



# ОСНОВЫ ПРОГРАММИРОВАНИЯ СИСТЕМ УПРАВЛЕНИЯ



Лекция №4

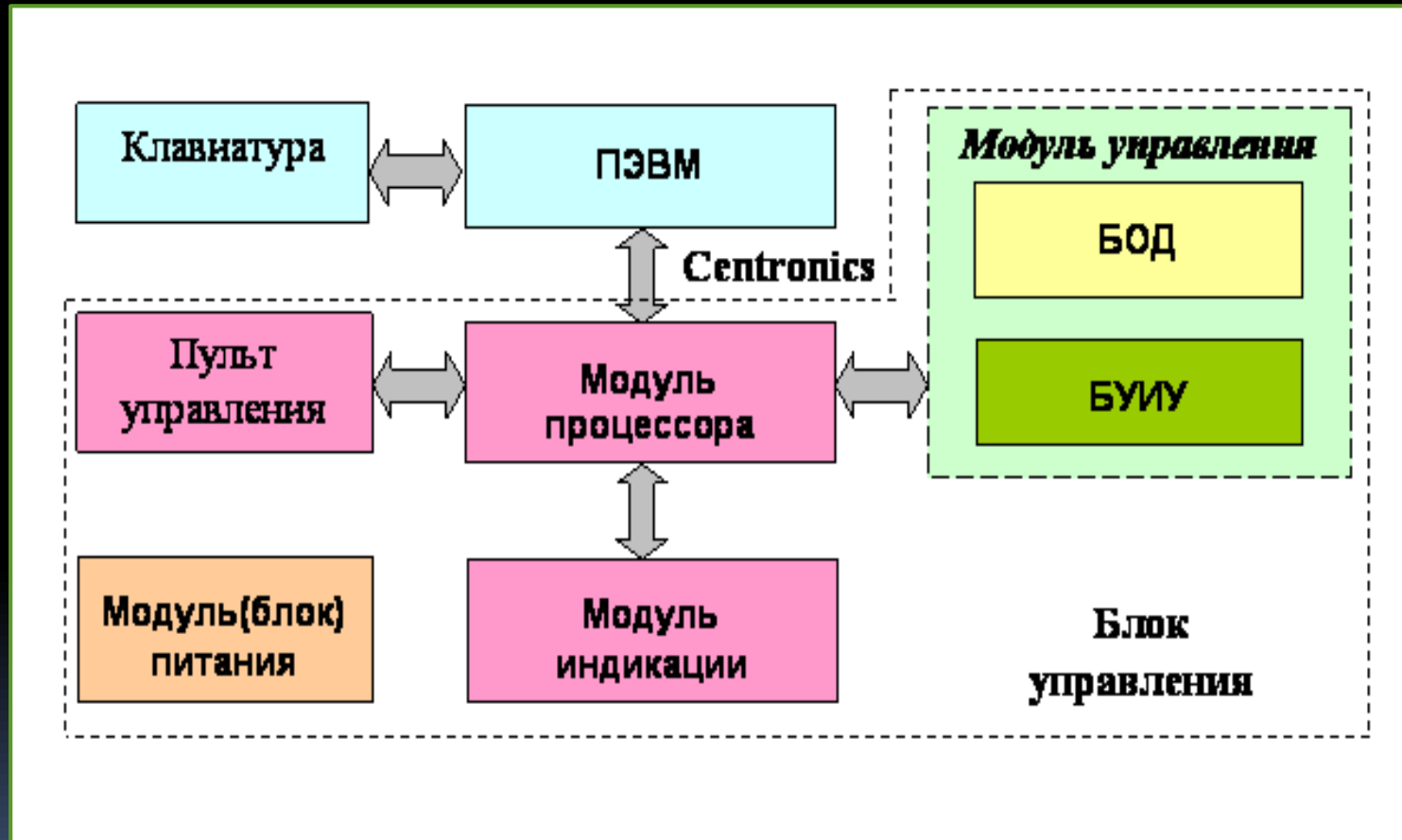
# Блок управления.

Устройство сопряжения – это комплекс аппаратно-программных средств, реализующий взаимодействие ЭВМ с периферийными устройствами.

Модуль процессора позволяет управлять:

- Шестью реверсивными мехатронными модулями (степени продольного, поперечного или вертикального перемещения робота);
- Тремя нереверсивными мехатронными модулями (захватное или фрезерное устройство).

# Структурная схема системы управления.



БОД – блок обработки датчиков.

БУИУ – блок управления исполнительными устройствами.

# Функции модулей блока управления.

- **Модуль процессора** микроконтроллера выполняет обмен данными с ПЭВМ, модулем управления, модулем индикации; обрабатывает сигналы с пульта управления и с клавиатуры; посылает управляющие команды модулю индикации и модулю управления.
- **Модуль управления** производит первичную обработку состояния датчиков и обеспечивает подачу напряжения на двигатели робота.
- **Модуль индикации** выполняет обработку элементов управления, расположенных на передней панели блока управления: светодиодной панели, кнопок сброса СУ и управления цепями высокого напряжения питания.
- **Блок питания** является источником питания для слаботочных схем всех модулей (логических схем) и для сильноточных схем исполнительных устройств робота (двигатели и электромагнитные реле).

# Исполнительные устройства УРТК.



Двигатели, перемещающие степени робота;



Обмотки возбуждения двигателей;



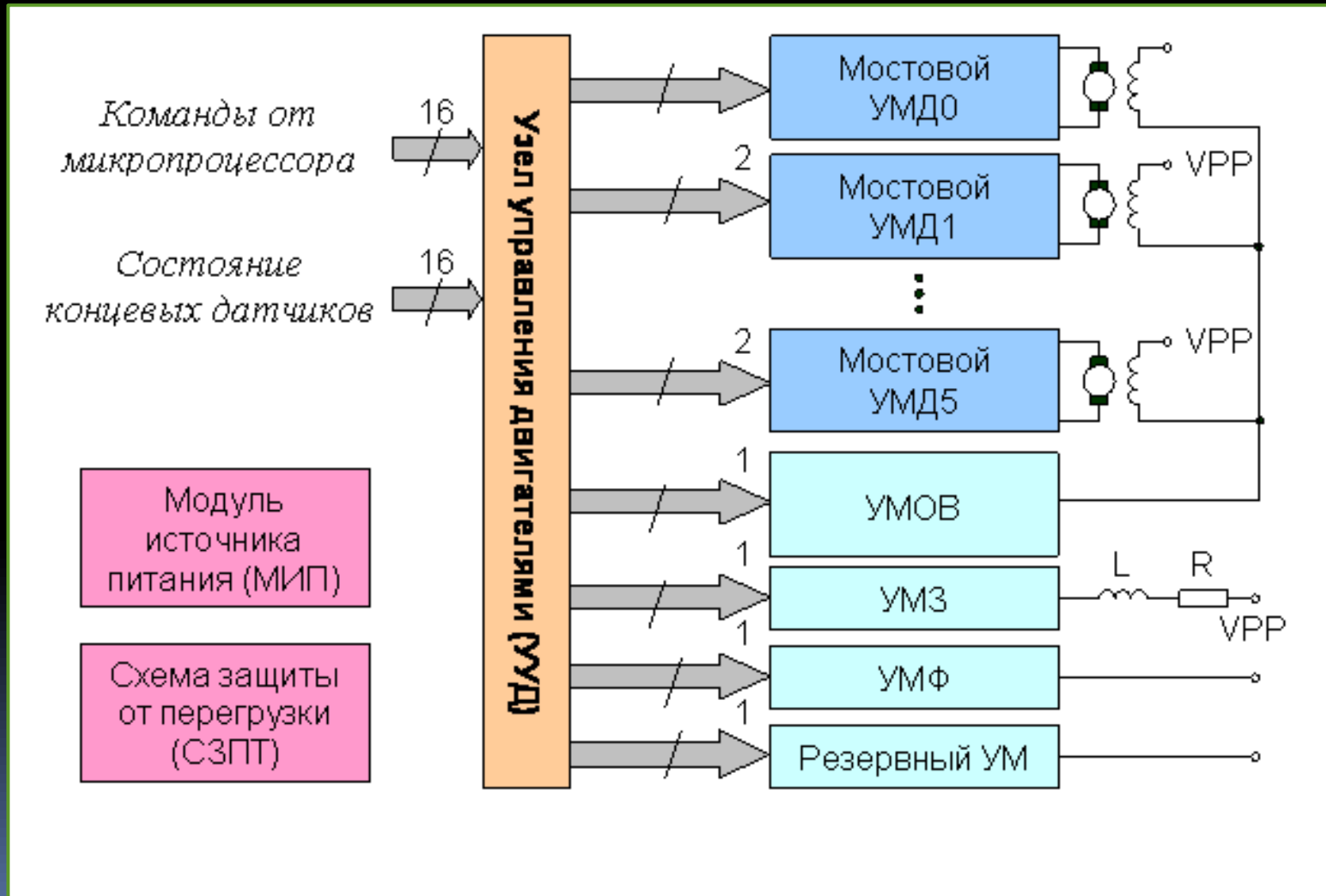
Захватное и фрезерное устройства;



Резервный канал СУ.



# Структурная схема блока управления ИИ в БУ УРТК.



УМД0-УМД5 – усилители мощности двигателей 0-5.

УМОВ – усилитель мощности обмоток возбуждения двигателей.

УМЗ – усилитель мощности захватного устройства.

УМФ – усилитель мощности устройства фрезеровки.

# Функции узла управления двигателями.

- Принимает команды от модуля процессора и сигналы состояния концевых датчиков положения.
- Проверяет состояние концевых датчиков положения данной степени.
- Подает сигнал на соответствующий усилитель мощности.
- Выключает двигатель при поступлении сигнала о достижении одного из концевых датчиков.
- Подает напряжение на обмотки возбуждения двигателей.
- Осуществляет включение захватного устройства и устройства фрезеровки.

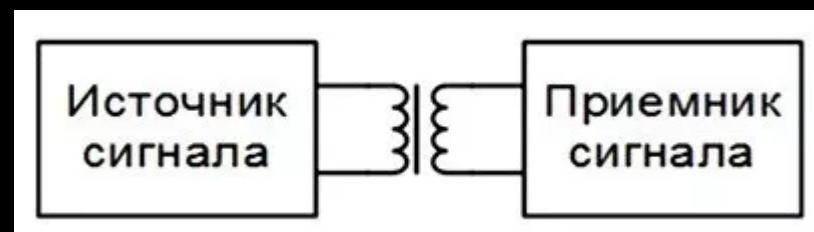
# Гальваническая развязка.

Гальваническая развязка — передача энергии или информационного сигнала между электрическими цепями, не имеющими непосредственного электрического контакта между ними.

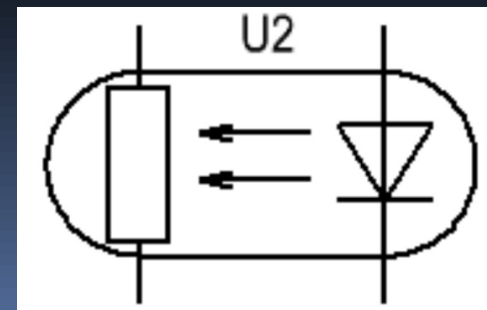
Виды гальванических развязок:

- Трансформаторные.
- Оптоэлектронные.
- Акустические.
- Радиоканалы.
- Звуковые.
- Емкостные.
- Механические.

Трансформаторная развязка:



Оптрон:





# Узел управления двигателями.

Число линий шины сопряжения УУД и УГР - 16 разрядов.

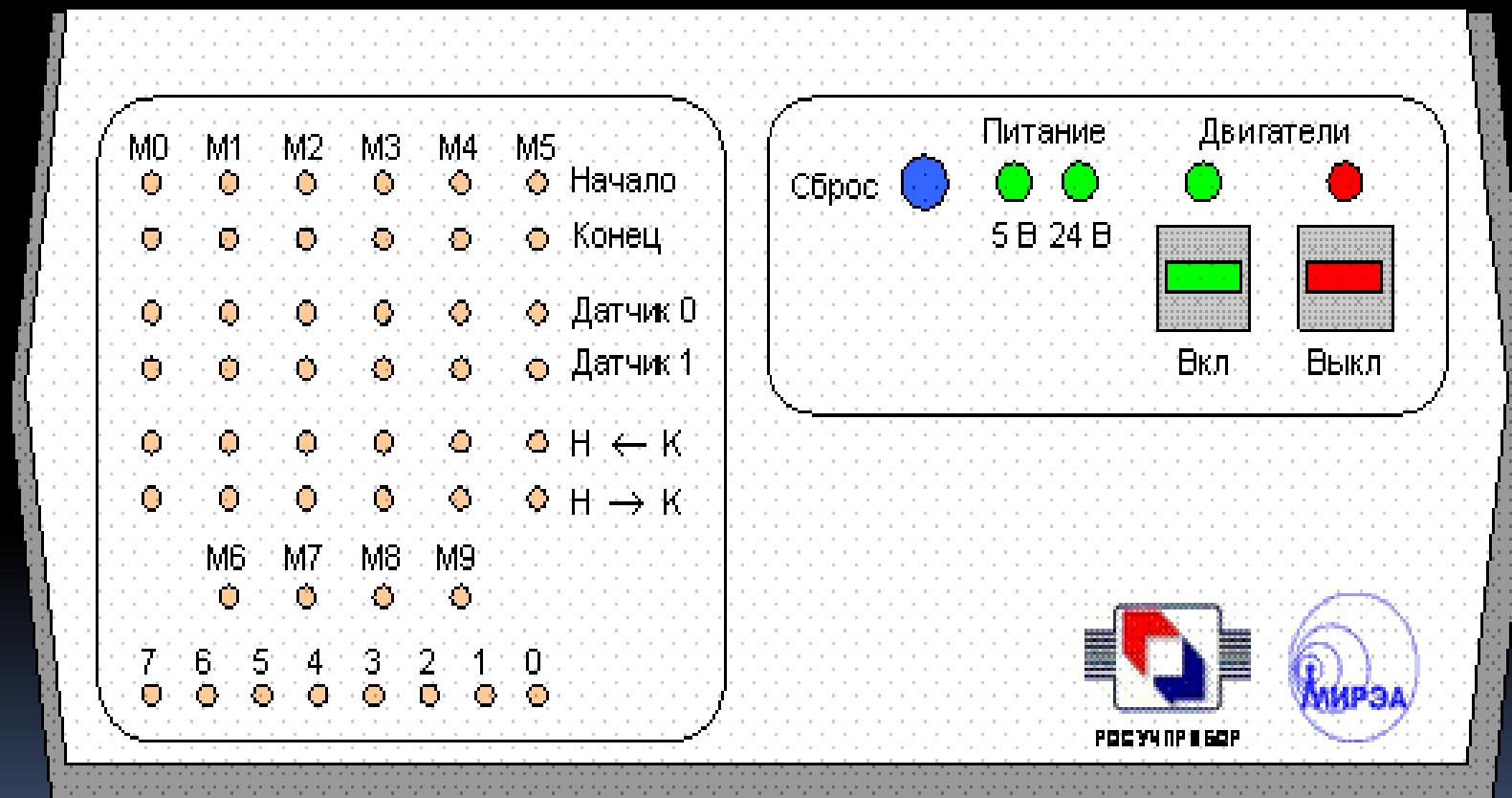
На каждый УМД приходит 2 разряда управления.

Высокий уровень напряжения (логическая единица) на каждом из разрядов обеспечивает вращение двигателя в соответствующем направлении.

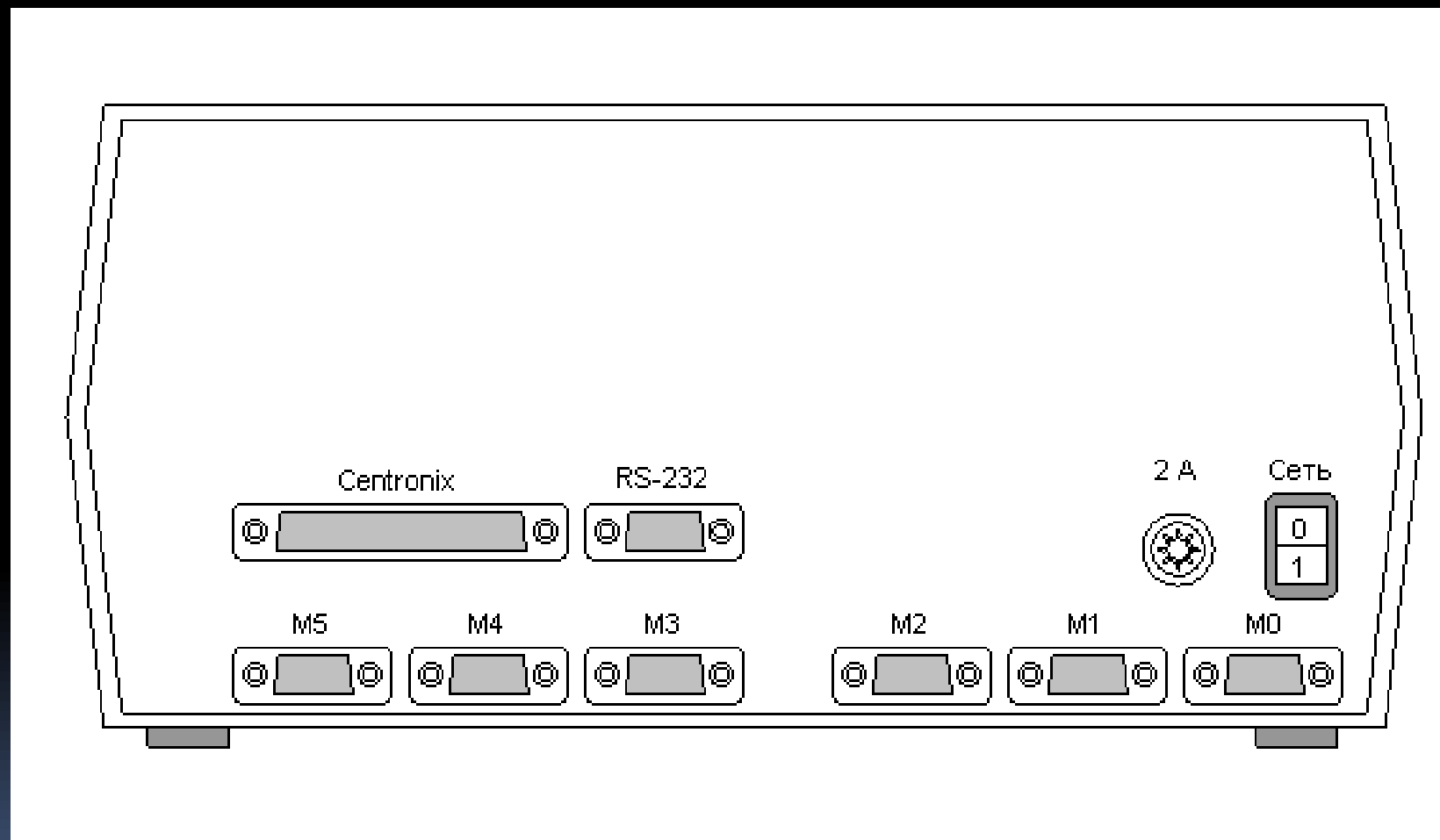
Низкий уровень напряжения (логический ноль) на обоих разрядах при отсутствии напряжения на обмотке – двигатель останавливается.

Высокий уровень напряжения на обоих разрядах – двигатель неподвижен.

# Передняя панель блока управления.



# Задняя панель блока управления.






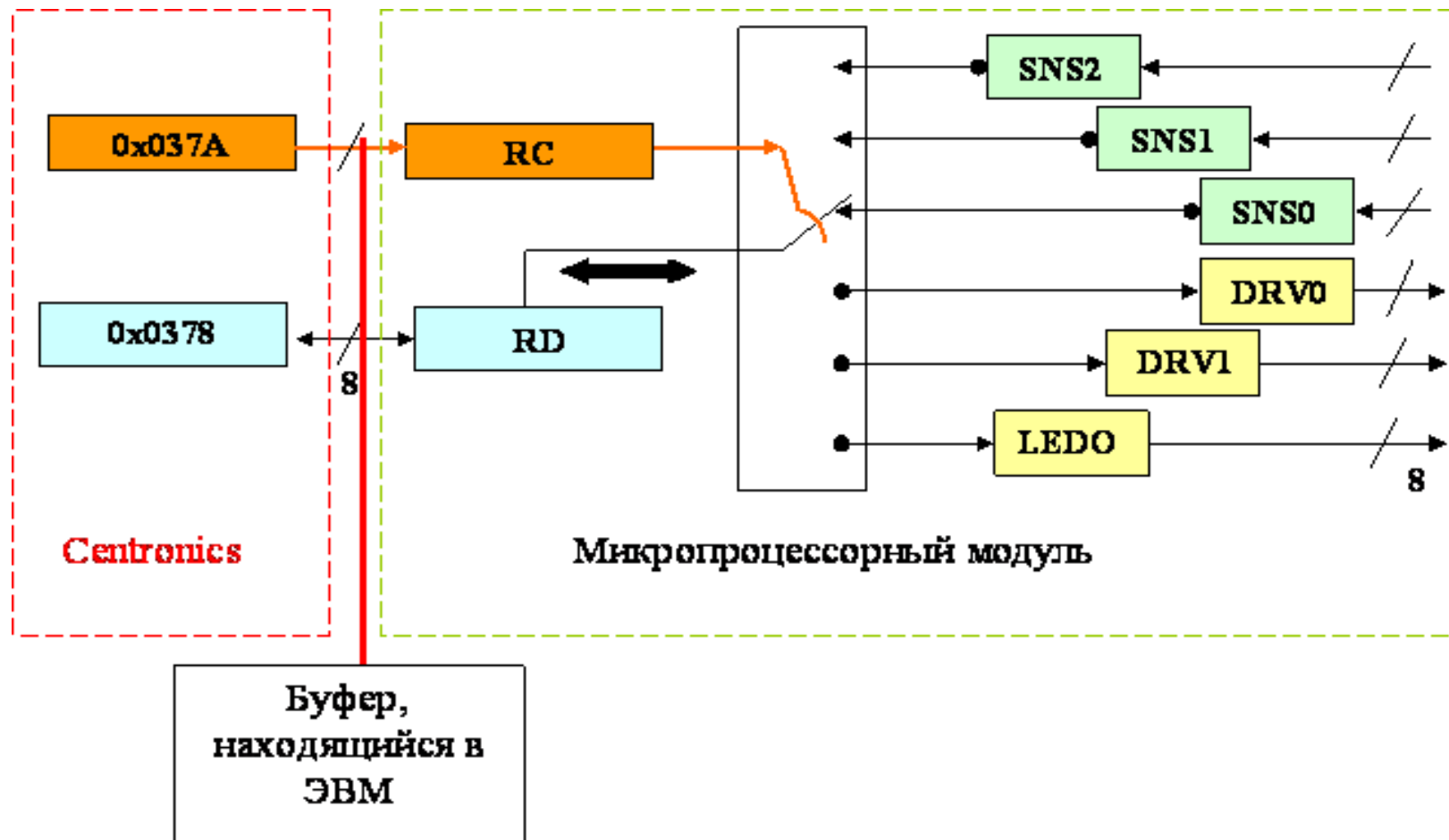
# Устройство сопряжения.

Устройством сопряжения в широком смысле этого определения называется любое устройство обеспечивающее взаимодействие между двумя техническими средствами (системами).

Предназначение:

- Прием информации от компьютера.
  - Обработка информации по заданному алгоритму.
  - Выдача результата обработки информации в компьютер.
- 

# Схема обмена данными между Centronics и микропроцессорным модулем БУ.



# Чтение и запись информации.

## *Схема диалога с блоком управления:*

- В регистр RC заносится константа – управляющее слово (команда), определяющая режим диалога.
- Через регистр RD производится чтение или запись данных. Если необходимо передать более одного байта, передача байт через регистр RD производится последовательно.
- После отправки последнего байта действие считается завершенным.

# Алгоритм действий.

Значение константы	Описание действия	Алгоритм действия
11h	Установка состояния двигателей	1. В регистр RC отправить константу 11h; 2. Через регистр RD последовательно передать два байта состояния двигателей: DRV1, DRV0.
12h	Установка состояния светодиодных индикаторов	1. В регистр RC отправить константу 12h; 2. Через регистр RD передать байт состояния светодиодных индикаторов LED0.
13h	Получение текущего состояния датчиков	1. В регистр RC отправить константу 13h; 2. Через регистр RD последовательно принять три байта состояния датчиков: SNS0, SNS1 и SNS2.
14h	Получение текущего состояния клавиатуры	1. В регистр RC отправить константу 14h; 2. Через регистр RD последовательно принять три байта состояния клавиатуры: KBD0 и KBD1.

# Согласование временных циклов.

Необходимость согласования:

- Различные аппаратные средства и различное ПО СУ имеют различное быстродействие.
- Выполнение операций может быть синхронным или асинхронным.
- Выдача порции новой информации может быть разнотемповой.
- Различный объем информации.

Варианты решения:

- Применение временных задержек.

Функция `delay();`





# Запись и чтение данных из регистра.

## Регистр управления (параллельный порт):

- Используется для управления процессом передачи данных.
- Заносится информация об адресе регистра назначения (RC или RD)
- Заносится информация о характере выполняемой операции (чтение или запись);



## Регистр данных (параллельный порт):

- Используется для физической передачи данных по физическому каналу в оба направления



# Функции чтения и записи.

***Запись данных:***

*outportb(Addr, Data)*

*Где:*

Addr – адрес порта,

Data – записываемые данные.

***Чтение данных:***

*inportb(Addr)*

*Где:*

Addr – адрес порта.

# Назначение разрядов регистра управления Centronics

Номер разряда	Название	Инверсия	Назначение разряда
0	RD	Да	Строб чтения данных из СУ
1	A0	Да	Младший разряд шины адреса
2	WR	Нет	Строб записи данных в СУ
3	A1	Да	Старший разряд шины адреса
4			Не используется. Должен быть равен 0
5	B5	Нет	Дополнительный бит для режима чтения
6			Не используется. Должен быть равен 0
7			Не используется. Должен быть равен 0



# Адрес регистра.

Адрес регистра (RC или RD) формируют два разряда байта  
 $A_0$  и  $A_1$

Если:

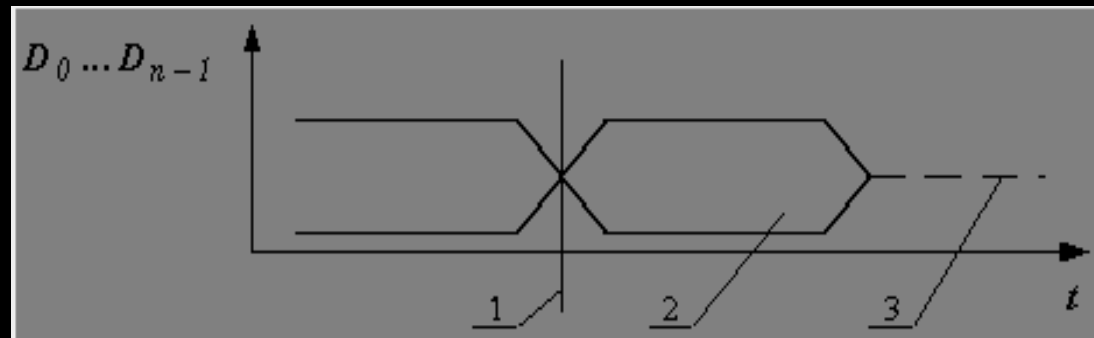
$A_0=1$  и  $A_1=1 \Rightarrow RD$

Если:

$A_0=0$  и  $A_1=0 \Rightarrow RC$

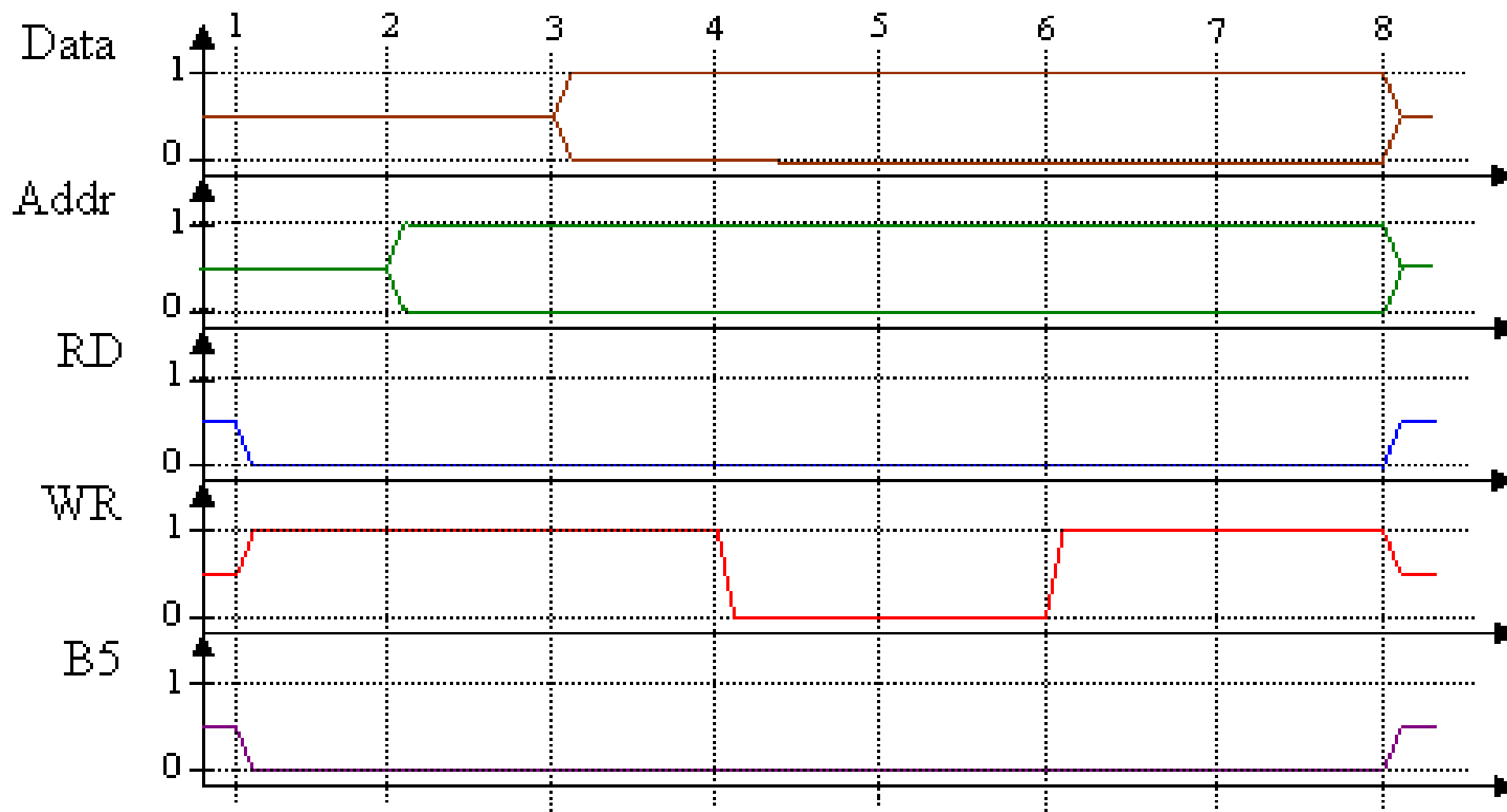
# Временные диаграммы.

**Временная диаграмма** - это изображение, представляющее определенный период времени и события, происходящие в этот период.



1. момент синхронного изменения кода, т.е. уровней сигнала на всех проводниках, образующих шину (могут изменяться не все сигналы шины);
2. неизменное устойчивое состояние сигналов на шине;
3. отсутствие сигнала в случае, когда выходные буферы всех устройств, подсоединенных к шине, находятся в отключенном (третьем) состоянии.

# Временные диаграммы записи данных в СУ.



# Последовательность этапов записи.

- Регистр RC устанавливается в начальное положение (бит соответствующий B<sub>5</sub> в алгоритме записи не используется).
- Происходит выбор адреса регистра в который будет записан байт.
- Происходит запись в регистр RC управляющей константы.
- Происходит запись данных в выбранный регистр.
- На пятом интервале – задержка.
- Регистр RC устанавливается в начальное положение и завершается передача байта.

## Формирование данных.

1-й такт :  $RD \sim 0$  ,  $A_0 \sim 0$  ,  $WR \sim 1$  ,  $A_1 \sim 0$  , 0 , 0 , 1 , 0

2-й такт :  $RD \sim 0$  ,  $A_0 \sim 0$  ,  $WR \sim 1$  ,  $A_1 \sim 0$  , 0 , 0 , 1 , 0

3-й такт : В регистр ох378 байт *btt* пишется целиком

4-й такт :  $RD \sim 0$  ,  $A_0 \sim 0$  ,  $WR \sim 0$  ,  $A_1 \sim 0$  , 0 , 0 , 0 , 0

5-й такт : Устанавливается задержка перед записью в *RC*

6-й такт :  $RD \sim 0$  ,  $A_0 \sim 0$  ,  $WR \sim 1$  ,  $A_1 \sim 0$  , 0 , 0 , 1 , 0

7-й такт : На этом такте ничего не изменяется

8-й такт : На этом такте регистр *RC* устанавливается в исходное состояние



# Формирование данных.

- 00000100 ~ 0<sub>x</sub>04 или `outportb(0x37A,0x04);`
- 00000100 ~ 0<sub>x</sub>04 или `outportb(0x37A,0x04);`
- Байт `btt` пишется целиком ~ `outportb(0x378, btt);`
- 00000000 ~ 0<sub>x</sub>00 или `outportb(0x37A,0x00);`
- Задержка 2-7 ms или `delay(5);`
- 00000100 ~ 0<sub>x</sub>04 или `outportb(0x37A,0x04);`

! На 7-м и 8-м такте ничего не меняется, поэтому операторы можно не писать

При записи в RC байт `btt` может иметь значение одной из констант 11h, 12h, 13h, 14h