

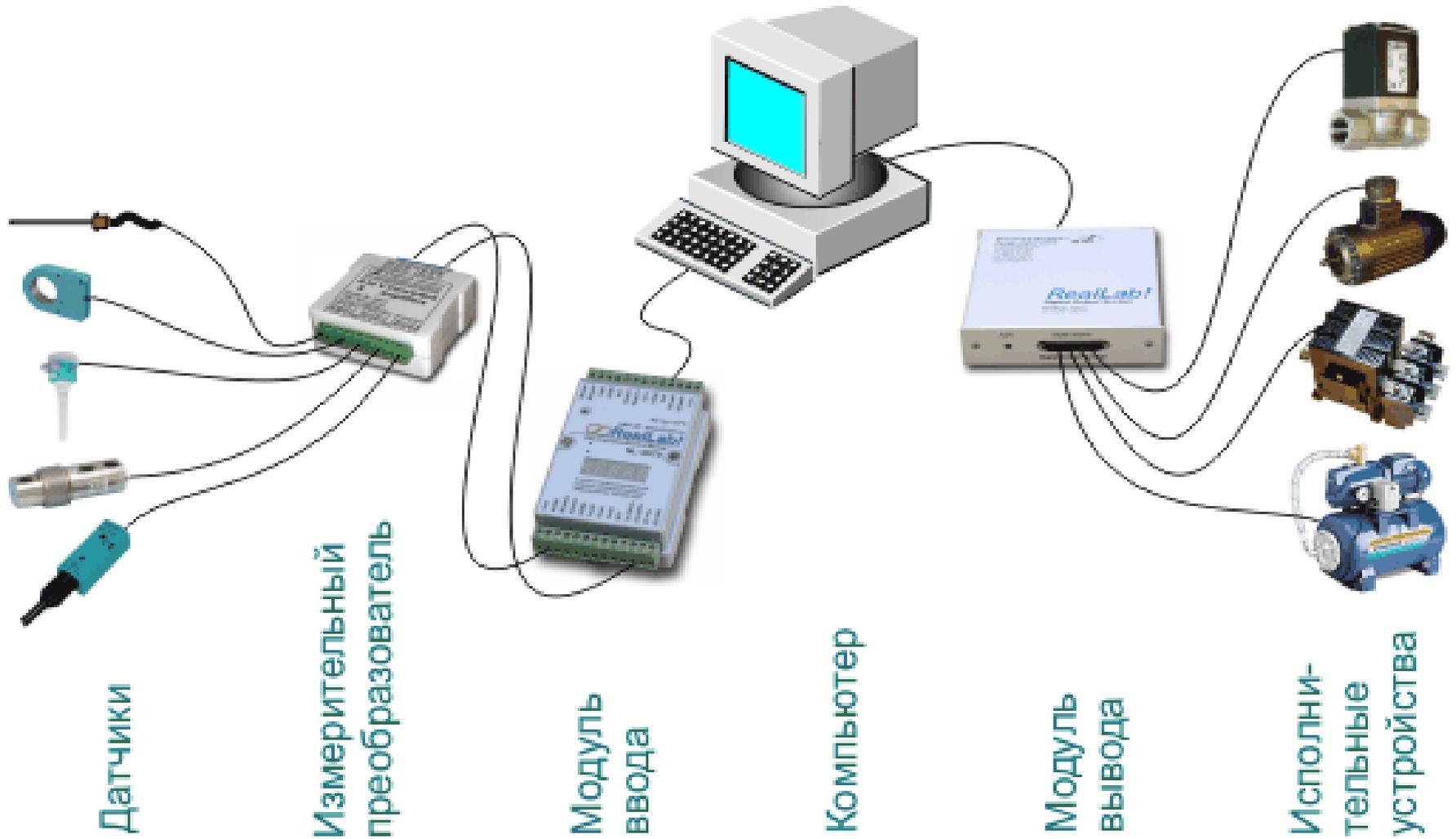
# **Программирование мобильных робототехнических комплексов**

# **Введение в мобильные средства автоматизации**

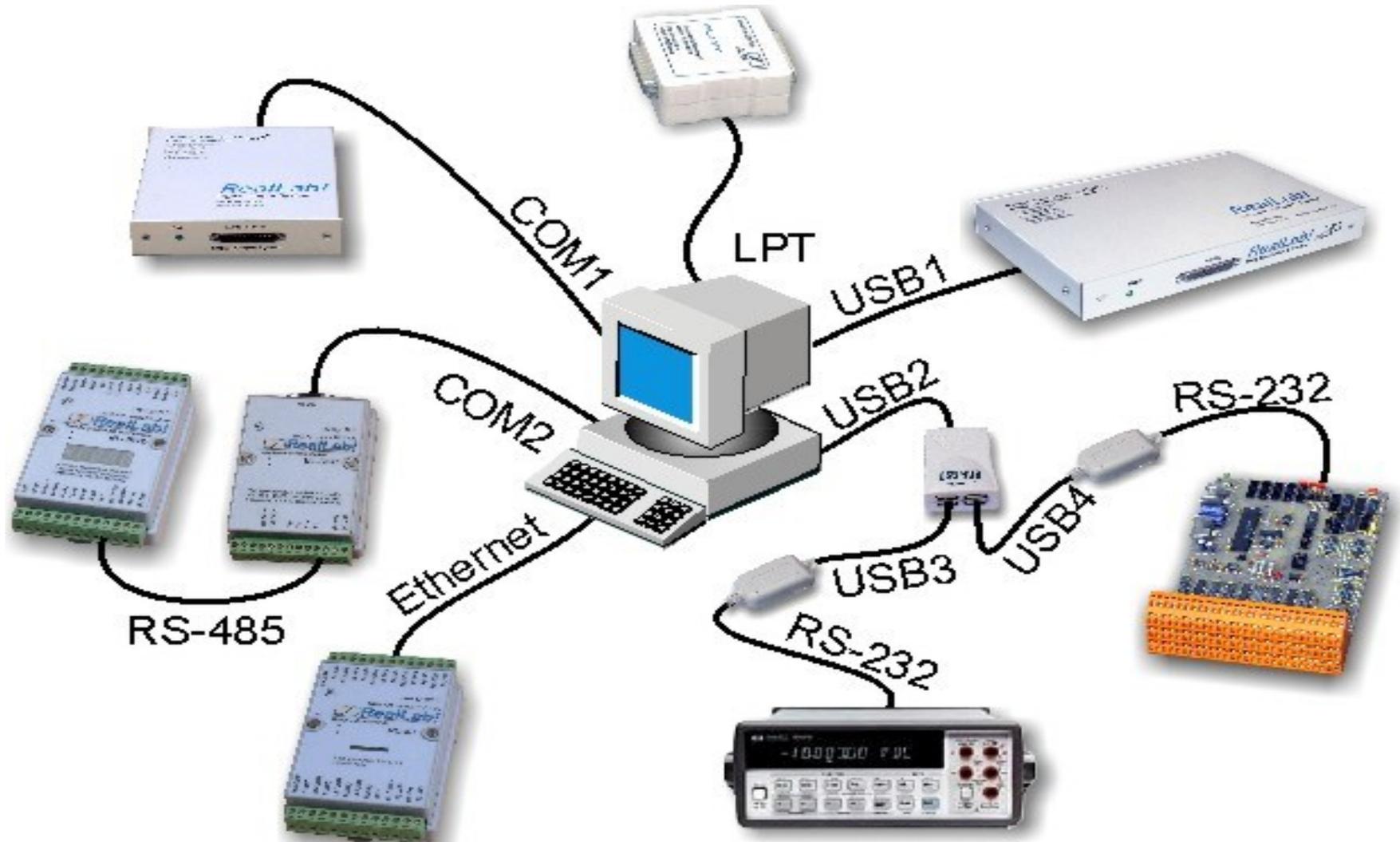
- **Понятие мобильных средств автоматизации (МСА)**
- **Архитектура аппаратной составляющей**
- **Организация взаимодействий, адресация,  
протоколы**
- **Роль МСА в автоматизации современных  
предприятий и производств**

# **Понятие мобильных средств автоматизации (МСА)**

# Архитектура аппаратной составляющей



# Организация взаимодействий, адресация, протоколы



# Протоколы

- **Modbus/RS-485 (EIA-485)**
- **HART**
- **Industrial Ethernet**
- **FOUNDATION Fieldbus HSE**
- **PROFINET (IEC 61158/IEC 61784)**
- **AS-Interface (Actuator Sensor Interface) (EN 50295)**
- **EIB (European Installation Bus) (EN 50090, ANSI EIA 776)**
- **CAN (Controller Area Network )**
- **CANbus**
- **CANopen**
- **DeviceNet**
- **BACnet (Building Automation and Control network)**
- **LON-Works (ANSI/EIA 709.1-A-1999)**

# **Роль МСА в автоматизации современных предприятий и производств**

# **Лингвистическое обеспечение мобильных средств автоматизации**

- **Языки программирования**
- **Языки проектирования**
- **Языки управления**
- **Трансляция**
- **Интерпретация**

# Языковые средства разработки интерфейса, разметки и управления.

## Представление информации



# Языки проектирования



# Технология Java Development Kit

- Назначение технологии Java Development Kit (JDK)
- Концепция кроссплатформенных интерактивных пакетов
- Компилятор `javac`
- Стандартные библиотеки классов Java
- Исполнительная система Java (JRE), виртуальная машина Java Virtual Machine
- Утилиты командной строки

# **Назначение технологии Java Development Kit (JDK)**

**Java Development Kit (сокращенно JDK) — комплект разработчика приложений на языке Java, включающий в себя:**

- компилятор Java (javac),**
- стандартные библиотеки классов Java,**
- примеры,**
- документацию,**
- различные утилиты,**
- исполнительную систему Java (JRE)**

# **Концепция кроссплатформенных интерактивных пакетов**

- **Кроссплатформенные среды исполнения**
- **Кроссплатформенный пользовательский интерфейс**
- **Единый стиль, общий для всех ОС**
- **Самоадаптирующийся интерфейс**

# Компилятор javac

**Javac — оптимизирующий компилятор языка java, включенный в состав многих Java Development Kit (JDK).**

## Структура javac

**parse — лексический и синтаксический анализ, генерация AST,  
enter — регистрация символов всех классов, определенных в программе,  
process annotations — обработка аннотаций,  
attribute — проверка типов, разрешение имен классов, свертка констант,  
вывод типов,  
flow — анализ потока управления (достижимость операций), анализ  
обработки исключений, проверка обращений к неинициализированным  
данным, проверка корректности инициализации final переменных,  
desugar — удаление синтаксического сахара (вложенные классы,  
классовые литералы, assert, foreach),  
generate — создание файла .class.**

# Стандартные библиотеки классов Java

- Встроенные классы
- Замещающие классы
- Подключаемые библиотеки классов
- Библиотека классов `java.util`
- Библиотека классов `java.io`
- Библиотека классов `java.net`
- Библиотека классов `java.awt`
- Библиотека классов `java.awt.image`
- Библиотека классов `java.awt.peer`
- Библиотека классов `java.applet`

# **Исполнительная система Java (JRE), виртуальная машина Java Virtual Machine**

- **Java Runtime Environment, сокращенно JRE – это исполнительная среда Java в которой выполняются программы, написанные на этом языке**
- **Среда состоит из виртуальной машины – Java Virtual Machine(JVM) и библиотеки Java классов.**
- **Виртуальная машина Java исполняет байт-код Java, предварительно созданный из исходного текста Java-программы компилятором Java (javac).**

# УТИЛИТЫ КОМАНДНОЙ СТРОКИ

**javac (или javac.exe)**

**java (или java.exe)**

```
public class Main {  
    public static void main(String[] args) {  
        System.out.println("Hello,  
World!");  
    }  
}
```

# Технология Eclipse

- Назначение технологии Eclipse
- Архитектура Eclipse, ядро платформы
- Рабочая среда
- Портируемый инструментарий виджетов
- Промежуточный слой GUI JFace(файловые буферы, работа с текстом)
- Поддержка архитектур и систем Linux, Mac OS X, Windows, Android

# Назначение технологии Eclipse

- **Eclipse — свободная интегрированная среда разработки модульных кроссплатформенных приложений.**
- **Развивается и поддерживается Eclipse Foundation**

# Архитектура Eclipse, ядро платформы

- Eclipse-плагины Rich Client Platform (RCP)
- Eclipse-редакторы
- Eclipse-представления

Платформа RCP включает в себя такие компоненты как:

- Среду выполнения на основе OSGi,
- Библиотеки SWT и JFace,
- Графическую многооконную Workbench-среду
- Связанные с ней компоненты.

Набор компонентов Eclipse-платформы:

- Platform Runtime Core
- Resource Management
- Workbench UI
- Team support
- Debug support
- Help System