

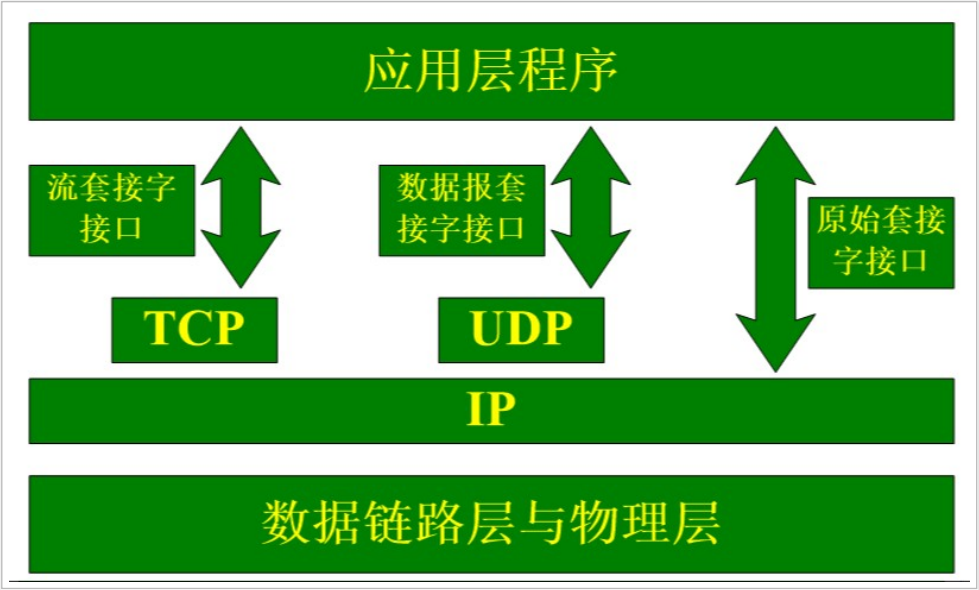
## 7.1 UDP 编程之基础知识\_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 7.1 UDP 编程之基础知识涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

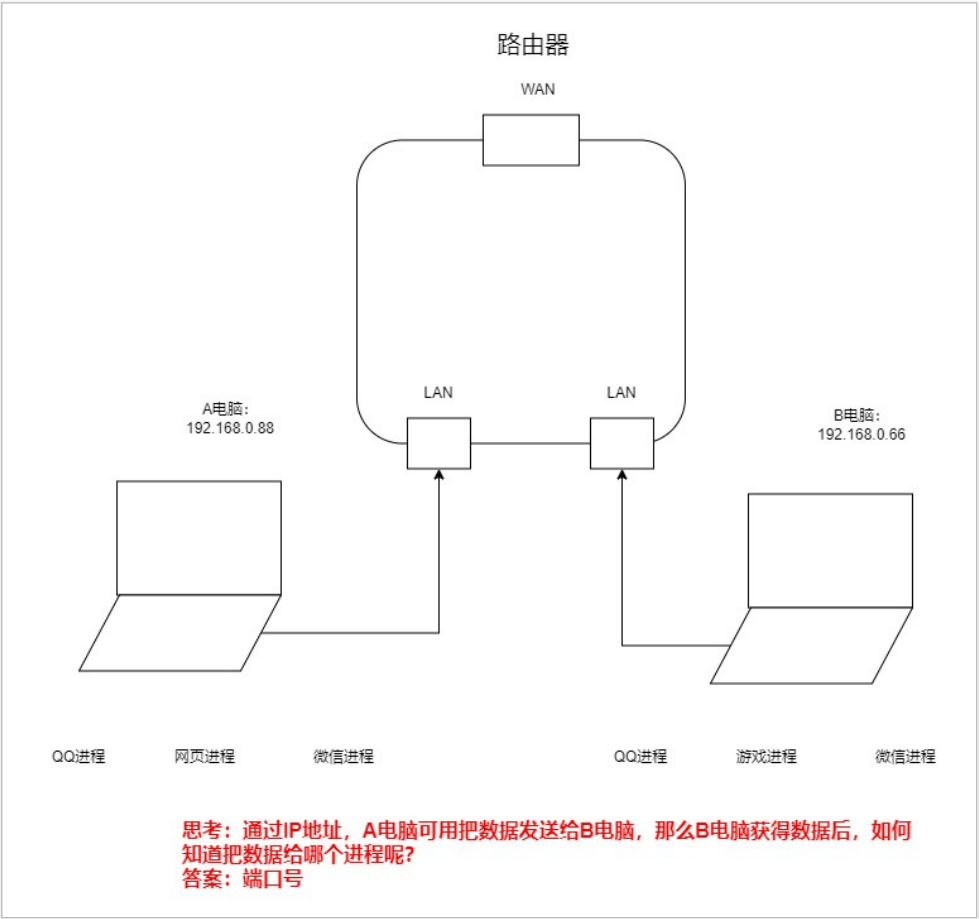
TCP/IP 五层网络模型的 \*\* 应用层编程接口称为 Socket API，\*\*Socket(套接字) 本身有 "插座" 的意思，它是对网络中不同主机上的应用进程之间进行双向通信的端点的抽象。一个套接字就是网络上进程通信的一端，提供了应用层进程利用网络协议交换数据的机制。从所处的地位来讲，套接字上联应用进程，下联网络协议栈，是应用程序通过网络协议进行通信的接口。

socket 可以看成是两个网络应用程序进行通信时，各自通信连接中的端点，这是一个逻辑上的概念。

- 为什么需要 Socket
  - 普通的 I/O 操作
    - 打开文件 -> 读 / 写操作 -> 关闭文件
  - 网络通信
    - TCP/IP 协议被集成到操作系统的内核中，引入了新型的 "I/O" 操作，本质为内核借助缓冲区形成的伪文件
    - 既然是文件，那么我们可以使用文件描述符引用套接字。与管道类似，Linux 系统将其封装成文件的目的是为了统一接口，使得读写套接字和读写文件的操作一致。
- Socket 套接字类型
  - 流式套接字 (SOCKET\_STREAM)
    - 提供了一个面向连接、可靠的数据传输服务，数据无差错、无重复的发送且按发送顺序接收。内设置流量控制，避免数据流淹没慢的接收方。数据被看作是字节流，无长度限制。
  - 数据报套接字 (SOCK\_DGRAM)
    - 提供无连接服务。数据包以独立数据包的形式被发送，不提供无差错保证，数据可能丢失或重复，顺序发送，可能乱序接收。
  - 原始套接字 (SOCK\_RAW)
    - 可以对较低层次协议如 IP、ICMP 直接访问。
- Socket 的位置



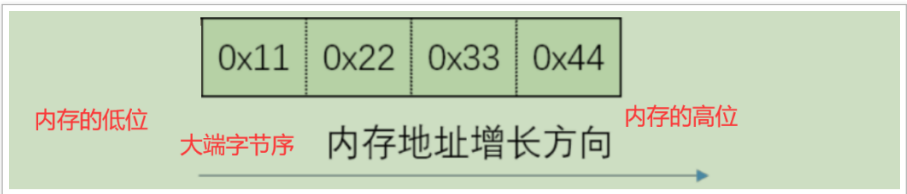
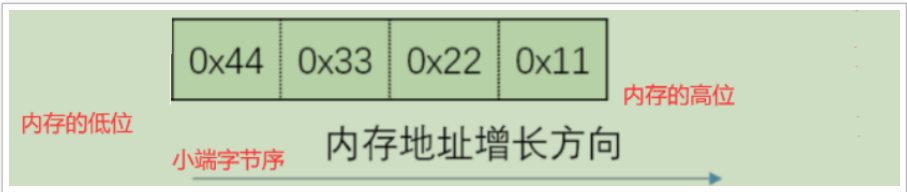
网络通信的本质是不同主机，不同进程之间的通信。



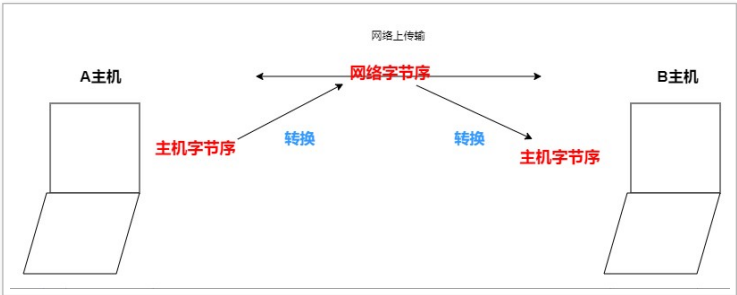
两个不同主机，不同进程之间的通信。A 主机需要知道 B 主机的 IP 地址 + 端口号。

- 简介
  - Internet 中的主机要与别的机器通信必须具有一个 IP 地址
  - IPV4 中 IP 地址为 32bit
  - 每个数据包都必须携带目的 IP 地址和源 IP 地址，路由器依靠此信息为数据包选择路由
- 思考

- 问题：我们每个人电脑有不同的 CPU，不同 CPU 对数据的存储方式不同，分为大端模式和小端模式，那么不同的 CPU 数据的存储方式不同，如何定义网络数据流的存储方式呢？
- 常见 CPU 的存储
  - 小端模式的 CPU：intel, mips 等。
  - 大端模式的 CPU：ARM, 摩托罗拉等
- 大端和小端
  - 小端模式：低地址存储数据的低位，高地址存储数据的高位。
  - 大端模式：高地址存储数据的低位，低地址存储数据的高位。
- unsigned int data = 0x11223344;



- 解决方案
- 为了统一标准，TCP/IP 协议规定，网络数据流采用大端字节序 (低地址存储高字节位).
- 不同主机网络传输方法



- A----->B，需要提前知道 B 主机的 ip 地址 + 端口。
- 简介
  - 为了区分一台主机接收到的数据包应该转交给哪个进程来进行处理，使用端口号来区分。
  - TCP 端口号与 UDP 端口号独立
  - 端口号一般由 IANA (Internet Assigned Numbers Authority, 互联网数字分配机构) 管理

- 端口分类

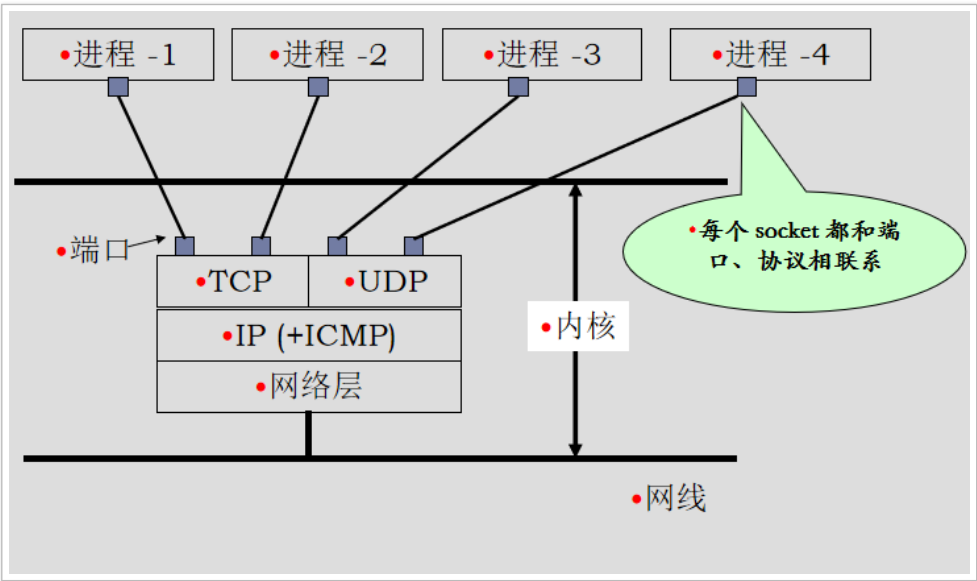
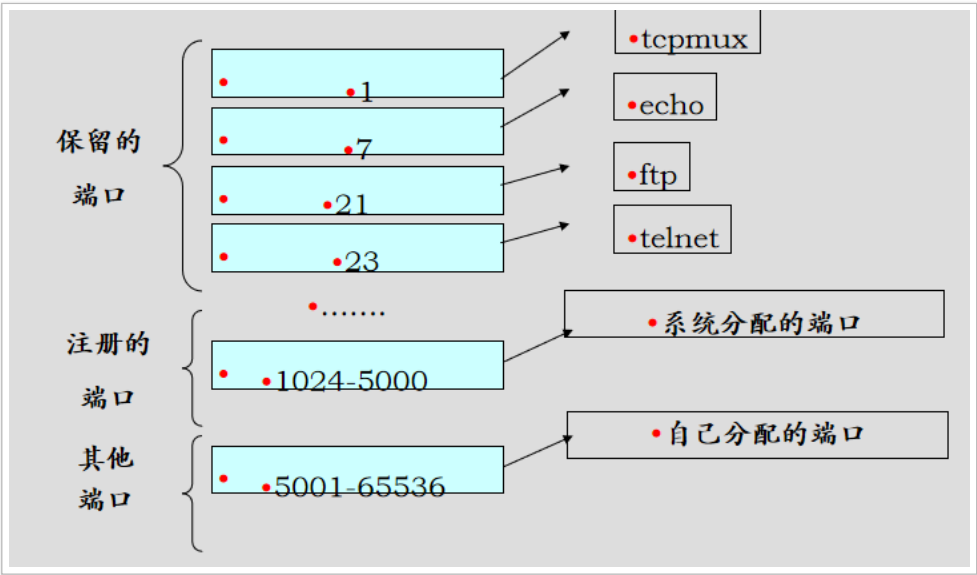
众所周知的端口：1 ~ 1023

已经登记的端口：1024~49151

动态或私有端口：49152~65535

- 已经公用的端口号
  - FTP 协议：21/tcp
  - ssh 安全登录 / scp 文件传输：22/tcp

- HTTP 协议：80 /tcp
- Telnet 协议：23 / tcp
- SMTP 协议：25/tcp
- Tftp 协议：69



练习：

请大家默写出 HTTP 协议，telnet 协议对应的端口号

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，点击查看详细说明

