



**YOU + MICROCHIP    ENGINEERING THE FUTURE TOGETHER**

# RTOS

## Операционные системы реального времени

# План лекции

- | Определение операционной системы
- | Особенности встраиваемых ОС
- | Процессы, задачи, нити
- | Системное время
- | Межпроцессное взаимодействие
- | Обработка прерываний
- | Примеры ОСРВ
- | Распределение функций

# Определение ОС

Операционная система – программная надстройка над архитектурой вычислительной машины

- | обеспечивает удобный пользовательский интерфейс;
- | берет на себя функции автоматического управления рядом подсистем;
- | предоставляет готовые процедуры управления внутренними и внешними ресурсами.

# Определение ОС

Преимущества использования ОС:

- | Формализованный набор средств для организации процессов
- | Механизмы планирования и приоритетов
- | Переносимость пользовательского ПО

# Определение ОС

Ограничения использования ОС:

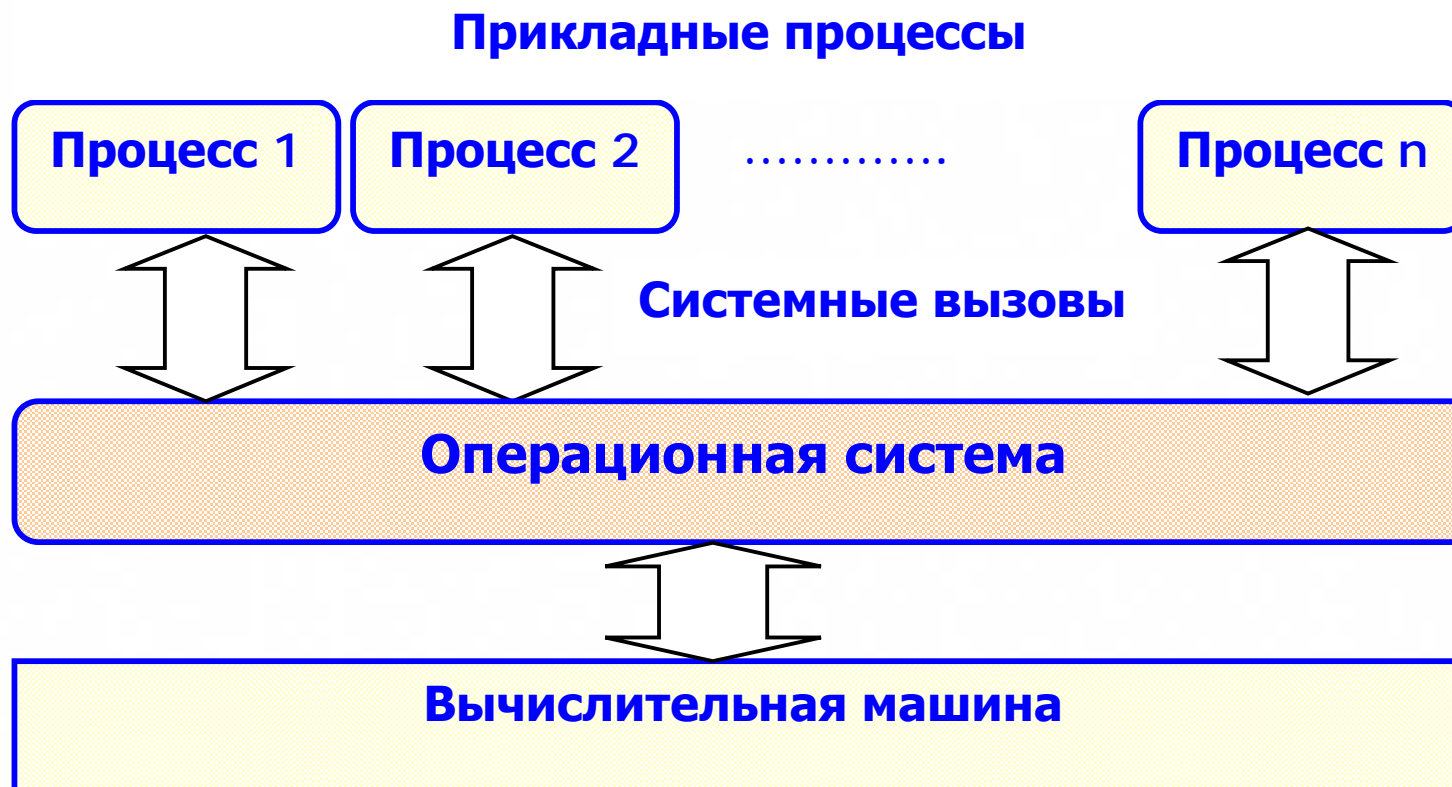
- | Недостаточная производительность микроконтроллера
- | Необходимость формирования «быстрых» временных диаграмм
- | При вытесняющей многозадачности – отсутствие механизмов управления стеком

# Определение ОС

Классификация ОС:

- | однозадачные или многозадачные
- | невытесняющая или вытесняющая многозадачность
- | поддержка нитей
- | пакетной обработки, разделения времени, реального времени
- | монолитное ядро или микроядерная ОС

# Определение ОС



# Многозадачность

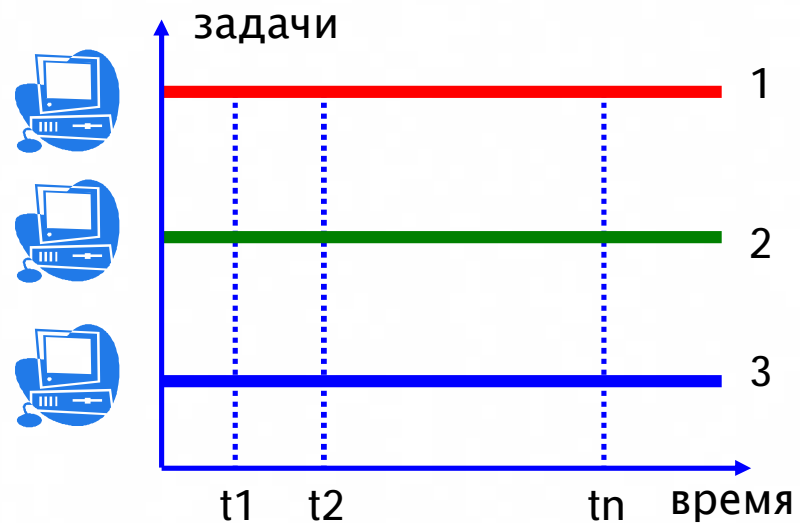
- | Многозадачная ОС распределяет ресурсы системы между несколькими процессами
- | Процессы выполняются последовательно
- | В зависимости от механизма распределения ресурсов ОС делятся на ОСРВ и ОС разделения времени



# Многозадачность

- | Вытесняющая многозадачность – операционная система может в любой момент забрать управление у текущего процесса
- | Невытесняющая (кооперативная) многозадачность – текущий процесс сам отдает управление системе

# Многозадачность



- Три вычислителя выполняют три разные задачи одновременно



- Один вычислитель выполняет все три задачи с разделением по времени

# Встраиваемые системы

- | Ограниченный объем физической памяти
- | Отсутствие файловой системы
- | Ограниченный набор периферии
- | Список процессов известен заранее, на этапе компиляции
- | Концепция «микроядра» - сведение к минимуму функций ядра системы

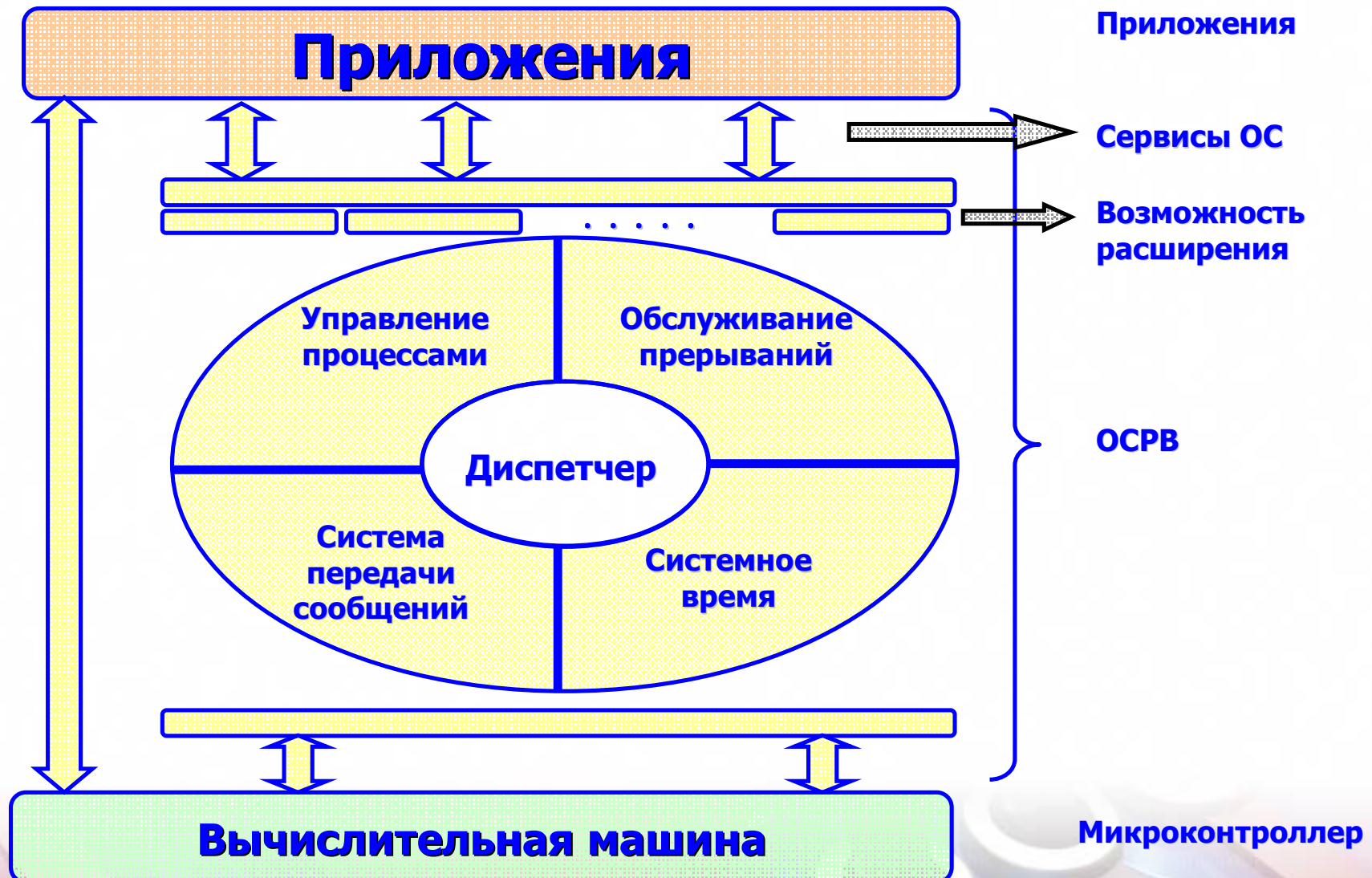
# ОС реального времени

- | Основное требование к распределению ресурсов между процессами – время реакции системы на внешнее событие
- | Время реакции – заданный интервал времени между запуском программы и получением результата (управляющего воздействия)

# ОС реального времени

- | RTOS жесткого реального времени
  - | Время реакции не должно превышать определенного интервала. Как правило время реакции детерминировано и составляет десятки мкс.
  - | Все вытесняющие RTOS для однокристальных встраиваемых систем
  - | Многие RTOS для однопроцессорных систем (например, VxWorks)
- | RTOS мягкого реального времени
  - | Время реакции не детерминировано, задержка не приводит к необратимым последствиям
  - | Кооперативные и Round-Robin RTOS
  - | Linux, Windows CE

# ОС реального времени



# Применение ОС – плюсы и минусы

## Плюсы:

- | Гибкое управление работой процессов
- | Предоставление базовых сервисов
- | Быстрая разработка и модификация ПО

## Минусы:

- | Большой объем памяти
- | Снижение производительности



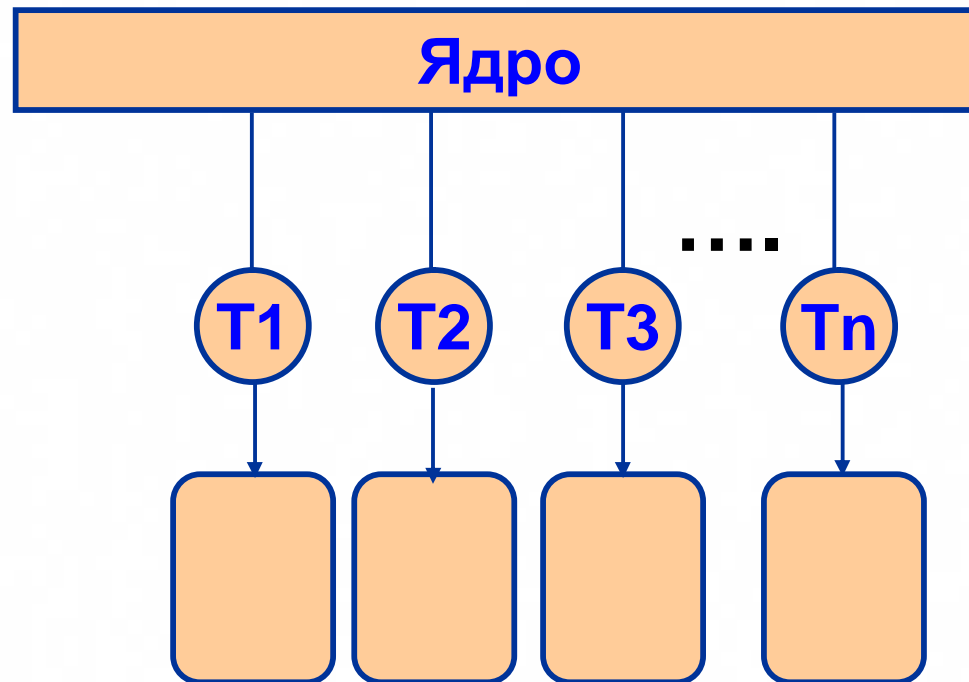
MASTERS  
CONFERENCE

# Процессы, задачи, потоки, нити

- | Процесс (задача) – абстракция, описывающая выполняющуюся программу
- | Процесс является единицей работы для операционной системы
- | Подсистема управления процессами – сердце операционной системы

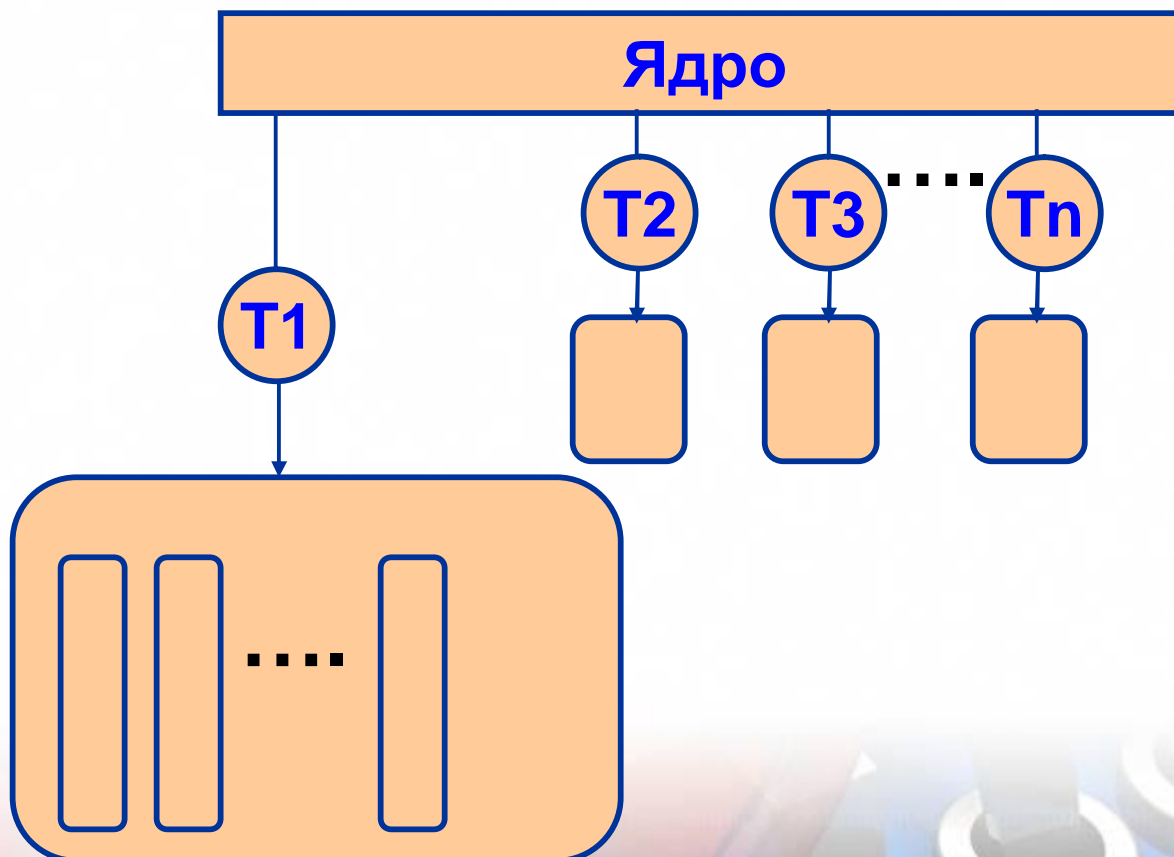


# Процессы



# Нити

- Нить (поток) – параллельная ветвь выполнения в рамках одного процесса



# Процессы, задачи, потоки, нити

## Граф состояния процесса



# Контекстное переключение

- | Приостановление выполнения одного процесса и запуск другого называется контекстным переключением
- | Во время переключения сохраняются данные о приостановленном процессе – контекст
- | Объем сохраняемых данных (контекст) у кооперативной ОС значительно меньше

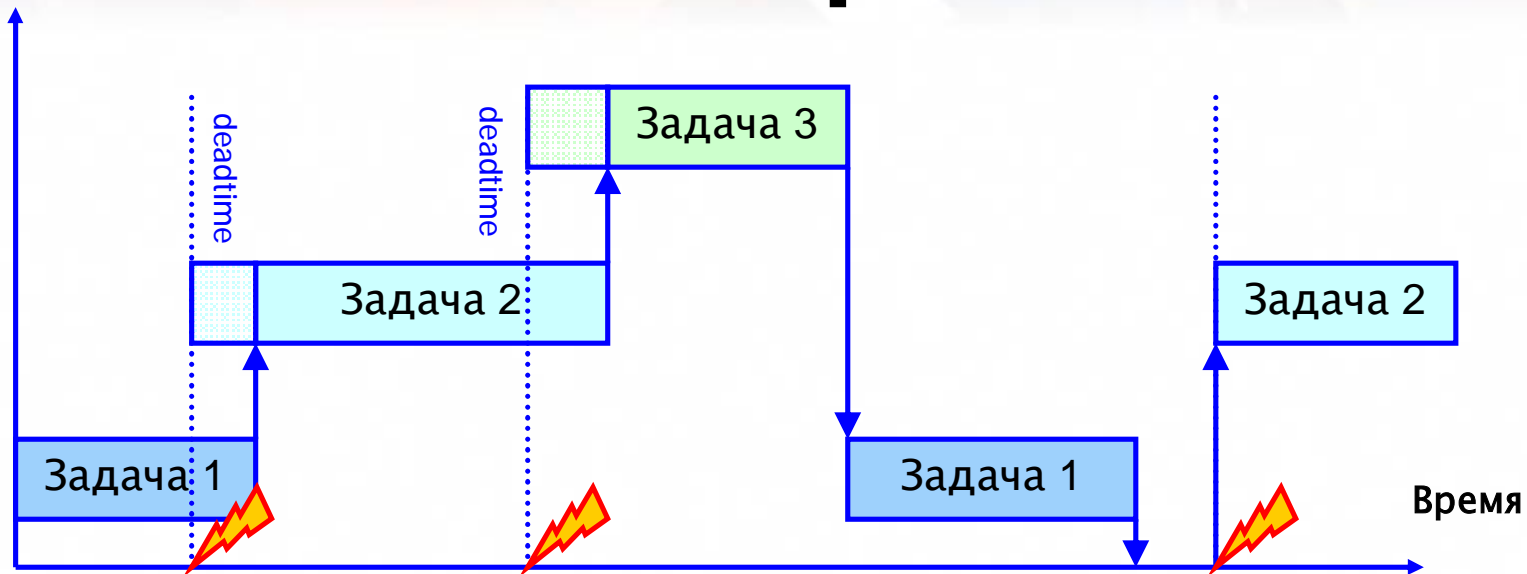
# Контекстное переключение

Приоритет задачи

ВЫСОКИЙ

СРЕДНИЙ

НИЗКИЙ



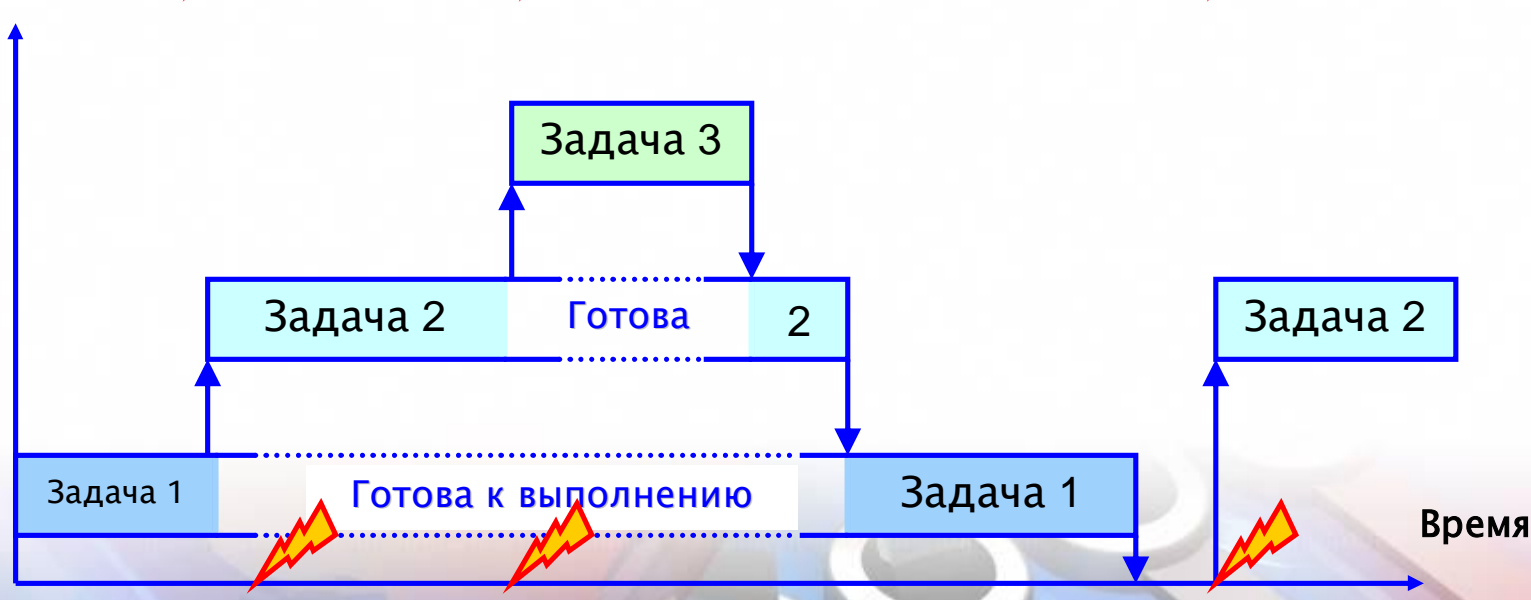
IDLE

ВЫСОКИЙ

СРЕДНИЙ

НИЗКИЙ

IDLE



# Приоритеты

- | Если ОС поддерживает приоритеты, то во время контекстного переключения на выполнение запустится наиболее приоритетный процесс
- | Алгоритм управления приоритетами – важная особенность ОС

# Системное время

- | Системное время измеряется тиками
- | Для отсчета тиков требуется периодическое событие
- | Чаще всего применяется таймер
- | Частота тиков выбирается из требований задач

# Системное время

- | При вытесняющей многозадачности контекстное переключение может производиться каждый тик
- | В кооперативной многозадачности тики используются для отсчета задержек и таймаутов



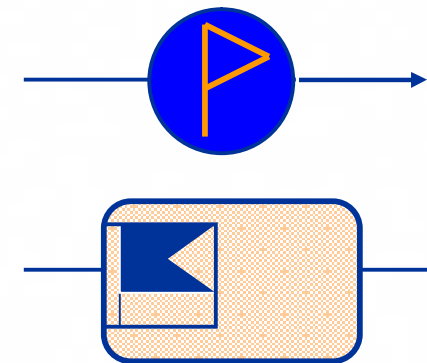
# RTOS для контроллеров MICROCHIP

Название	Тип	Семейство	
	кооперативная	PIC16, PIC18, PIC24, dsPIC	FREE
 The RTOS that runs in tiny places.™	кооперативная	PIC16, PIC18, PIC24, dsPIC, PIC32	~ 1K USD
	вытесняющая	PIC24, dsPIC, PIC32	> 5K USD
 The Real-Time Kernel	вытесняющая	PIC18, PIC24, dsPIC, PIC32	3K – 27K USD
	вытесняющая	PIC24, dsPIC, PIC32	~ 6K USD
	вытесняющая	PIC24, dsPIC	FREE
	вытесняющая	PIC18, PIC24, dsPIC	FREE
	вытесняющая	PIC18, PIC24, dsPIC, PIC32	FREE



# Межпроцессное взаимодействие (IPC)

- | Межпроцессное взаимодействие – вторая основная задача ядра ОС
- | Основные механизмы IPC:
  - | Счетные семафоры
  - | Бинарные семафоры (mutex)
  - | События
  - | Сообщения
  - | Очереди сообщений
  - | Разделяемая память



# IPC - Семафоры

- | Семафор – механизм ядра, предназначенный для синхронизации процессов
- | Аналог железнодорожного семафора
- | Для разных задач можно интерпретировать по-разному

# IPC - Семафоры

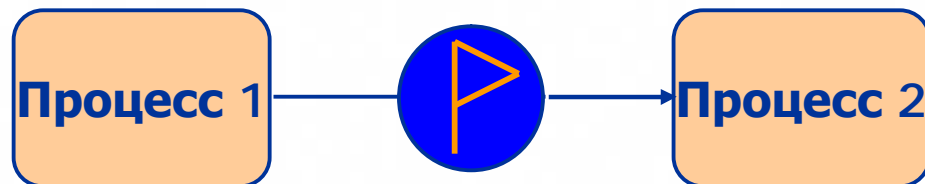
- | Счетный семафор имеет множество целочисленных значений:
  - | «не установлен» - 0
  - | «установлен» - 1
  - | «установлен» - 2
  - | И т.д.
- | Применяется для сложных синхронизаций

# IPC - Семафоры

- I Два основных действия:
  1. Установить семафор
  2. Проверить семафор
    1. Если установлен – сбросить
    2. Если сброшен – ждать

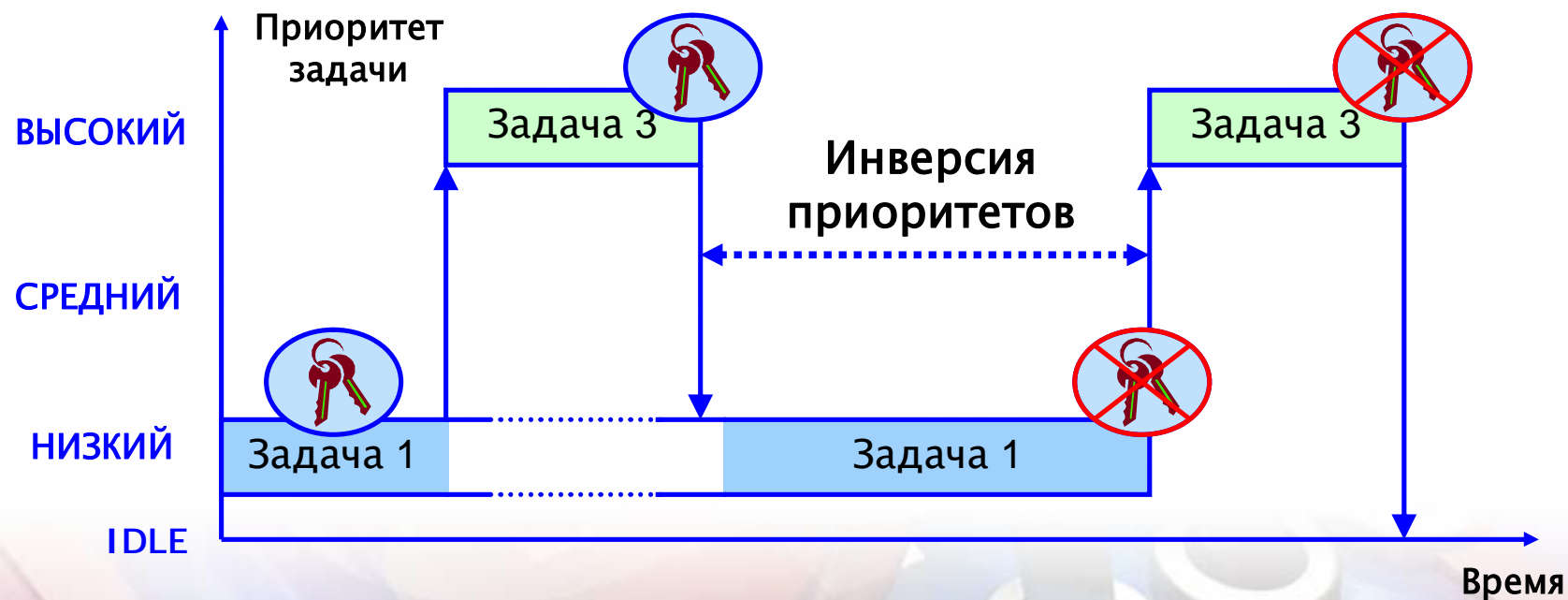
# IPC - Mutex

- Бинарный семафор имеет два значения:
  - «не установлен» - 0
  - «установлен» - 1
- Бинарный семафор + управление приоритетами = mutex



# IPC - Mutex

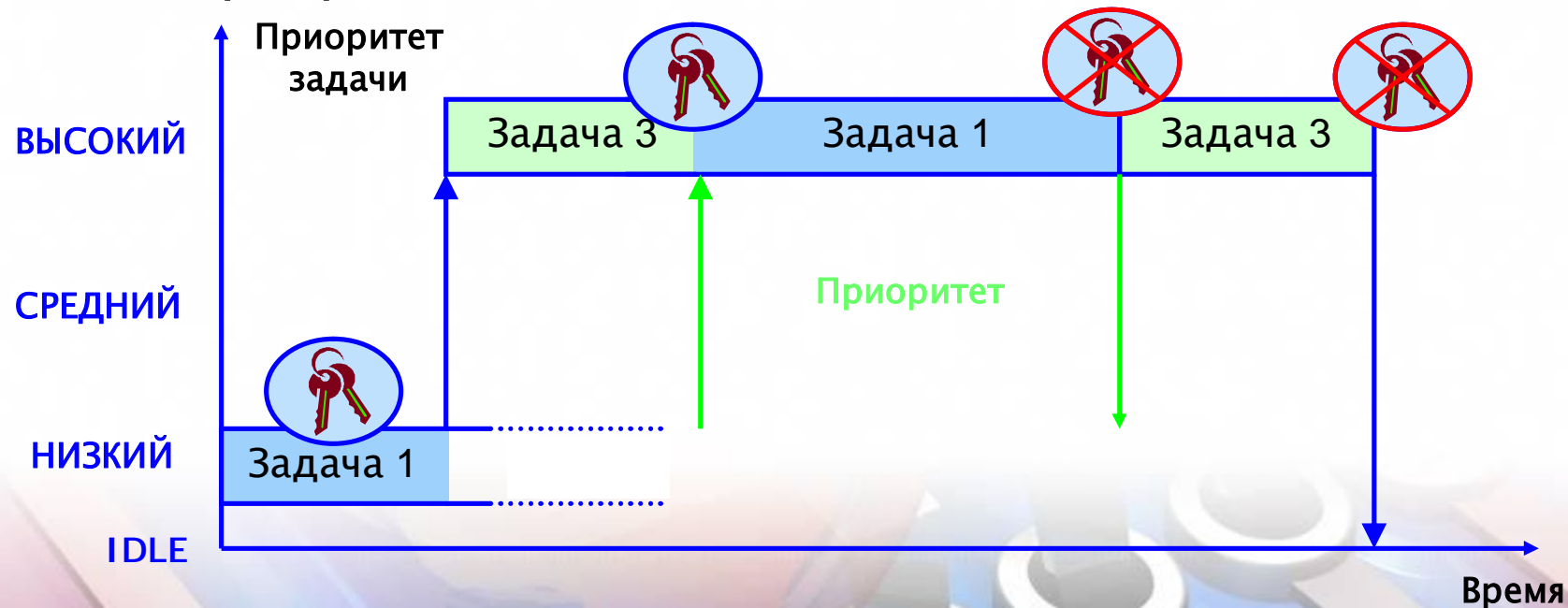
- Инверсия приоритетов возникает, когда более приоритетная задача ждет освобождения ресурса, который использует менее приоритетная задача





# IPC - Mutex

- Полностью исключить инверсию нельзя, можно лишь снизить время ожидания ресурса самой приоритетной задачей
  - Priority Inheritance Protocol – протокол наследования приоритета
  - Priority Ceiling Protocol – протокол предельного приоритета



## IPC - События

- | События – набор флагов
- | В отличие от семафоров проверка состояния флагов не вызывает их изменения



# IPC - События

- I Три основных действия:
  1. Установить флаг события
  2. Сбросить флаг события
  3. Проверить флаг события

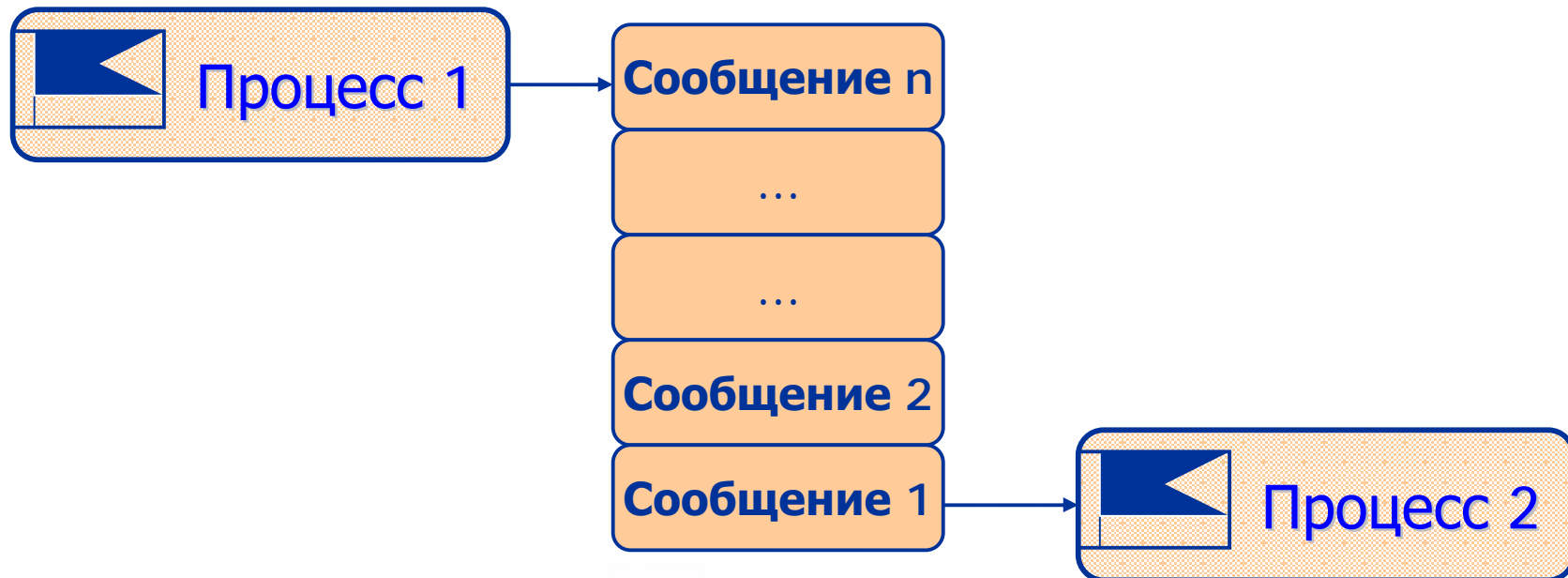
# IPC - Сообщения

- | Сообщения – данные, передаваемые между процессами с использованием специальных сервисов ядра
- | Помимо передачи данных осуществляют синхронизацию процессов



# IPC – Очереди сообщений

- Сообщения могут организовываться в очереди FIFO



# IPC - Сообщения

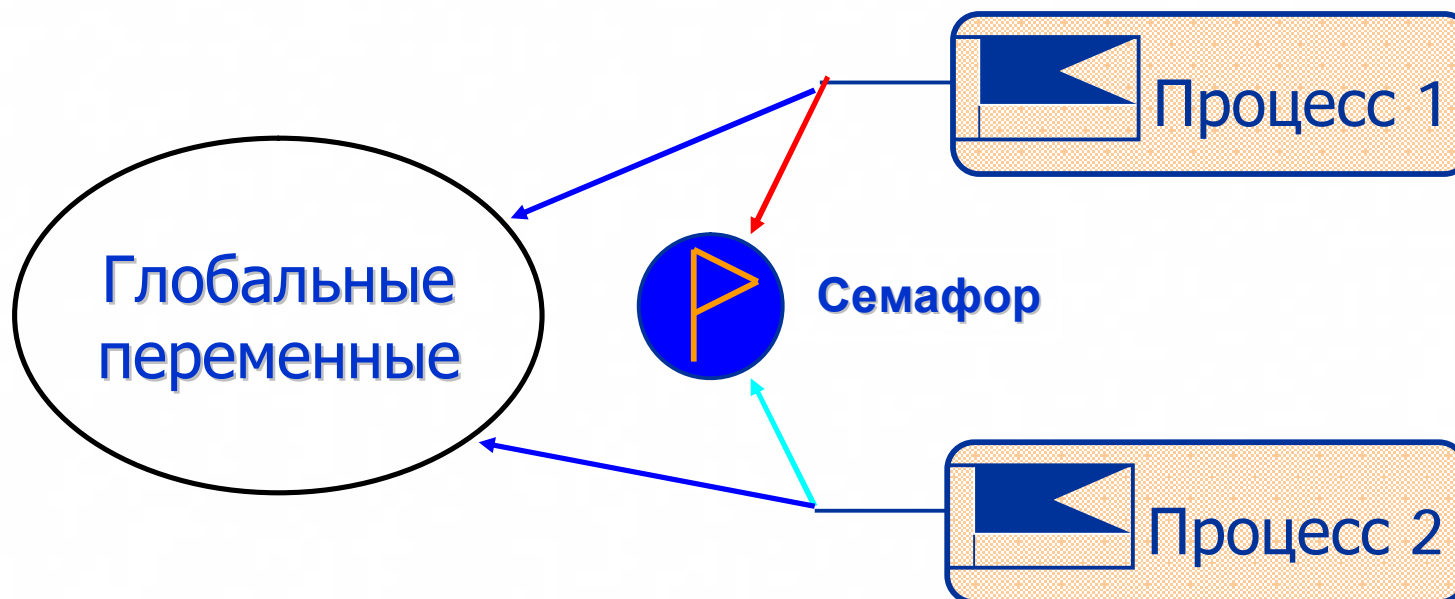
## I Три основных действия:

1. Создать дескриптор сообщения (очереди сообщений)
2. Отправить сообщение
3. Принять сообщение:
  1. Если сообщение есть (очередь не пуста) – считать сообщение и удалить его
  2. Если сообщения нет (очередь пуста) – ждать сообщение

## IPC – Разделяемая память

- | Используется для обмена большими объемами данных
- | Самый простой способ межпроцессного обмена – использование глобальных переменных
- | Проблема синхронизации доступа решается с помощью семафора

# IPC – Разделяемая память





# Обработка прерываний

- | Обработка прерываний производится вне операционной системы
- | Из обработчика прерываний возможен вызов сервисов ядра: установка семафора, отправка событий и сообщений
- | Исключение может составлять обработка прерываний системного таймера

# Обработка прерываний

- | Критические участки кода ядра защищены от прерываний
- | В собственной программе необходимо использовать явный запрет прерываний на критических участках

# Создание приложения с ОСРВ

- | Декомпозиция:
  - | Требования к аппаратному обеспечению
  - | Устройства ввода-вывода
  - | Требования реального времени:
    - | Непериодические события и времена реакций на события
    - | Периодические события, частота периодических событий
- | Задачи
  - | Методы синхронизации задач

# Создание приложения с ОСРВ

- | Система сбора данных
  - | Микроконтроллер
  - | Внешний АЦП (2 канала) с выходом SPI
  - | Клавиатура
  - | ЖКИ
  - | RS-232

## Распределение функций – прерывания

- Для работы с АЦП используем внешнее прерывание PIC INT0. По окончании цикла преобразования посылается сообщение с результатом преобразования
- Обработчик прерывания от USART при приеме посылки формирует сообщение с данными

## Распределение функций – процессы

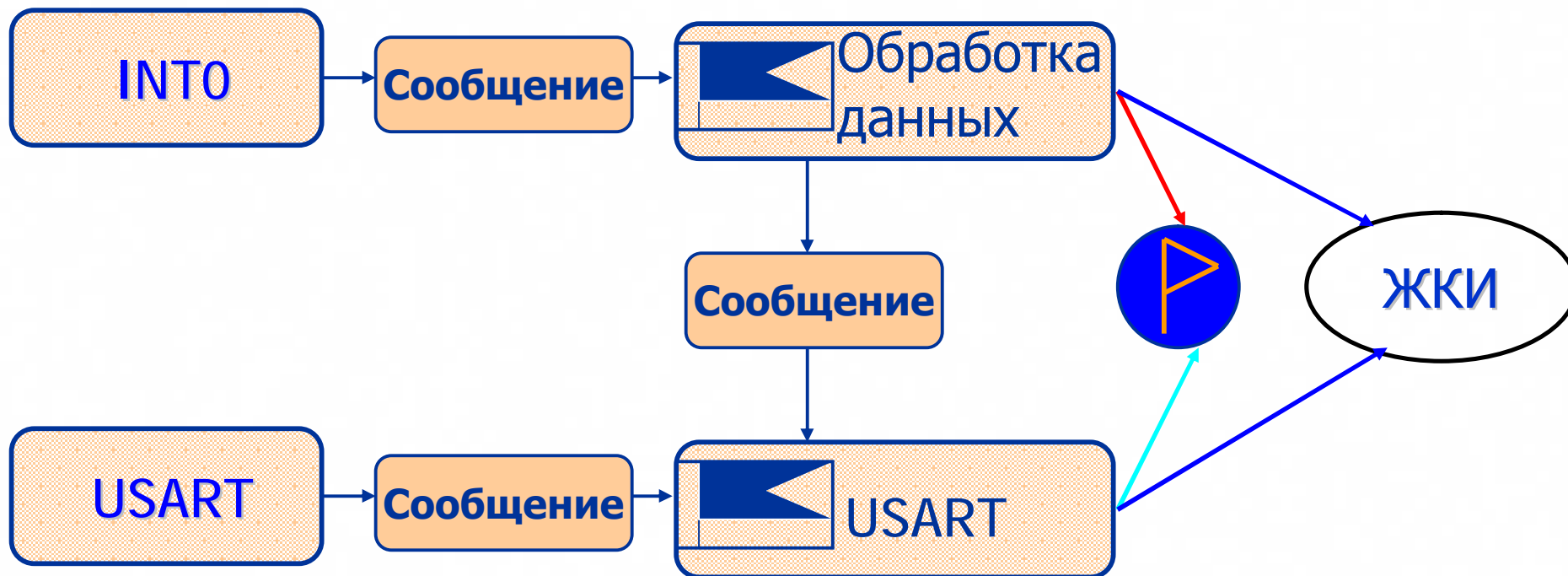
- Процесс обработки данных ждет сообщение с преобразованными данными, по приходу которых производит необходимые вычисления
- Полученные значения передаются процессу обслуживания USART и на ЖКИ

# Распределение функций – процессы

- I К ЖКИ могут обращаться:
  1. Процесс обработки данных
  2. Процесс обслуживания USART

Синхронизация доступа к ЖКИ решается с помощью семафора

# Распределение функций







**YOU + MICROCHIP ENGINEERING THE FUTURE TOGETHER**

# Выводы

# Достоинства ОСРВ

- | Вытесняющая ОСРВ имеет детерминированное время реакции на событие в отличие от однозадачного приложения
- | Каждую задачу можно рассматривать как отдельную программу
- | Микроконтроллеры PIC10, PIC12, PIC16 для кооперативных ОСРВ
- | PIC18, PIC24, dsPIC, PIC32 – для вытесняющих

## С чего начать

- | Руководство по применению jacOS: <http://jacos.narod.ru/>
  - | TNKernel: <http://www.tnkernel.com>
  - | TNKernel для MCHP: <http://www.pic24.ru>
  - | FreeRTOS: [www.FreeRTOS.org](http://www.FreeRTOS.org)
- 
- | Встроенные тесты для обеспечения отказоустойчивости ПО (Class B Safety Software Library) – AN1229

# Спасибо за внимание!

