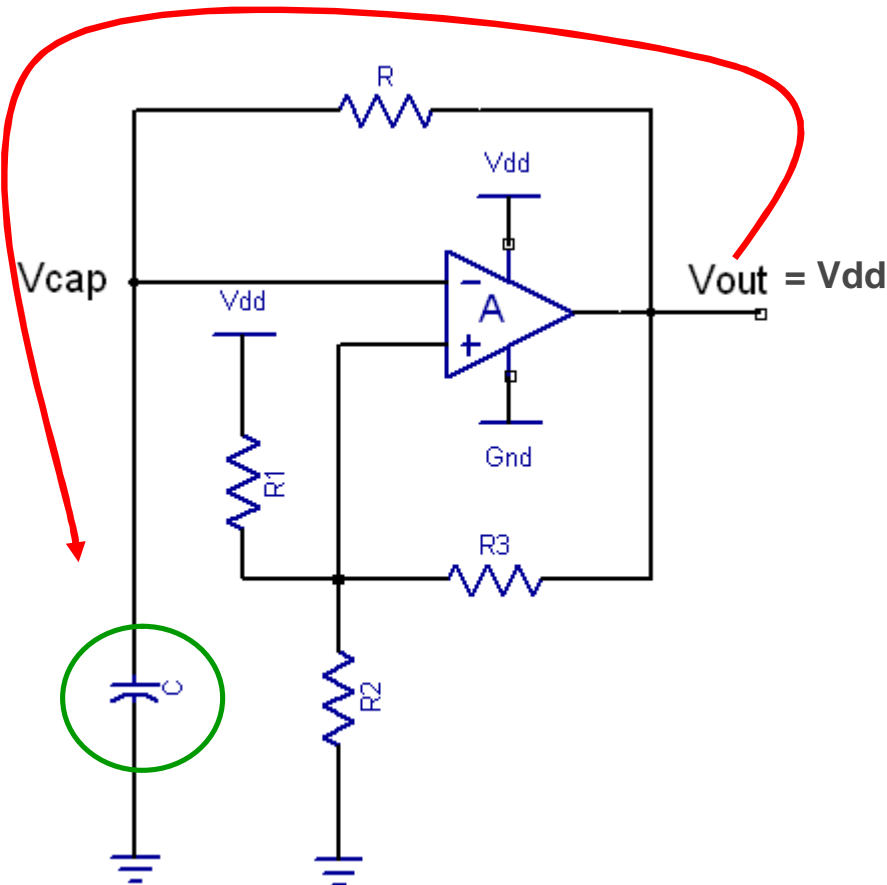
Преобразование Аналога в Цифру

Напряжение в частоту:



Преобразование напряжение-частота

Релаксационный генератор: Основная схема

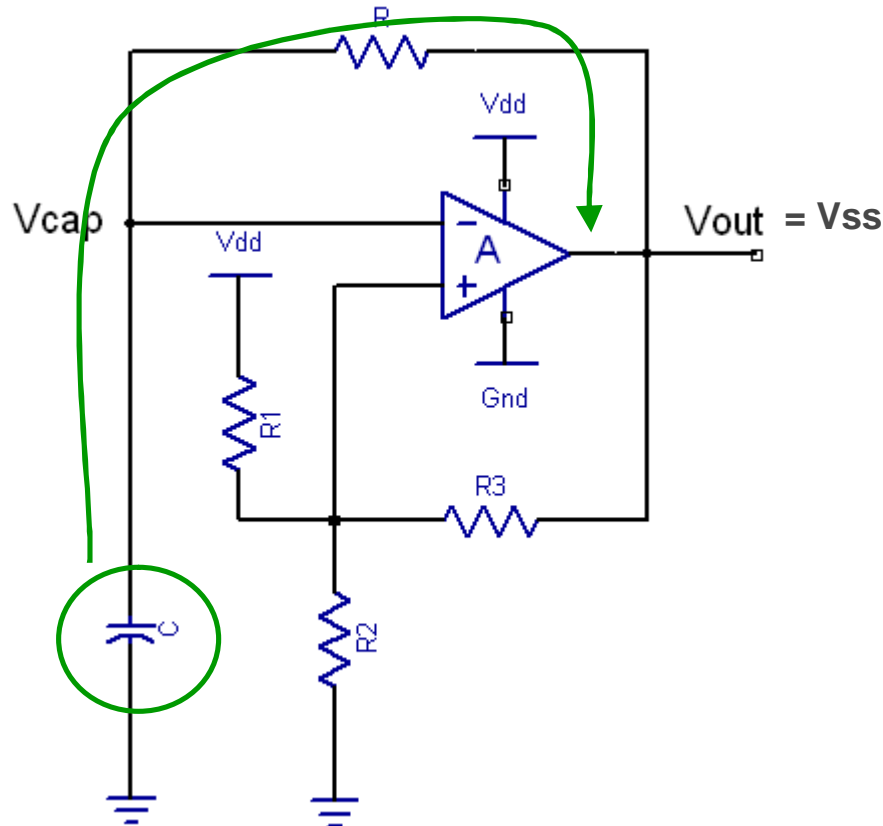


- | Работает на попеременном **заряде** и разряде конденсатора

- | Частота определяется постоянной времени RC-цепочки и гистерезисом

Преобразование напряжение-частота

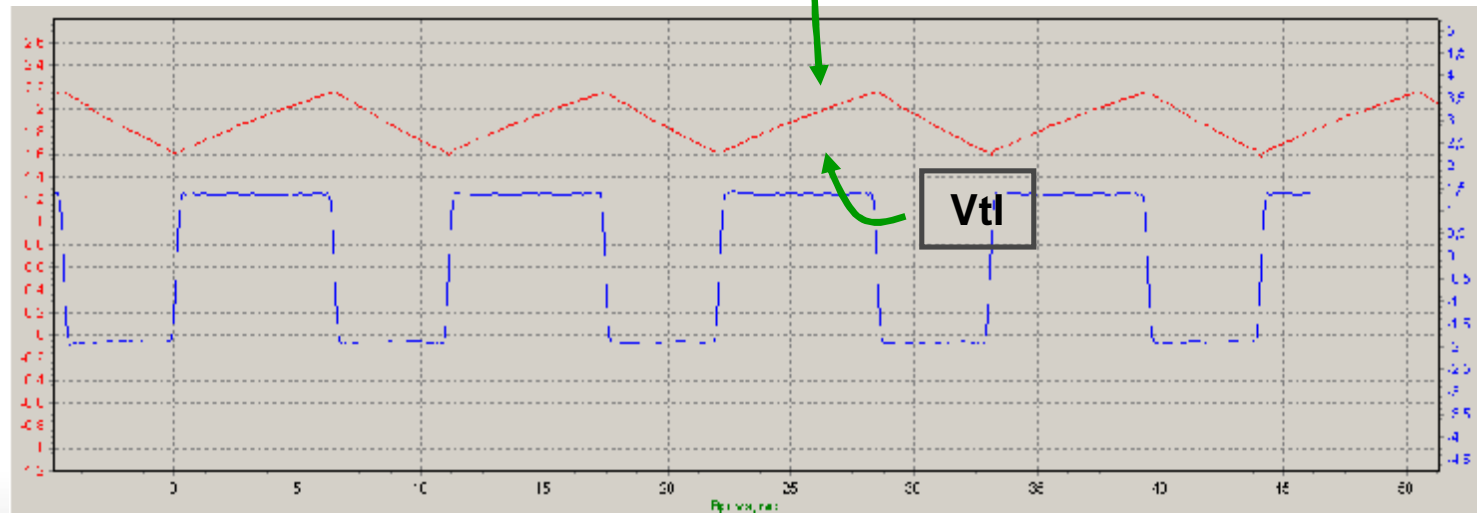
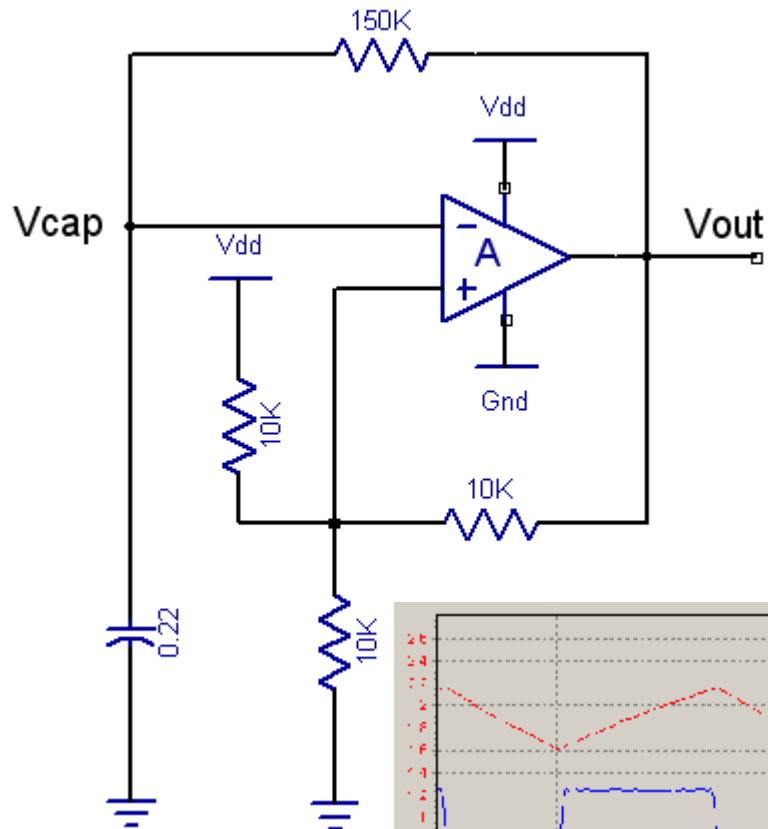
Relaxation Oscillator: Basic Circuit



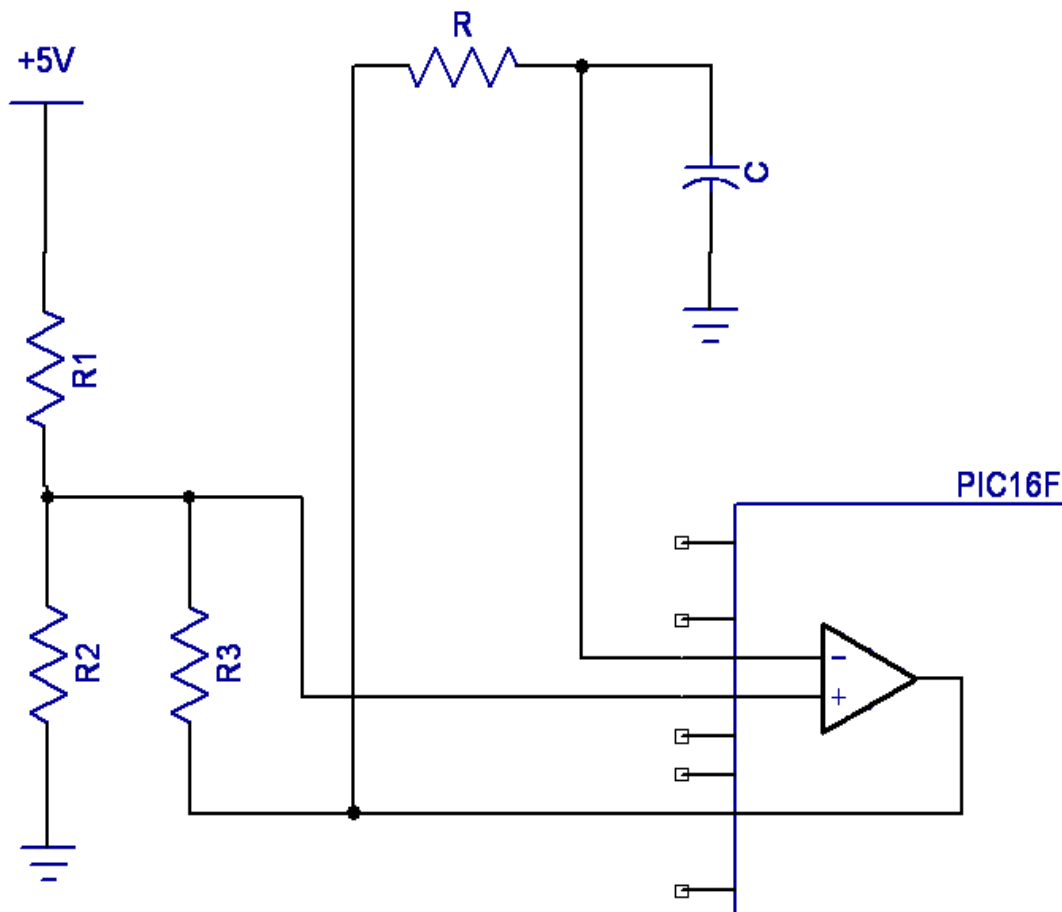
| Работает на попеременном заряде и **разряде** конденсатора

| Частота определяется постоянной времени RC-цепочки и гистерезисом

Релаксационный генератор



Как сделать Релаксационный на PIC®



Компаратор
конфигурируется с
помощью
регистров
CM2CON0 CM2CON1

В новых
контроллерах
вместо делителя
R1 и R2 можно
использовать
внутренний
источник опорного
напряжения

Как сделать Релаксационный на PIC®

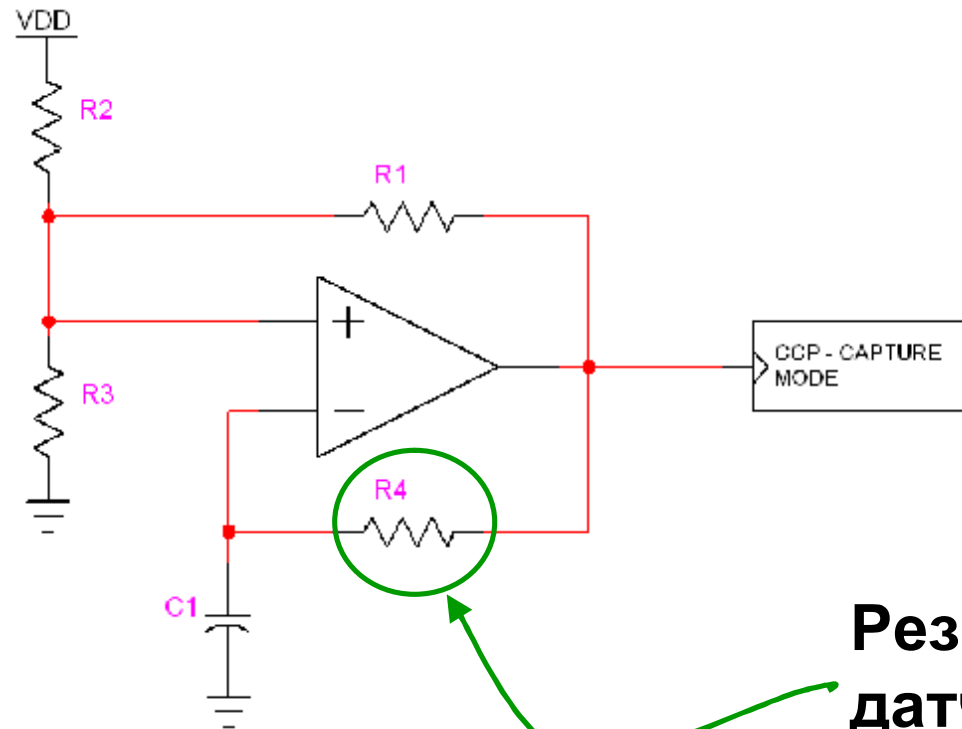
Конфигурирование компаратора C2

```
main()
{
    ANSEL = 0x33;           // ANALOG CHANNELS FOR COMP input
                           // C1IN+, C2IN+, C12IN0-, C12IN1-
    TRISA = 0b00111011;    // RA0, RA1 inputs, RA2 output
    PORTA = 0;
    CM2CON0 = 0xA1;        // Enable C2 C2OUT, C2IN+, C12IN1-

    while(1)
    {
        /*Your code here*/
    }
}
```

Преобразование сигнала датчика в цифру

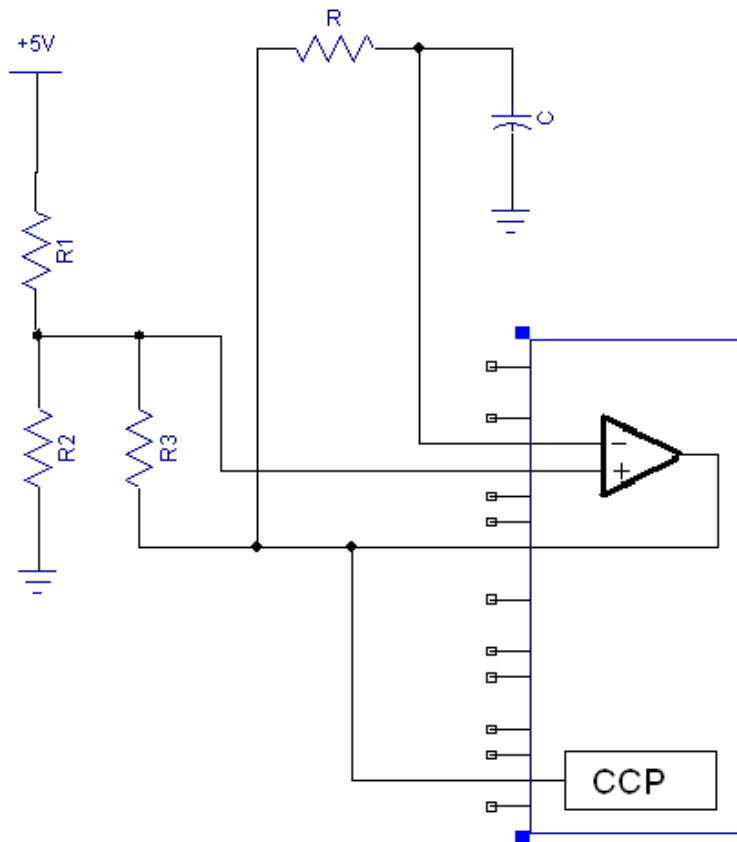
Преобразование сигнала датчика в цифру с помощью релаксационного генератора:



Резистивный датчик

Преобразование сигнала датчика в цифру

Решение на базе PIC16F



Решение:

Использование внутреннего компаратора для преобразования сигнала датчика в частоту.

Завести выход компаратора на вход захвата для измерения периода.

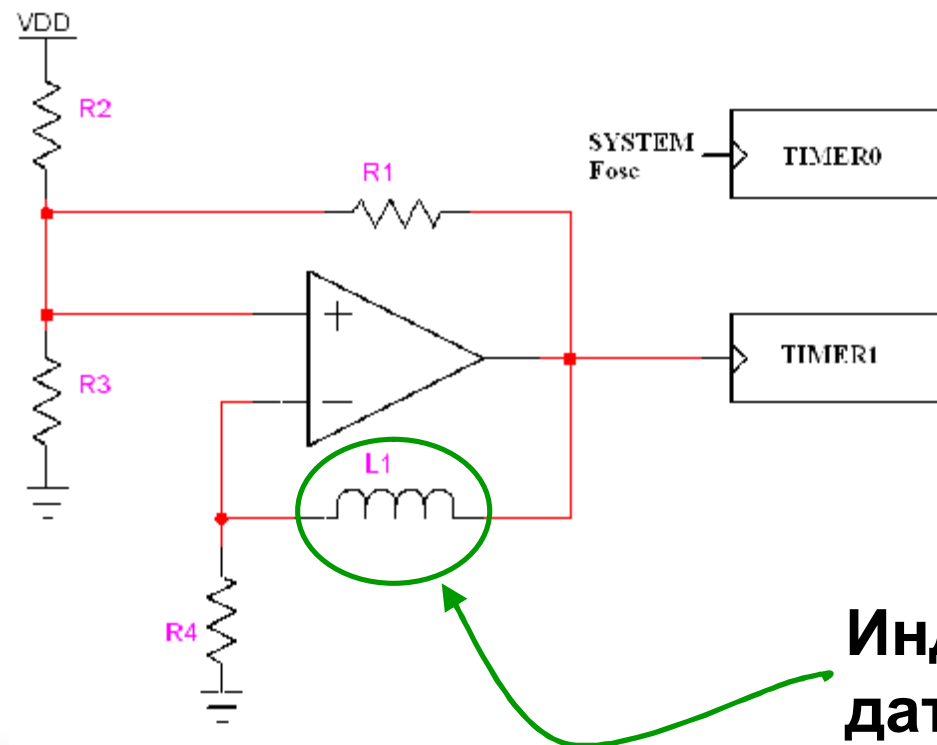
Преобразование сигнала датчика в цифру

Используя модуль ССР



Преобразование сигнала датчика в цифру

Преобразование сигнала датчика в цифру с помощью релаксационного генератора:



Преобразование сигнала датчика в цифру

Используя
Timer1



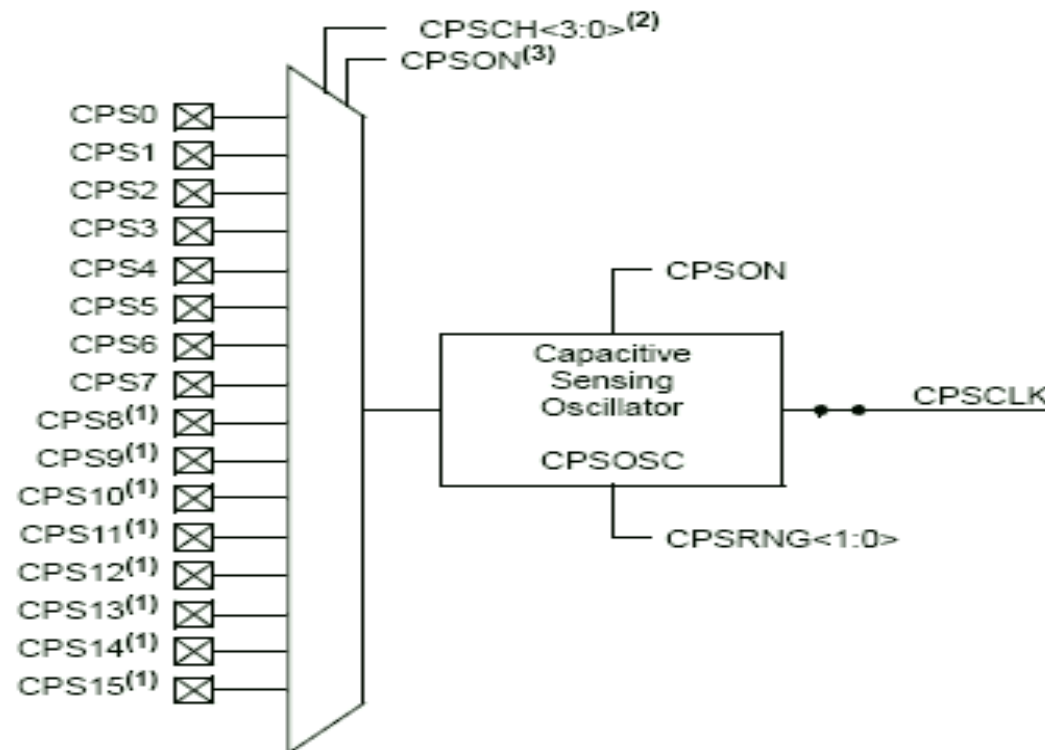


Модуль CSM – Capacitive Sense Module

- | **Использует схему генератора**
- | **С источником постоянного тока**
- | **Доступен в PIC16F72X**
- | **Не нужны дополнительные
внешние элементы**

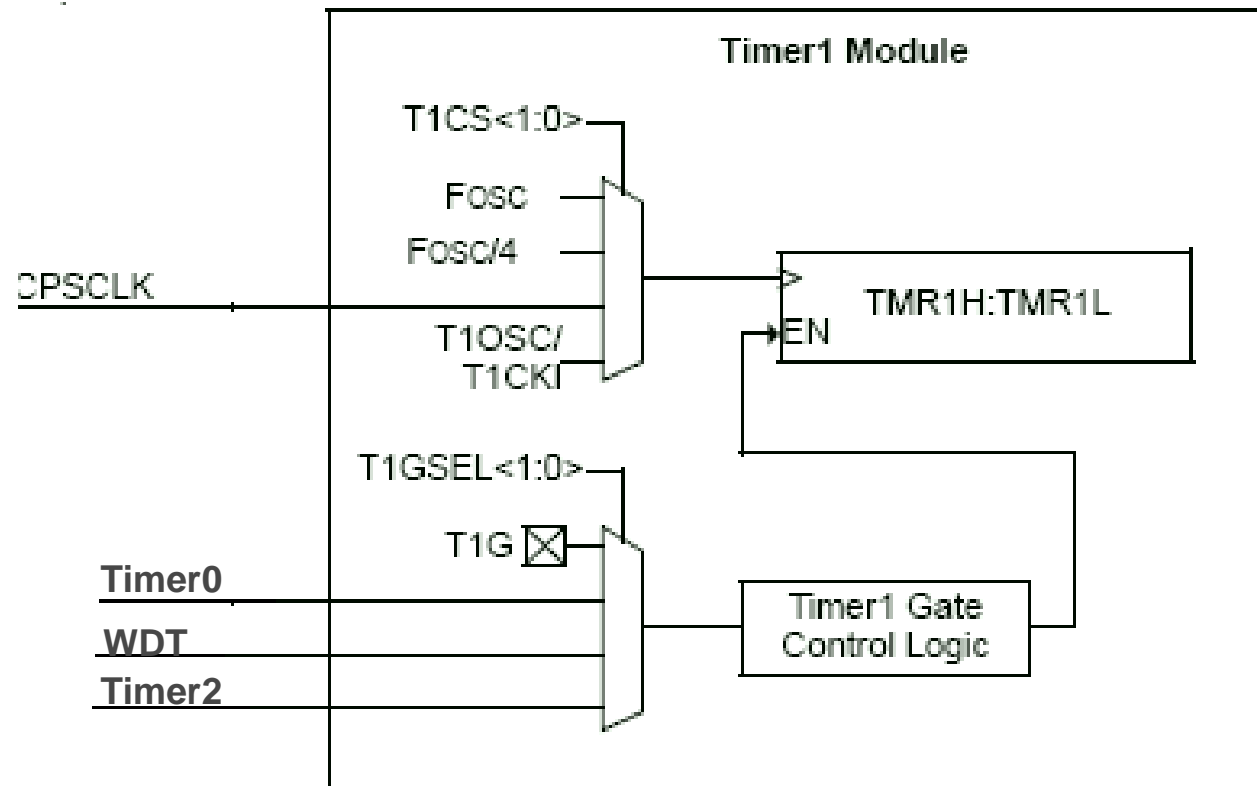
Capacitive Sense Module

Схема:



Capacitive Sense Module

Измерение частоты:



Capacitive Sense Module

Временная база от Timer0:

- ТМR0 становится источником разрешения счета Timer1

```
OPTION = 0xC3;      // fosc/4, hi-lo edge transition, 1:16 prescaler
TMR0IF = 0;        // clear TMR0 interrupt flag
TMR0IE = 1;        // enable TMR0 interrupt
T1CON  = 0xC5;     // Timer1 initialization
T1GCON = 0xE1;     // Timer1 gate init /Toggle Mode/TMR0 time base
TMR1GIF = 0;      // Clear Gate Interrupt Flag
TMR1GIE = 1;      // Enable Gate Interrupt
```

Capacitive Sense Module

Временная база от Timer2:

- ТМR2 источник разрешения счета
Timer1

```
T2CON = 0x04; // T2ON, prescaler & postscaler = 1:1
T2CON = 0x01; // adjust prescaler
PR2 = 0xB4; // w/pres.1:1, 0xB4 sets 125us scan rate.
TMR2IF = 0x00;
TMR2IE = 0x01;
T1CON = 0x45; // Timer1 init
T1GCON = 0xE2; // Timer1 gate init/ Toggle Mode
// set T1GSS for Timer2 match PR2
TMR1GIF = 0; // Clear Gate Interrupt Flag
TMR1GIE = 1; // Enable Gate Interrupt
```

Capacitive Sense Module

- | **Временная база от WDT:**
 - | Watchdog источник разрешения счета Timer1
 - | Используется для работы емкостного сенсора в режиме SLEEP

°CW

```
OPTION = 0xCB;           // Prescaler assigned to WDT
T1CON  = 0x45;          // Timer1 init
T1GCON = 0xE3;          // Timer1 Gate Enabled / WDT time base
TMR1GIF = 0;            // clear gate Interrupt flag
TMR1GIE = 1;            // enable gate interrupt
```

Capacitive Sense Module

Временная база от WDT в режиме Sleep:

```
CLRWDT ();           // Reset Watchdog
SLEEP ();            //Enter Sleep mode
                    //The first wakeup Gate is enabled and timer1 will
                    //start counting.

SLEEP ();

                    //Second wakeup, Gate is disabled, and timer1 gate
                    //will interrupt.
```

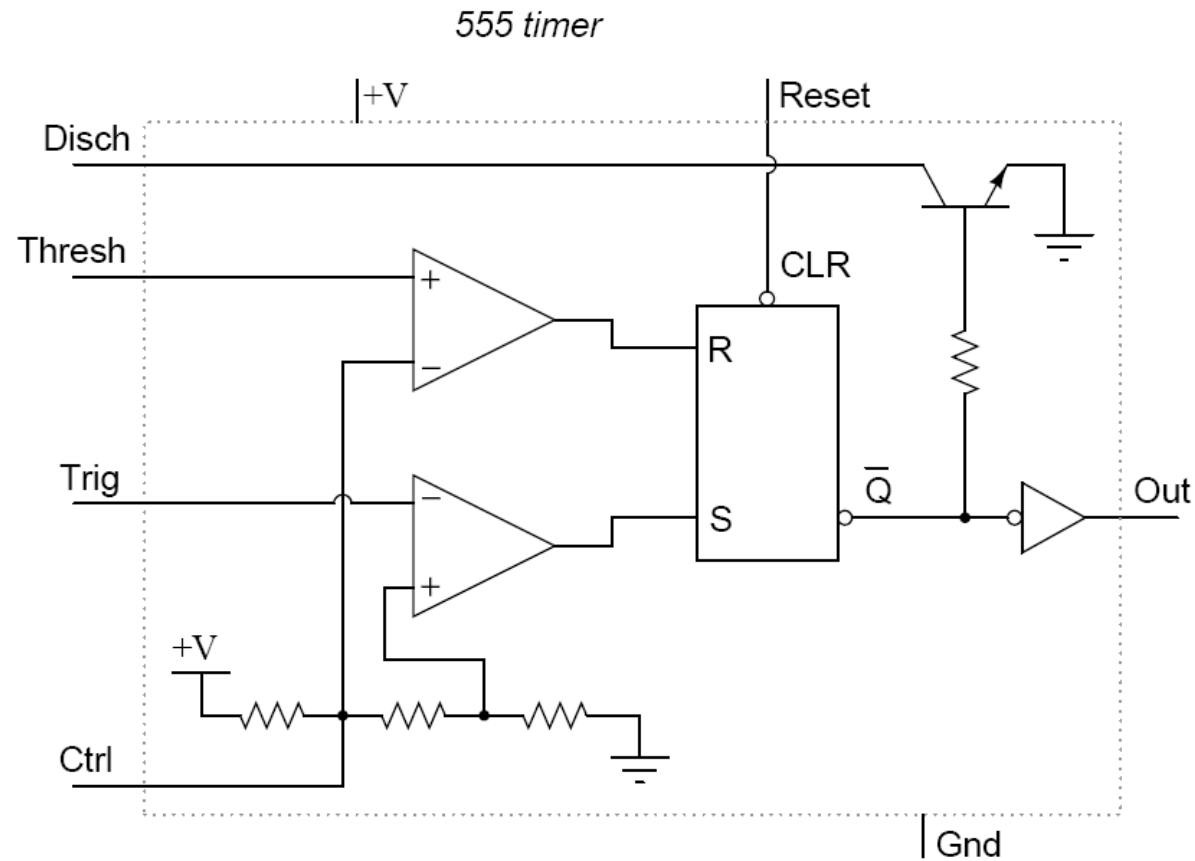


Другие применения аналоговой периферии

Таймер 555 как периферия контроллеров PIC[®]

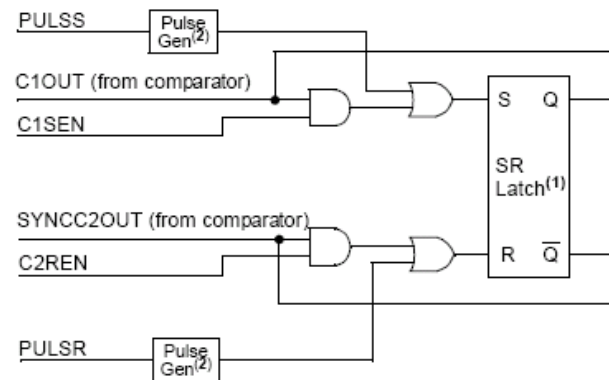
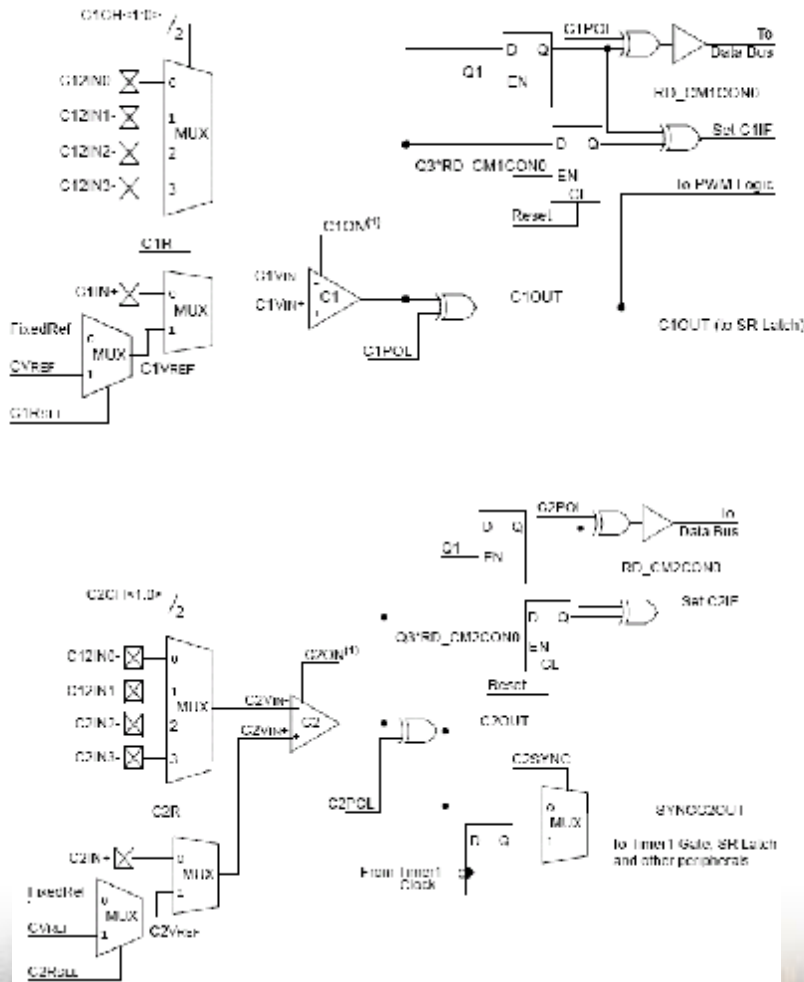
Таймер 555 - периферия PIC

Что такое 555-й Таймер?



Таймер 555 - периферия PIC

Что же мы имеем внутри PIC-а?



Таймер 555 - периферия PIC

| **...555 таймер может использоваться для построения**

| Генераторов

| Одновибраторов

| Генераторов задержки

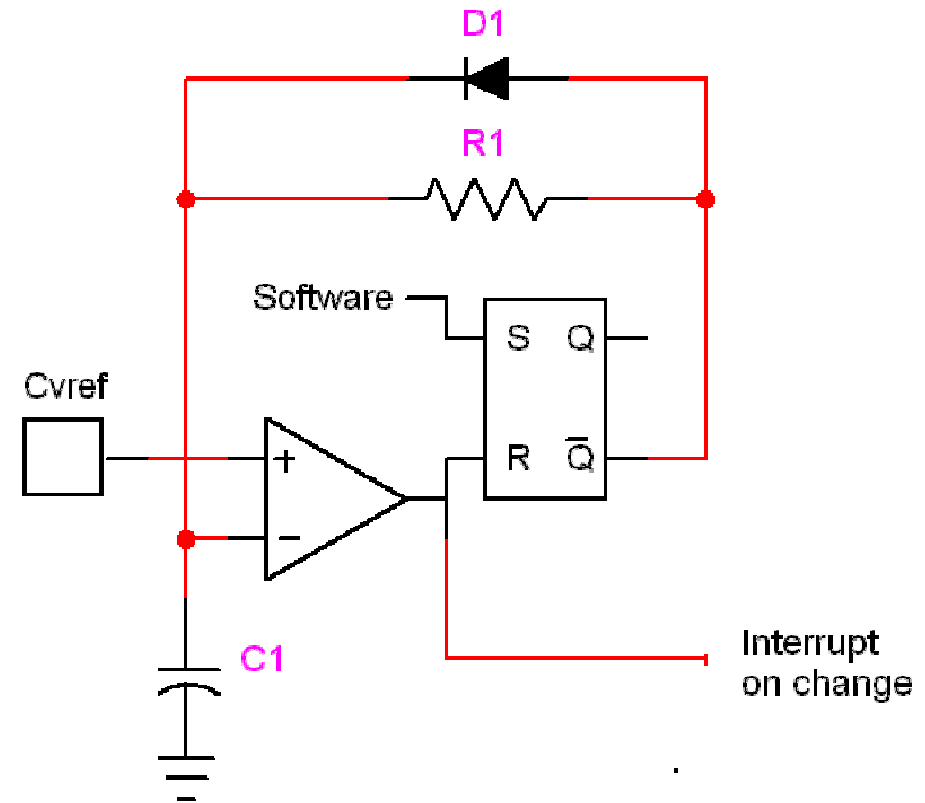
И вы можете применять это для всевозможных задач...

Таймер 555 - периферия PIC

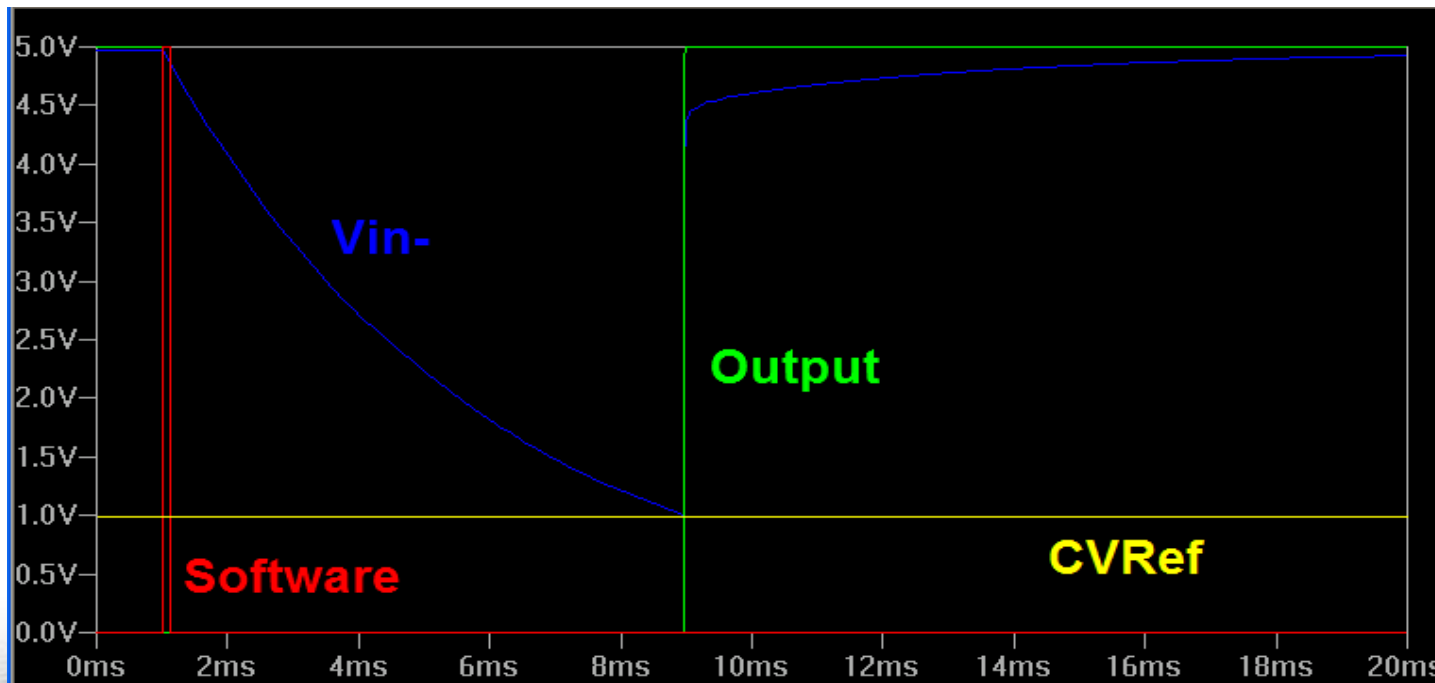
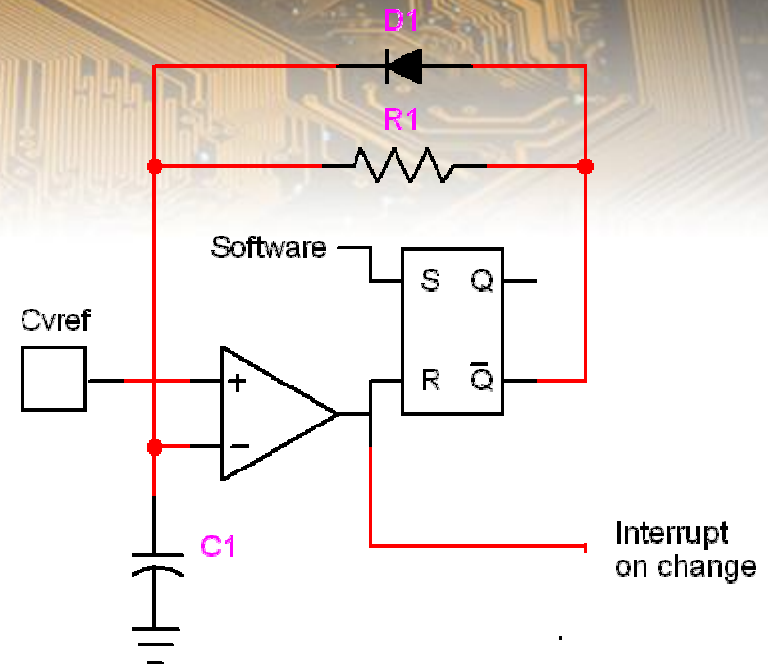
One Shot Timer (Моностабильный мультивибратор)

Короткая или длинная задержка One-Shot Timer

- | Программный триггер
- | Короткая или длинная задержка
- | Переменное сопротивление R для настройки
- | Используйте прерывание по изменению состояния для выхода из sleep

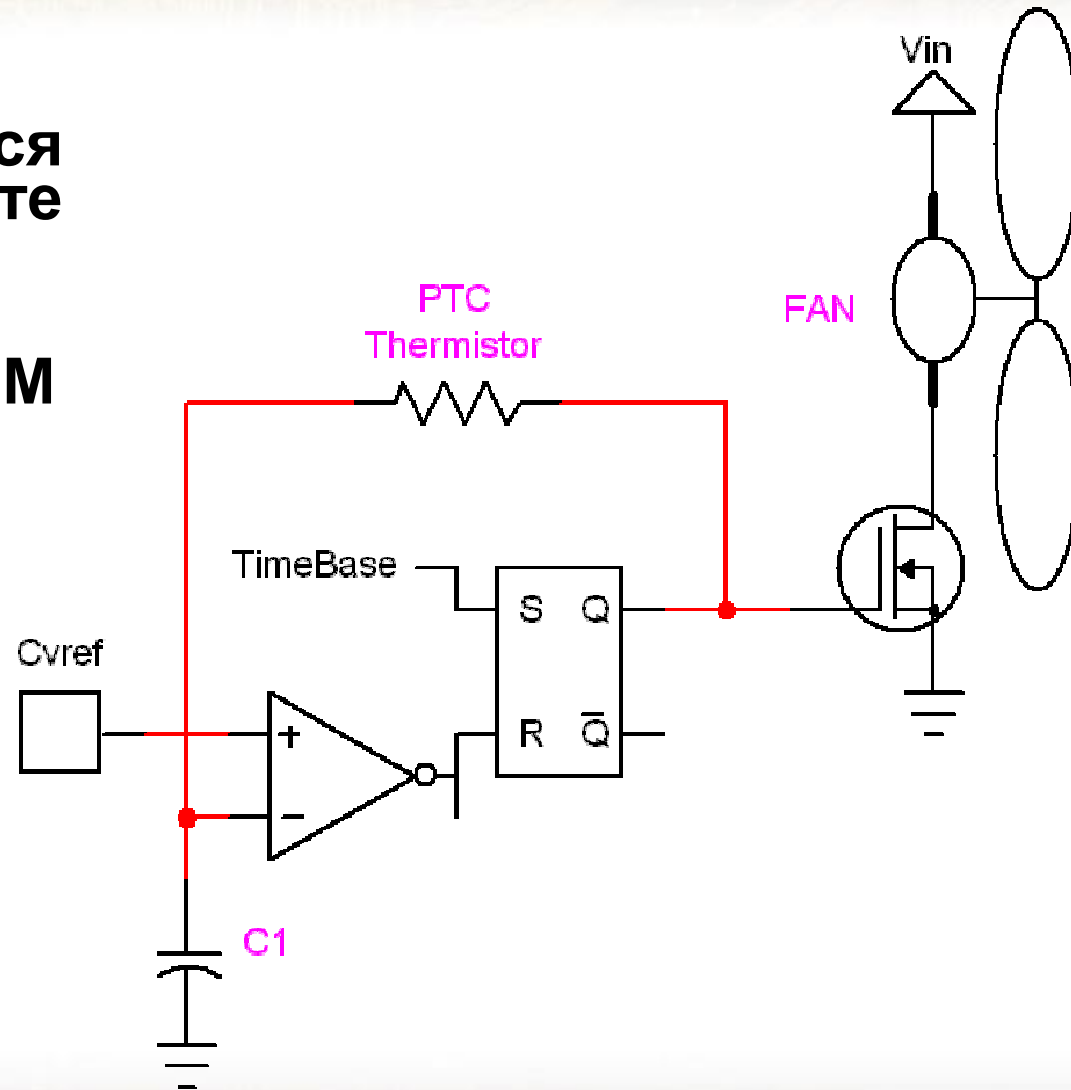


One-Shot



Управление вентилятором

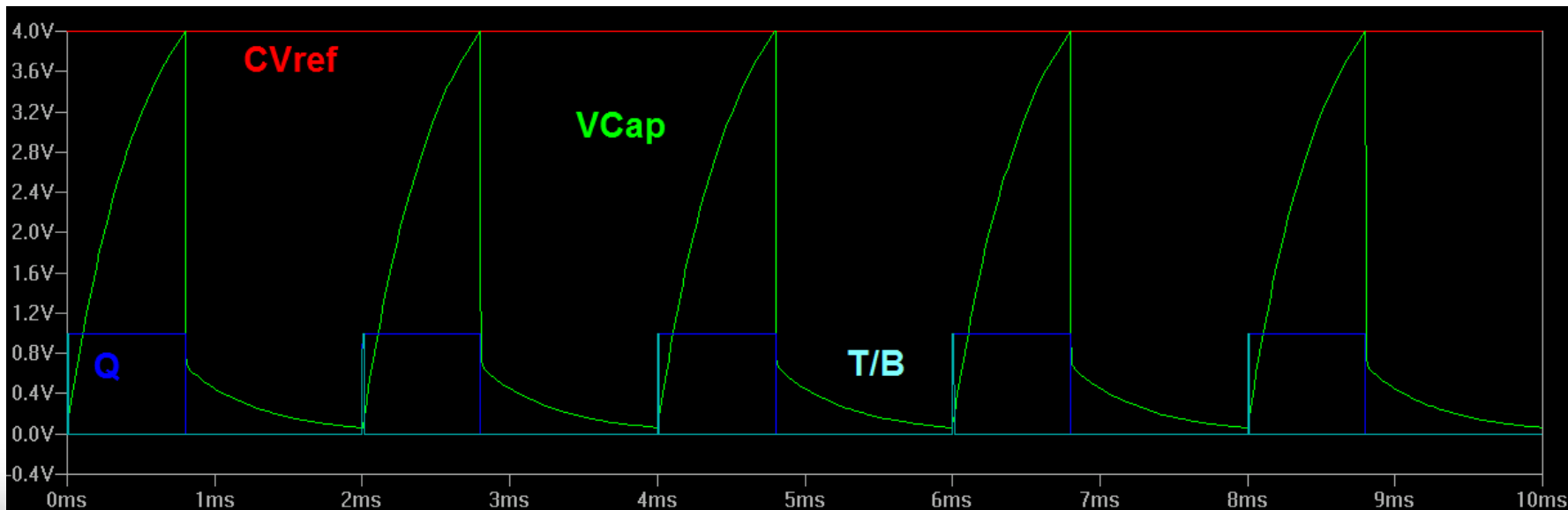
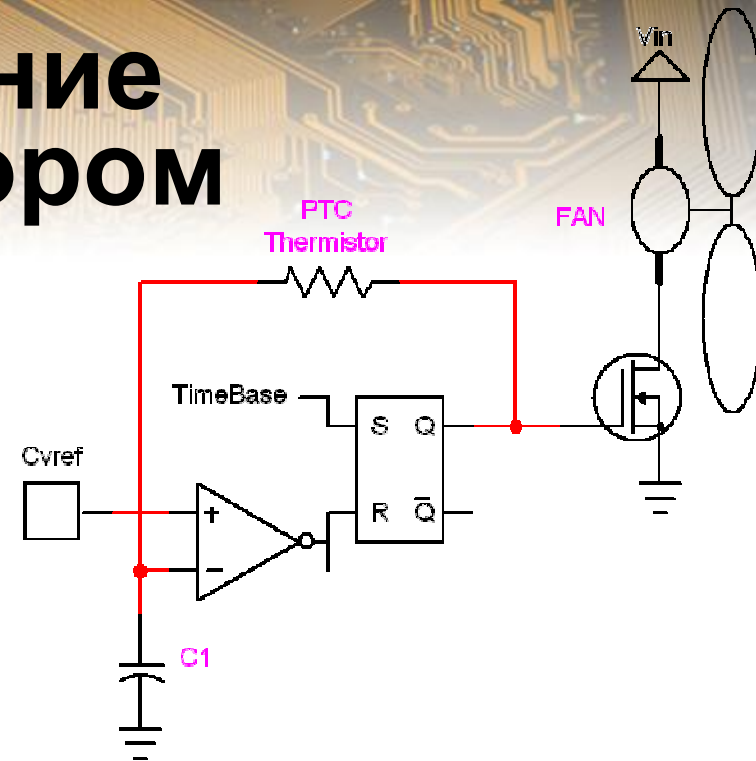
- | Вентилятор должен вращаться быстрее при росте температуры
- | Обычно применяется ШИМ со схемой обратной связи.



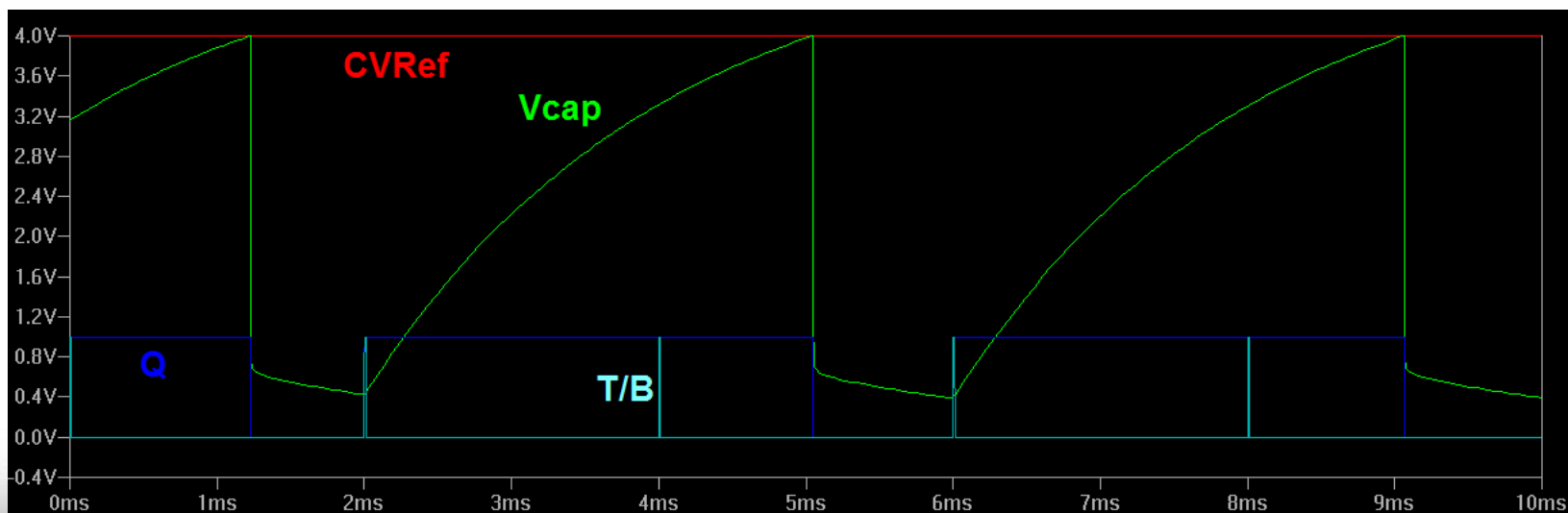
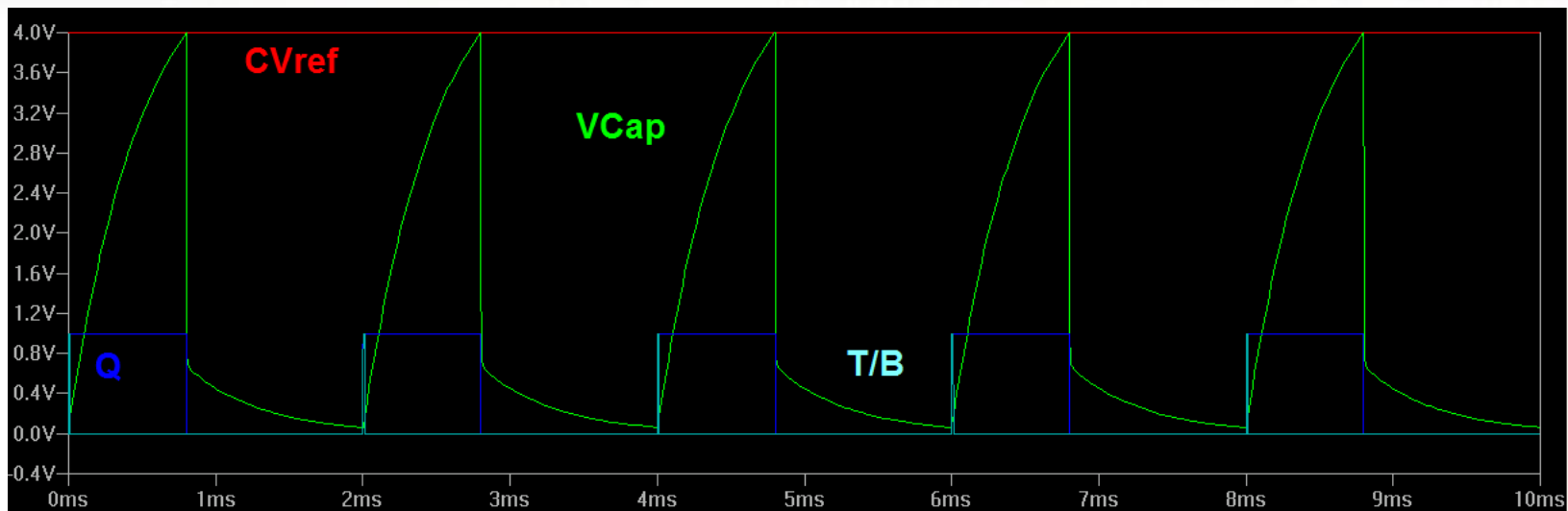
Управление вентилятором

Большое сопротивление ->
медленнее заряд -> Шире
импульсы ШИМ

Можно добавить диод для быстрого
разряда конденсатора – дает
более постоянную ширину
импульсов

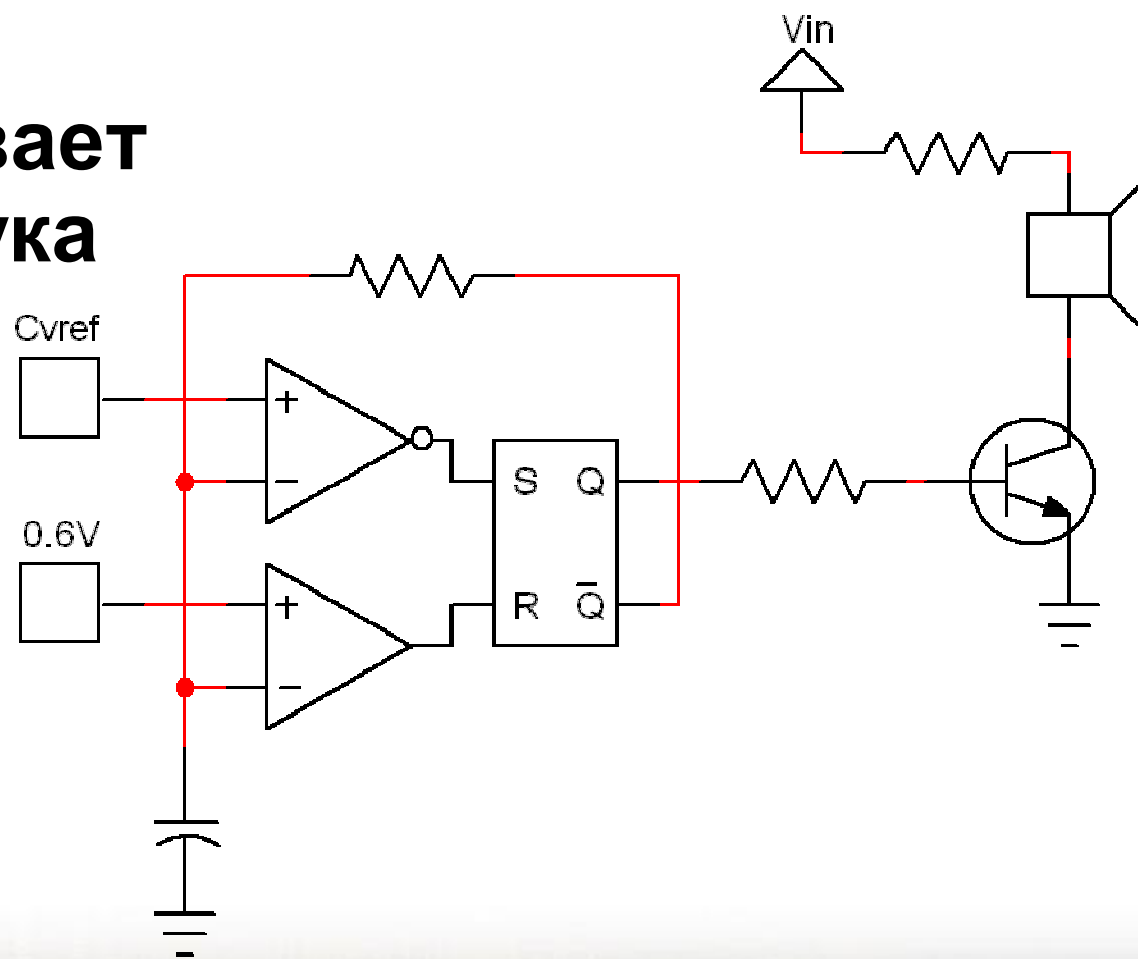


Управление вентилятором – малое или большое R

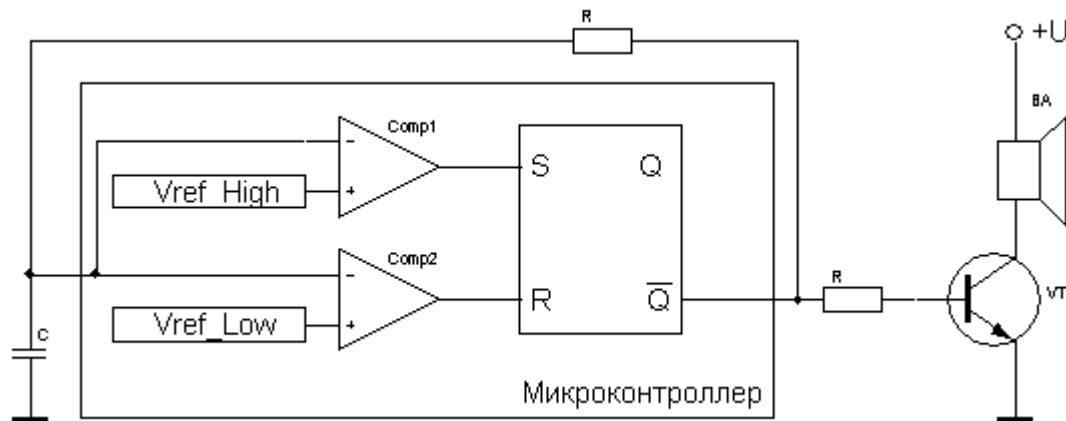


Программируемый генератор

| **Cvref**
устанавливает
частоту звука

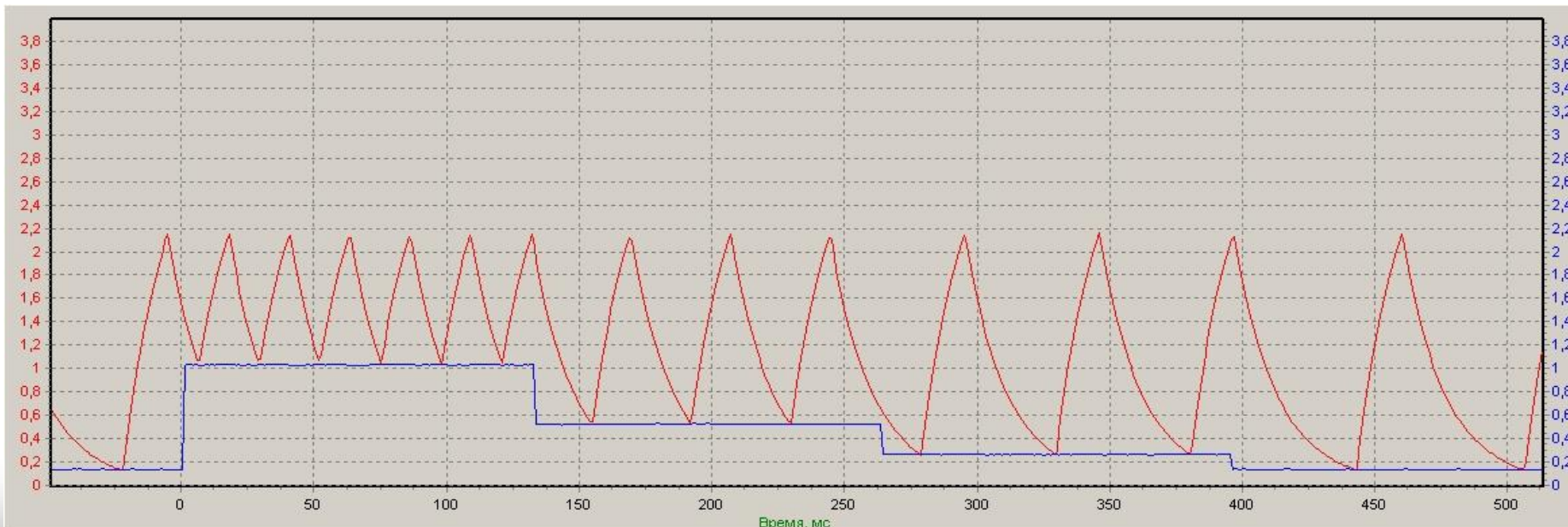


Программируемый генератор



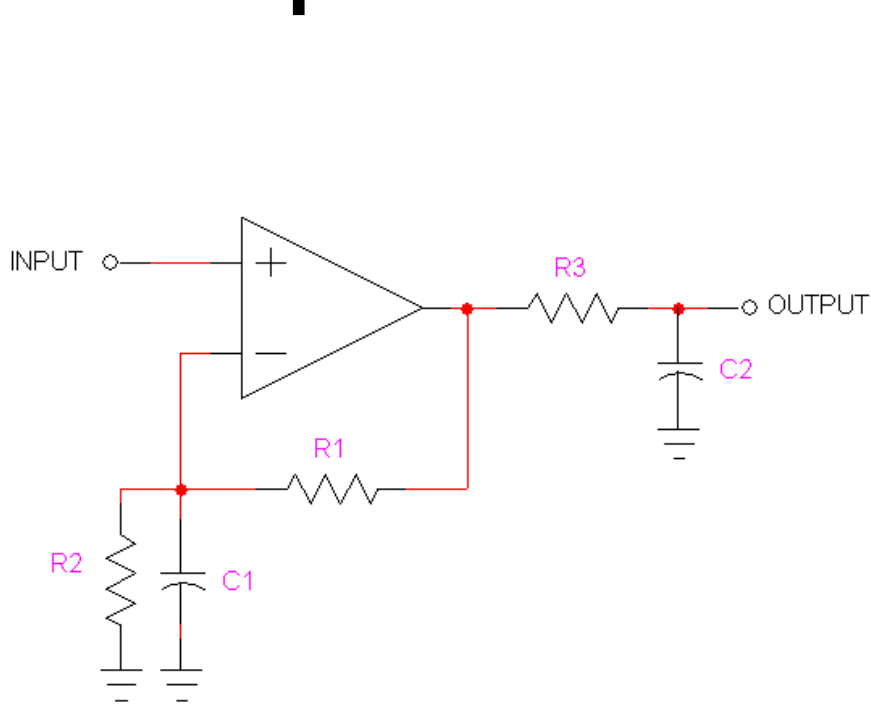
CVRef устанавливает частоту.

Помните, вы можете изменять напряжение CVRef программно (это ЦАП!)

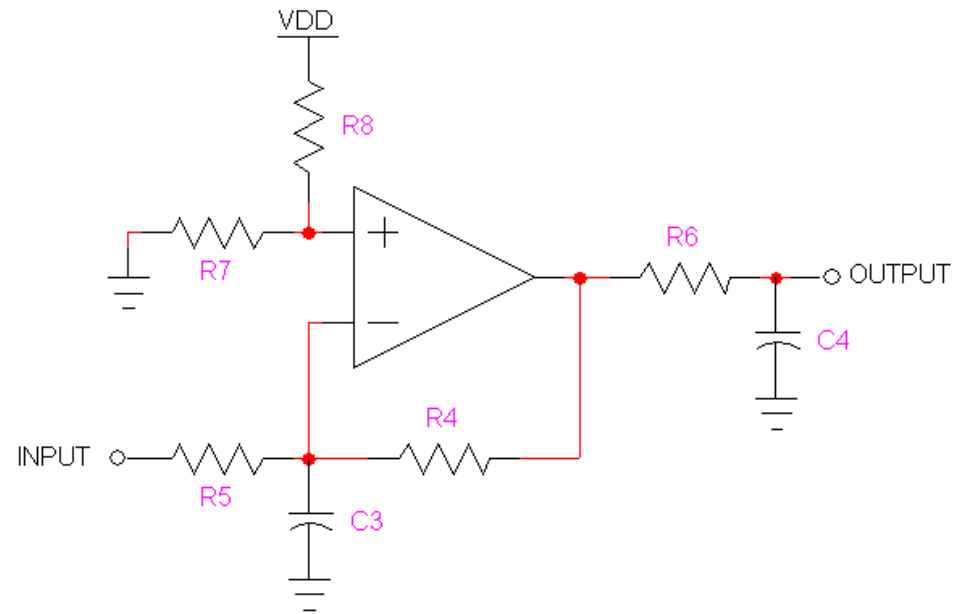


Компаратор как Операционный усилитель

Это работает!



Не инвертирующий



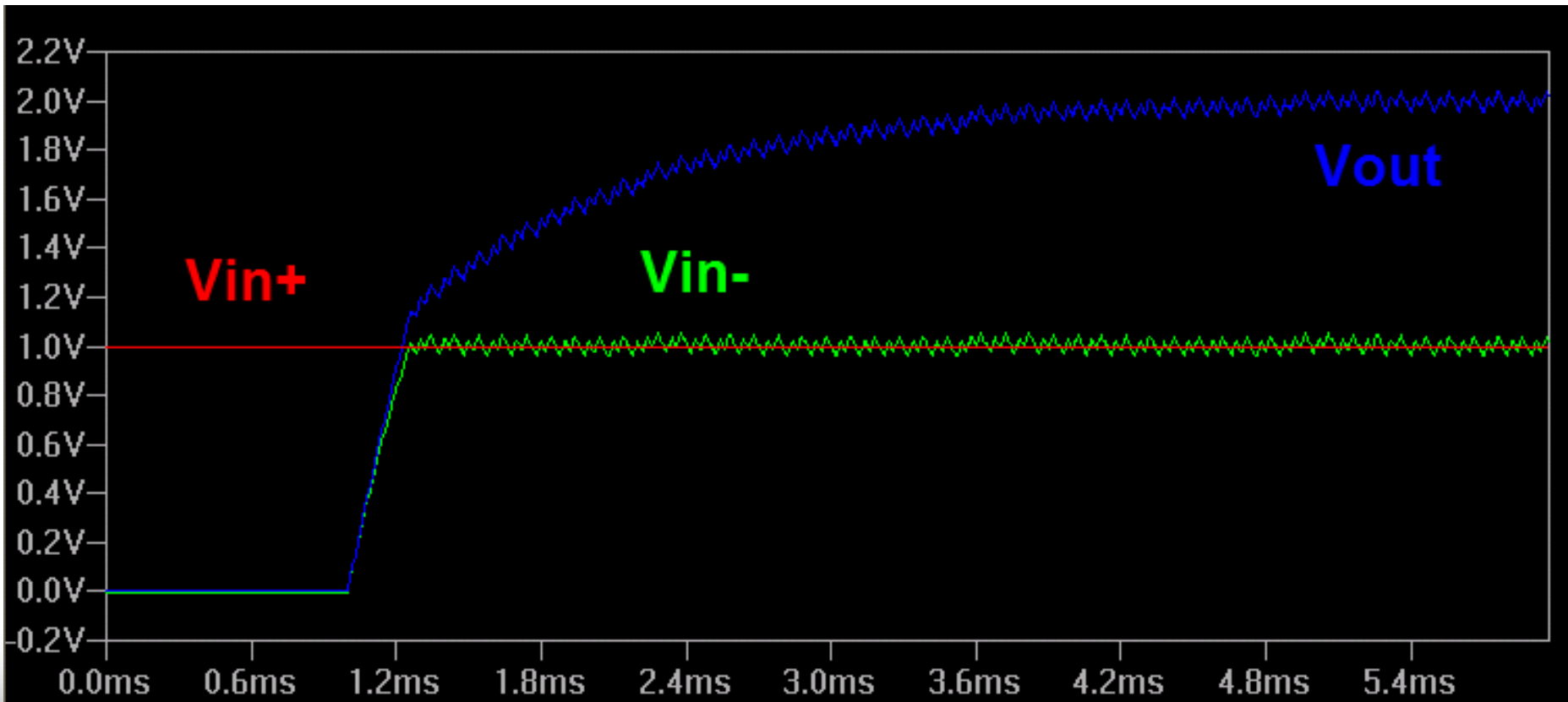
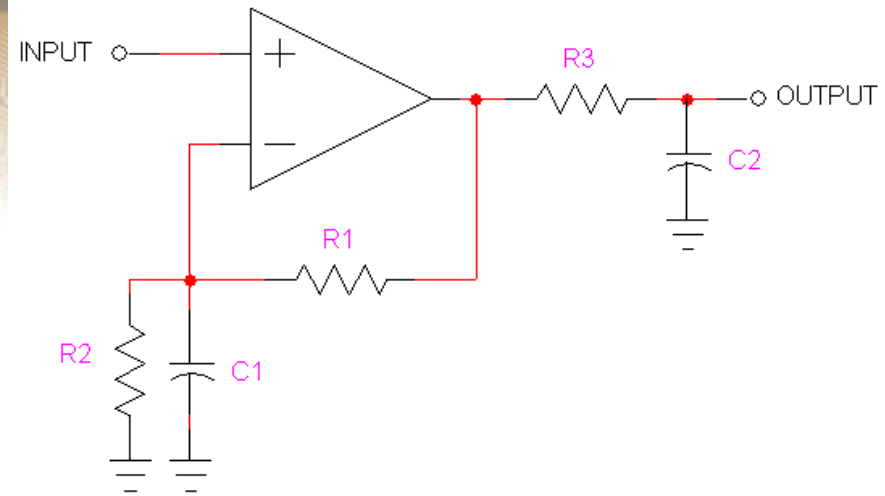
Инвертирующий



Компаратор - ОУ

$$\text{Gain} = (R2+R1)/R2$$

Нужен фильтр на выходе компаратора для уменьшения шума.



- | **Сегодня мы рассмотрели:**
 - | Характеристики АЦП
 - | Различные схемы АЦП, которые могут быть реализованы на модулях, присутствующих в PIC[®] микроконтроллерах
 - | Другие схемы, основанные на этих модулях, которые могут пригодиться в ваших разработках



Вопросы??...





Дополнительная литература

PIC[®] MCU Compiled Tips 'n Tricks Guide

**Using the PIC[®] MCU 555 timer peripheral
(Presentation by Keith Curtis)**

PIC16F88x, PIC16F690, PIC16F72X Datasheet

**AN1171 Using the Cap Sense module on the
PIC16F72X**

**AN929 Temperature Measurement Circuits for
Embedded Applications**

AN863 A Comparator Based Slope ADC

**Real Analog Solutions for Digital Designers by
Bonnie Baker**



Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, KeeLoq, KeeLoq logo, MPLAB, PIC, PICmicro, PICSTART, rfPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, ICEPIC, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, nanoWatt XLP, Omniscient Code Generation, PICC, PICC-18, PICkit, PICDEM, PICDEM.net, PICtail, PIC32 logo, REAL ICE, rfLAB, Select Mode, Total Endurance, TSHARC, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2009, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.