

Регистры

Регистры входят в категорию ПЛС и предназначены для хранения и обработки двоичных слов.

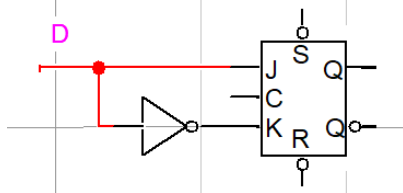
Основные компоненты любого регистра - это триггеры. Регистры также содержат комбинационные схемы для управления триггерами.

Длина регистров определяется количеством триггеров, из которых они состоят.

Регистр с n триггерами может хранить n -битное двоичное слово.

Регистры состоят из триггеров типа D или триггеров JK, RS, которые работают в режиме триггеров D.

Например, для работы триггера JK или RS в режиме триггера D достаточно соединить входы J и K (R и S) через инвертор.



В зависимости от выполняемых функций регистры делятся на следующие категории:

1. **Параллельные регистры.** Информация должна записываться в эти регистры и считываться параллельно.

2. **Последовательные регистры или регистры сдвига.** В этих регистрах информация вводится и считывается последовательно, бит за битом. Эти регистры могут сдвигать двоичные слова влево или вправо.

3. **Параллельно-последовательные** или **последовательно-параллельные** регистры, в зависимости от того, как считывается или записывается двоичное слово.

Существуют следующие основные типы регистров:

- **SISO** (Serial Input - Serial Output, последовательный вход - последовательный выход); - со сдвигом информации вправо, со сдвигом информации влево, двунаправленный

- **SIPO** (Serial Input - Parallel Output, последовательный ввод - параллельный вывод);

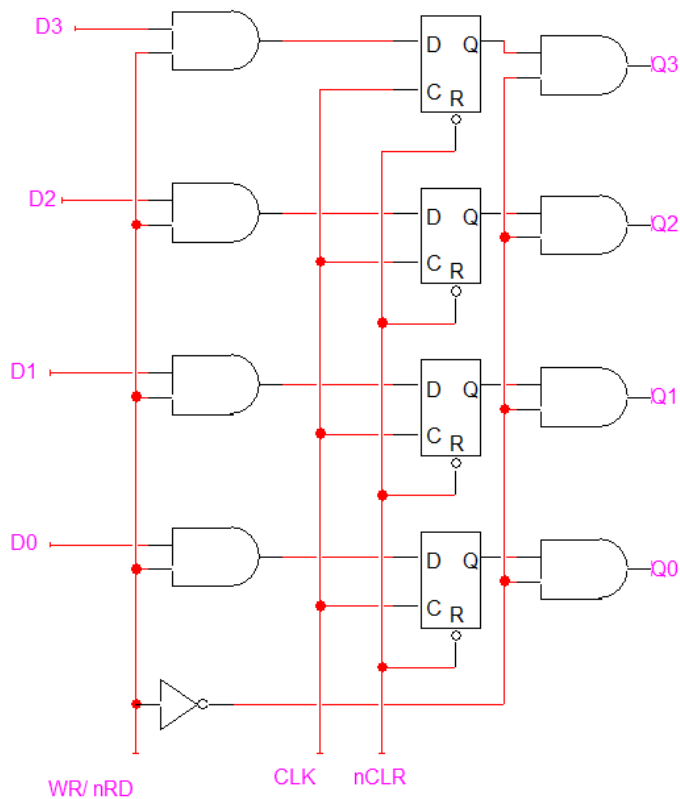
- **PISO** (Parallel Input - Serial Output, параллельный вход - последовательный выход);

- **PIPO** (Parallel Input - Parallel Output, параллельный ввод - параллельный вывод).

Параллельные регистры.

Основная функция этих регистров - хранить двоичные слова.

Пример. 4-битный параллельный регистр



В режиме параллельной загрузки слово для ввода подается на входы данных D3, D2, D1 и D0 – порт ввода. Данные загружаются в регистр, когда подается общий для всех триггеров тактовый сигнал CLK и сигнал записи $WR/nRD = 1$ (запись / n чтение). Вход сброса CLR (clear), активный по логическому нулю, присутствует в большинстве регистров и служит для асинхронной установки в 0 всех n триггеров. Введенное слово доступно на выходах Q3, Q2, Q1 и Q0 - порт вывода. Для параллельного чтения данных применяется сигнал $WR / nRD = 0$.

Последовательные регистры (сдвигающие)

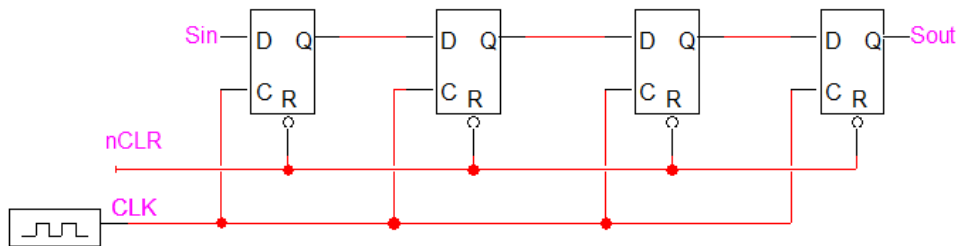
В регистрах сдвига двоичные слова можно вводить последовательно через один из входов, обозначенным S_{IN} (последовательный ввод), сдвигать двоичное слово вправо или влево и, соответственно, считывать последовательно только на одном из выходов, обозначенных S_{OUT} (последовательный выход). Вход сброса nCLR (clear), активный по логическому нулю служит для асинхронной установки в 0 всех n триггеров.

Сдвиг двоичных слов бывает 3-х видов: логический, арифметический и циклический.

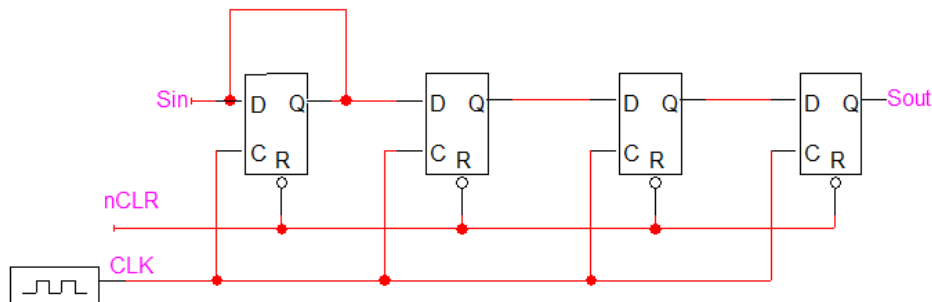
Регистр со сдвигом вправо

Этот тип регистра состоит из n триггеров типа D. Вход k + 1 триггера соединен с выходом Q k-триггера. Информация перемещается слева направо. Используемые D-триггеры являются двухтактными триггерами (flip-flop) и переключаются на заднем фронте тактового сигнала.

4-битный регистр со сдвигом вправо (логический сдвиг)

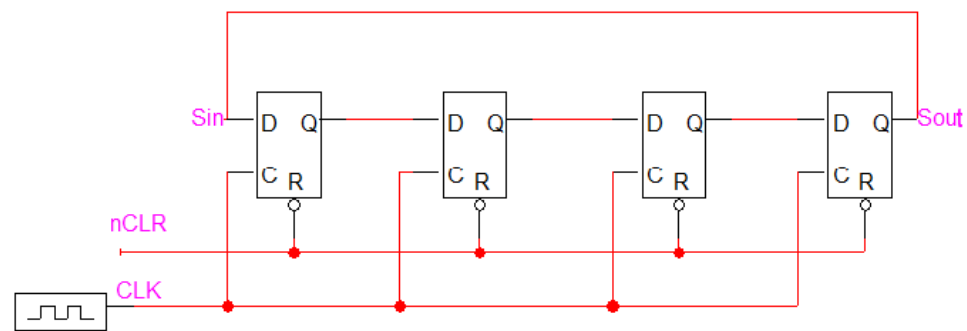


4-битный регистр со сдвигом вправо (арифметический сдвиг)

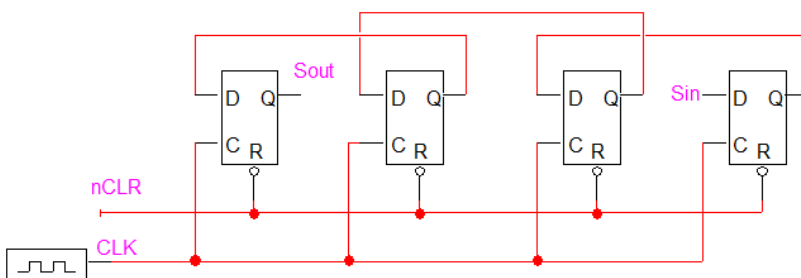


4-битный регистр со сдвигом вправо (циклический сдвиг)

N-битовое слово, записываемое в течение n тактов через вход Sin, последовательно побитно теряется на выходе Sout в течение следующих n тактов. Однако, если выход Sout подключить к входу Sin, получается кольцевая структура или структура регистров с циклическим сдвигом. В таком регистре первоначально введенное слово циркулирует внутри регистра.

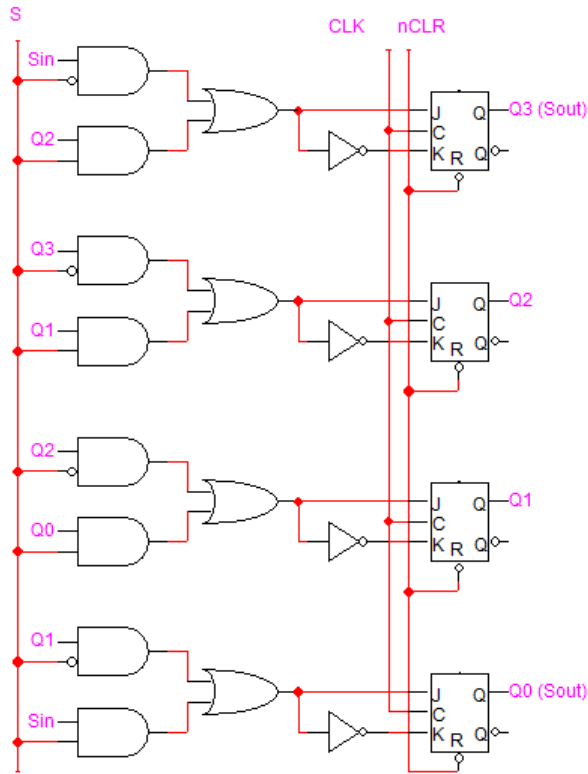


4-битный регистр со сдвигом влево (логический сдвиг)



Регистр, способный сдвигать данные как влево, так и вправо, называется реверсивным регистром сдвига. Этот регистр можно использовать не только для хранения двоичного

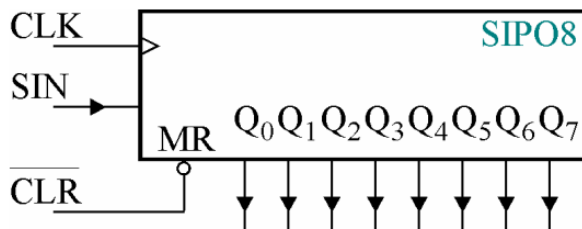
слова, но и для его обработки, потому что сдвиг вправо на i позиций эквивалентно операции деления слова на 2^i , а сдвиг влево - операции умножения на 2^i .



Применения регистров

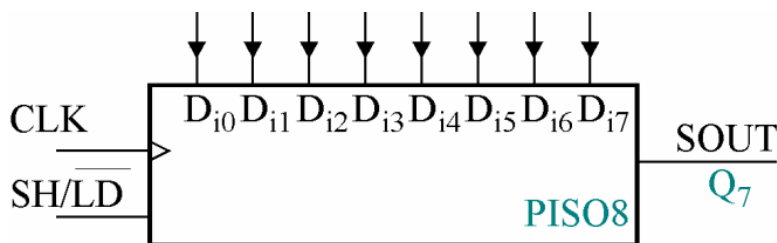
1. Преобразование последовательного двоичного слова в параллельное.

Последовательно-параллельное преобразование полезно для увеличения количества выходов в микроконтроллерах, особенно для микроконтроллера с небольшим количеством выходов. Используемый регистр - SIPO.



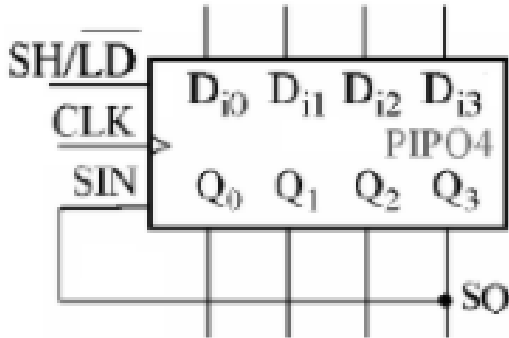
2. Преобразование параллельного двоичного слова в последовательное.

Используемый регистр - PISO. Параллельно-последовательное преобразование полезно для увеличения количества входов в микроконтроллерной системе.



3. Кольцевой счетчик

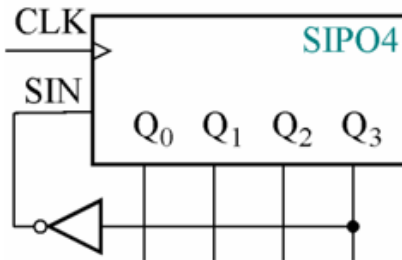
Основное применение - последовательная генерация управляющих сигналов, предназначенных для последовательного переключения n цифровых схем. Счетчик использует комбинированный регистр с параллельной загрузкой и чтением (PIPO), снабженный последовательным вводом и выводом. При параллельной загрузке регистр (кольцевой счетчик) всегда инициализируется двоичным словом из n битов, на одном бите записывается логическая единица, на остальных – 0. Параллельные выходы необходимы для получения управляющих сигналов без необходимости в декодере.



4. Счетчик Johnson

В этом варианте используется более простой регистр (последовательный ввод, параллельный вывод, SIPO).

Инициализация счетчика выполняется подачей импульса сброса. Количество различных состояний схемы в случае полного рабочего цикла $2n$.



Nr. tact	Q3	Q2	Q1	Q0
Inițializare	0	0	0	0
1	0	0	0	1
2	0	0	1	1
3	0	1	1	1
4	1	1	1	1
5	1	1	1	0
6	1	1	0	0
7	1	0	0	0
8	0	0	0	0
9	0	0	0	1
10	0	0	1	1

5. Память FIFO и LIFO

Они представляют собой временную память, организованную как n двоичные слова по m бит каждое, состоящих из m регистров последовательного сдвига SISO по n бит каждый.

FIFO – First In First Out (первое написанное слово - первое прочитанное).

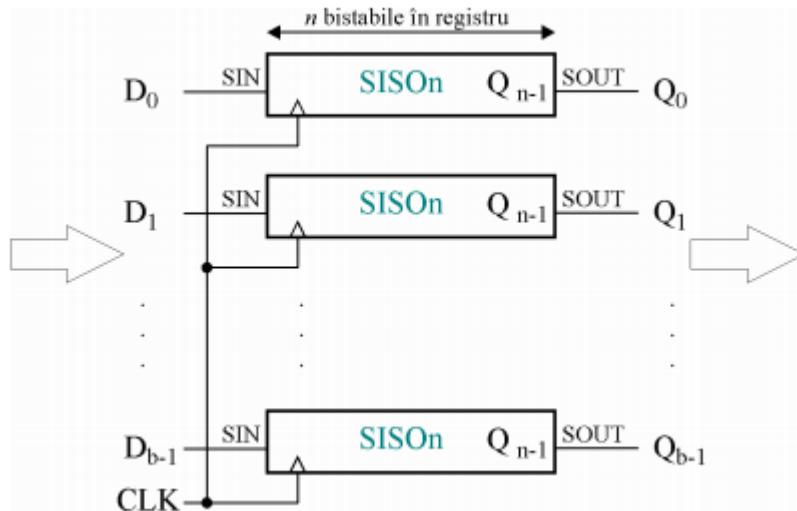
LIFO - Last In First Out (последнее написанное слово - первое прочитанное).

В зависимости от памяти сдвиг может быть односторонним или реверсивным. Если сдвиг происходит в обоих направлениях, в регистре должен присутствовать вход, указывающая направление сдвига.

Память FIFO

Запись двоичных слов из b битов в память выполняется параллельно на b последовательных входах при подачи сигнала синхронизации и сдвига регистров вправо. Считывание также выполняется параллельно на последовательных выходах, при подачи сигнала синхронизации. Память заполнена, когда введены все b -битные слова. После того, как память полностью заполнена, первое прочитанное слово - это первое слово, записанное в память. В процессе чтения, информация продолжает сдвигаться вправо с каждым импульсом сигнала синхронизации. При чтении информация теряется!

Этот тип памяти может использоваться для управления адресами других запоминающих устройств во время сбоя в работе микропроцессорной системы.



Память LIFO

Для этой памяти требуется двунаправленный регистр SISO. В этом случае регистры имеют управляющий сигнал R/nL , который определяет направление сдвига. Слова записываются как в памяти FIFO, сдвигая данные вправо при $R/nL = 1$, а чтение выполняется путем сдвига в противоположном направлении (влево) при $R/nL = 0$. Таким образом, последнее записанное слово также является первым прочитанным. Выходы данных представлены выходами $Q(n-1)$ каждого регистра.

Память LIFO используется как стековая память в микропроцессорных системах.

Синтез регистров

Покажем, как можно выполнить синтез 4-разрядного регистра с параллельной загрузкой и реверсивным сдвигом. Предположим, что синтезируемый регистр должен обеспечивать и режим хранения загруженного в регистр слова. Число триггеров равно четырем. Для обеспечения режима хранения коммутатор в i -ом разряде должен иметь дополнительный вход, который следует подключить к выходу триггера i -го разряда.

Два управляющих сигнала s_1 и s_0 используются для выбора рабочего режима.

Количество управляющих сигналов n зависит от количества рабочих режимов, и определяется согласно формуле $n = \lceil \log_2 m \rceil$.

Интегральная схема универсального регистра 74-194

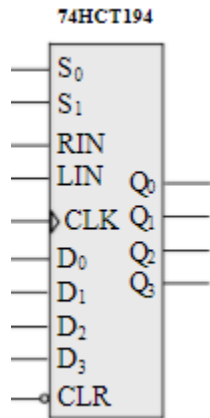


Таблица переходов для универсального регистра:

Режим работы	Сигналы управления		Выходы				t
	S1	S0	Q3	Q2	Q1	Q0	
Сохранение	0	0	Q_3	Q_2	Q_1	Q_0	t+1
Сдвиг влево	0	1	Q_2	Q_1	Q_0	DSS	
Сдвиг вправо	1	0	DSD	Q_3	Q_2	Q_1	
Параллельная загрузка	1	1	D_3	D_2	D_1	D_0	

Логические формулы:

$$J_3 = \bar{K}_3 = \bar{s}_1 \bar{s}_0 Q_3 \vee \bar{s}_1 s_0 Q_2 \vee s_1 \bar{s}_0 DSD \vee s_1 s_0 D_3,$$

$$J_2 = \bar{K}_2 = \bar{s}_1 \bar{s}_0 Q_2 \vee \bar{s}_1 s_0 Q_1 \vee s_1 \bar{s}_0 Q_3 \vee s_1 s_0 D_2,$$

$$J_1 = \bar{K}_1 = \bar{s}_1 \bar{s}_0 Q_1 \vee \bar{s}_1 s_0 Q_0 \vee s_1 \bar{s}_0 Q_2 \vee s_1 s_0 D_1,$$

$$J_0 = \bar{K}_0 = \bar{s}_1 \bar{s}_0 Q_0 \vee \bar{s}_1 s_0 DSS \vee s_1 \bar{s}_0 Q_1 \vee s_1 s_0 D_0.$$

Схема регистра

