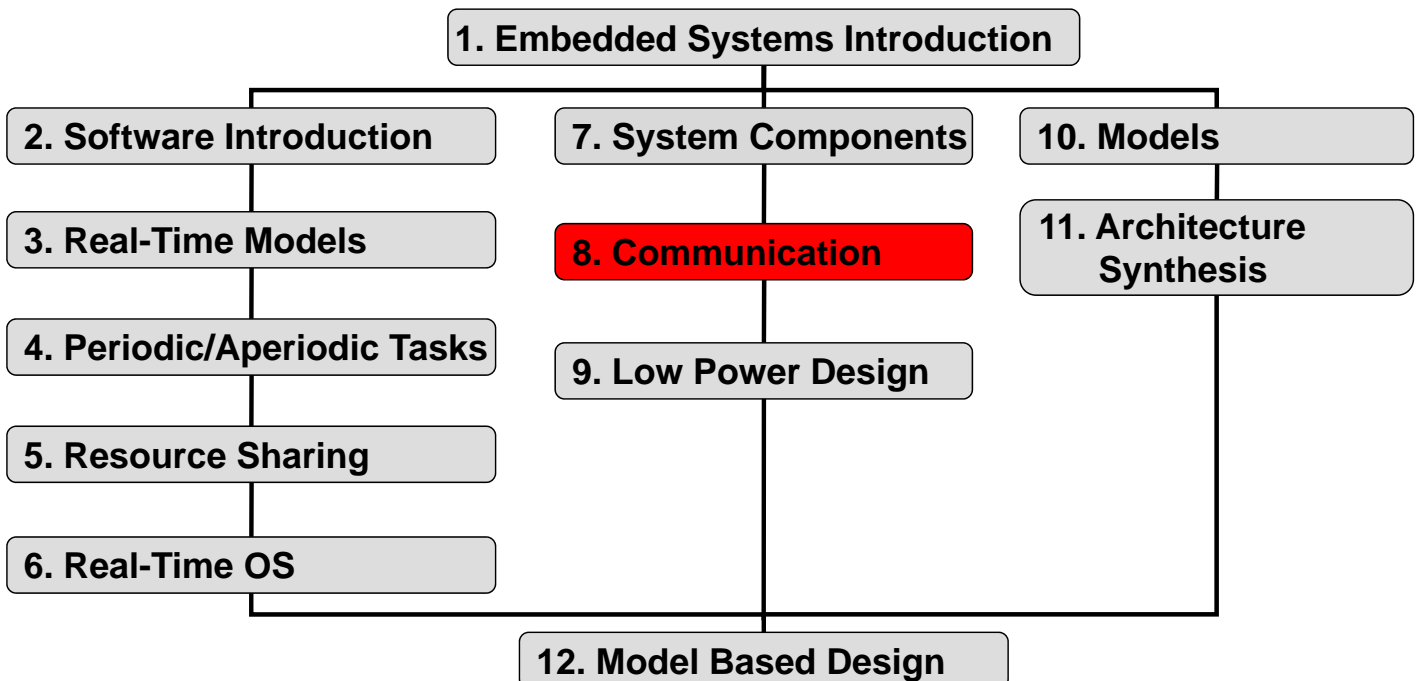


Embedded Systems

8. Communication

Lothar Thiele

Contents of Course



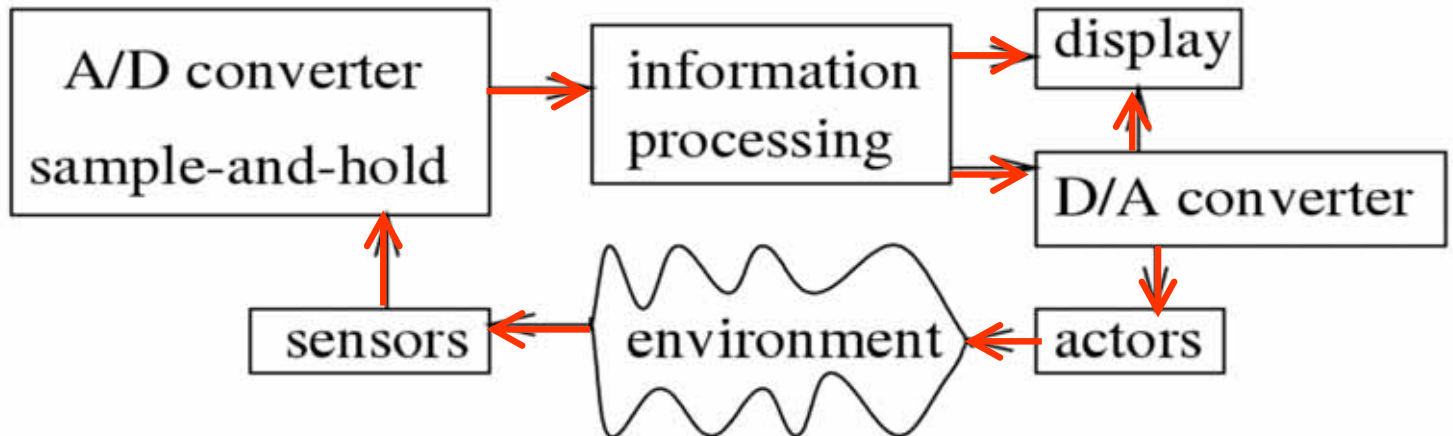
*Software and
Programming*

*Processing and
Communication*

Hardware

Communication

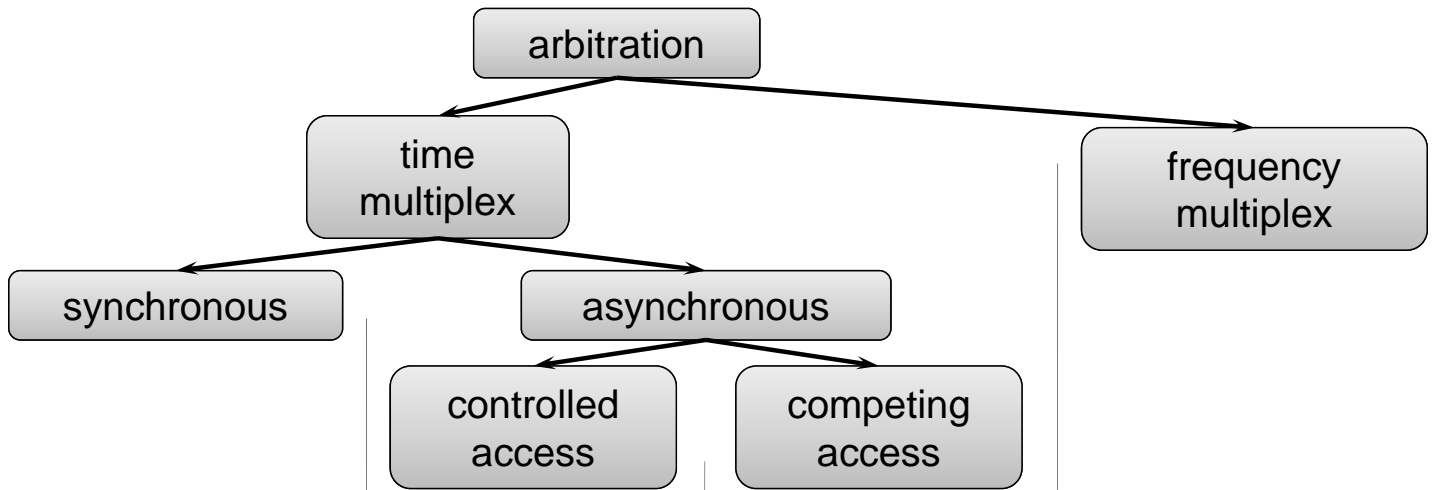
is everywhere ...



Communication: Requirements

- Performance
 - bandwidth and latency
 - guaranteed behavior (real-time)
- Efficiency
 - cost (material, installation, maintenance)
 - low power
- Robustness
 - fault tolerance
 - maintainability, diagnoseability
 - security, safety

Protocol Classification



covered in lecture

TDMA

CSMA/CR/CA
Token Bus/Ring

CSMA/CD
Random

Bluetooth

FlexRay

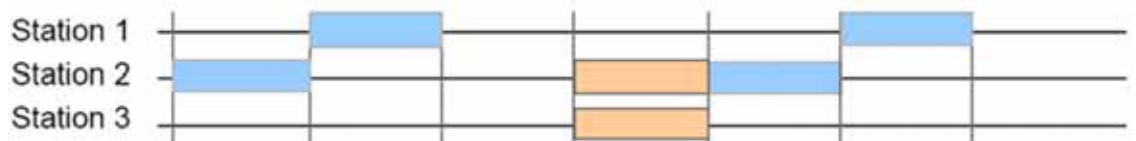
Random Access

▶ Random access to communication medium

- no access control; requires low medium utilization



- improved variant: slotted random access

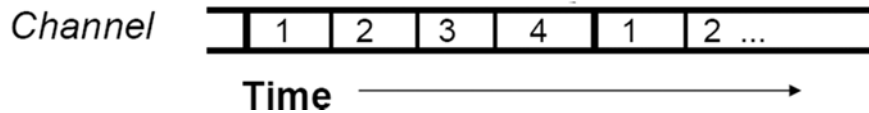


- What is the optimal sending rate p in case of n stations?

- probability that a slot is not taken by others: $(1 - p)^{n-1}$
- probability that a station transmits successfully: $P = p \cdot (1 - p)^{n-1}$
- determine maximum with respect to p : $dP/dp = 0 \rightarrow p = 1/n$

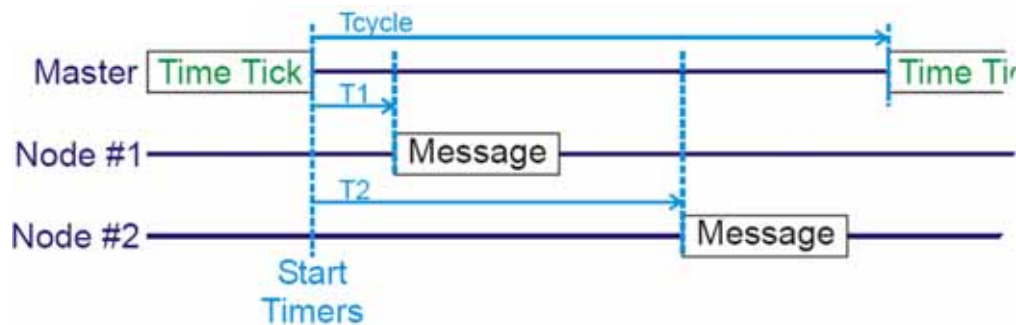
TDMA

- ▶ Communication in statically allocated time slots
- ▶ Synchronization among all nodes necessary:
 - periodic repetition of communication frame or



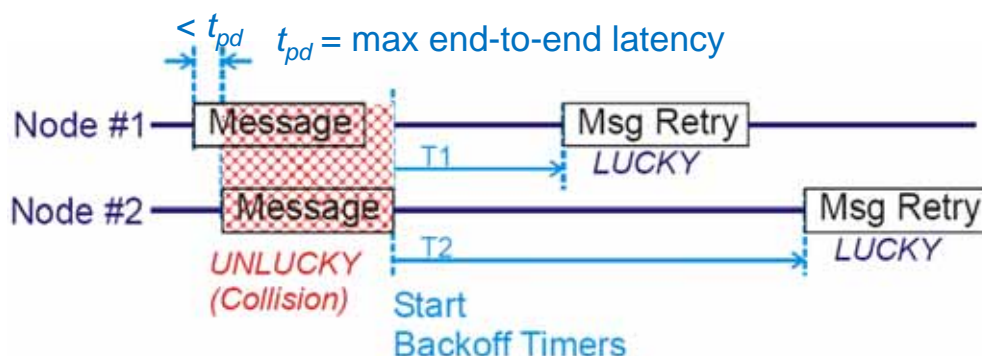
- master node sends out a synchronization frame

- ▶ Examples: TTP, static portion of FlexRay, satellite networks



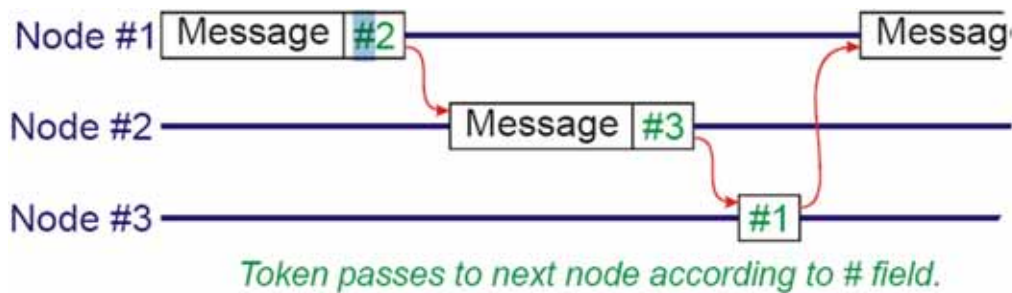
CSMA/CD

- ▶ Carrier Sense Multiple Access / Collision Detection
- ▶ Try to avoid and detect collisions:
 - before starting to transmit, check whether the channel is idle
 - if a collision is detected (several nodes started almost simultaneously), wait for some time (backoff timer)
 - repeated collisions result in increasing backoff times
- ▶ Examples: Ethernet, IEEE 802.3

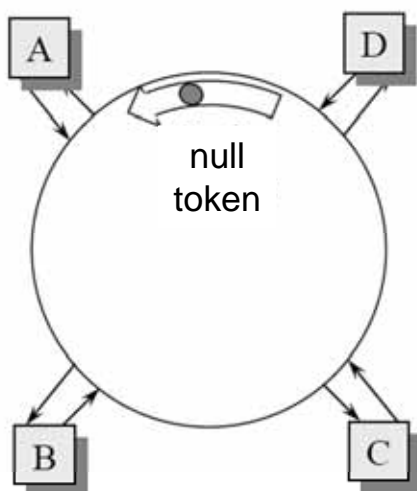


Token Protocols

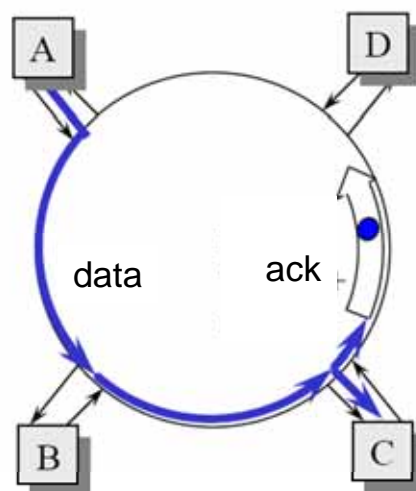
- ▶ Token value determines which node is transmitting and/or should transmit next
 - Only the token holder may transmit
 - Master/slave polling is a special form
 - Null messages with tokens must be passed to prevent network from going idle
- ▶ Examples: IEEE 802.4, Profibus, TokenRing



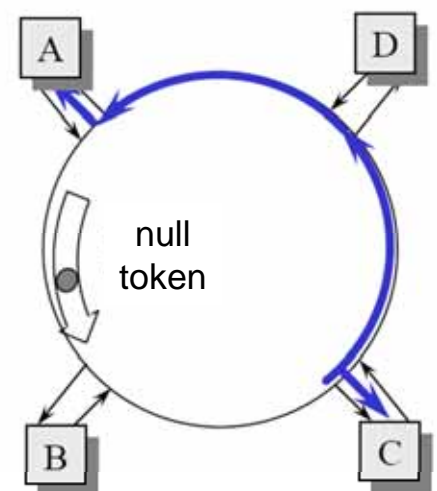
Token Ring



A requests transmission



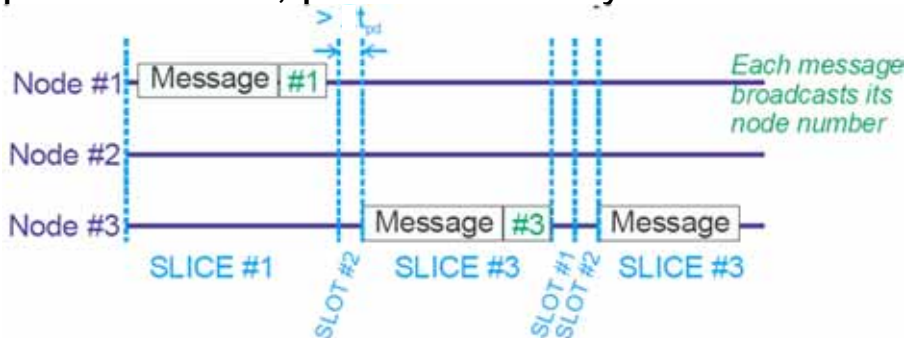
A is token owner
A sends data to C
C sends acknowledge



A sends null token

CSMA/CA – Flexible TDMA (FTDMA)

- ▶ Carrier Sense Multiple Access / Collision Avoidance
- ▶ Operation:
 - reserve s **slots** for n nodes; note: slots are normally idle – they are (short) time intervals, not signals; if slot is used it becomes a **slice**.
 - nodes keep track of global communication state by sensing
 - nodes start transmitting a message only during the assigned slot
 - If $s=n$, no collisions; if $s < n$, statistical collision avoidance
- ▶ Examples: 802.11, part of FlexRay

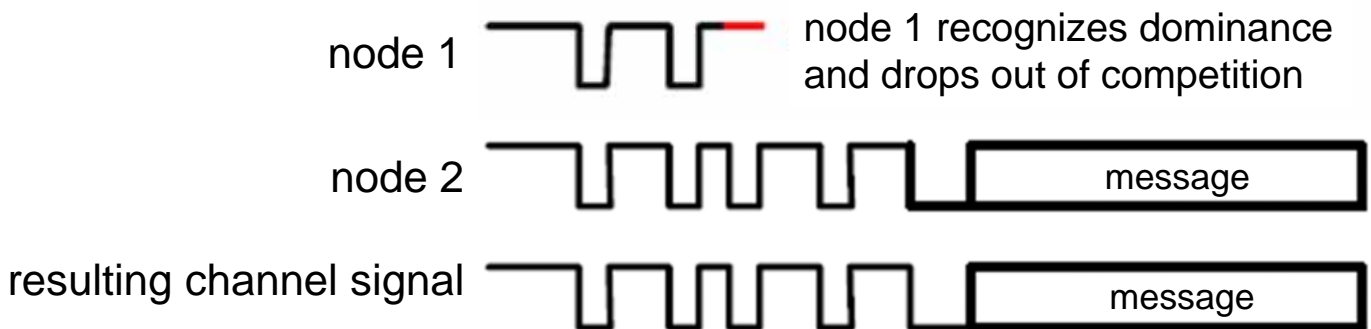
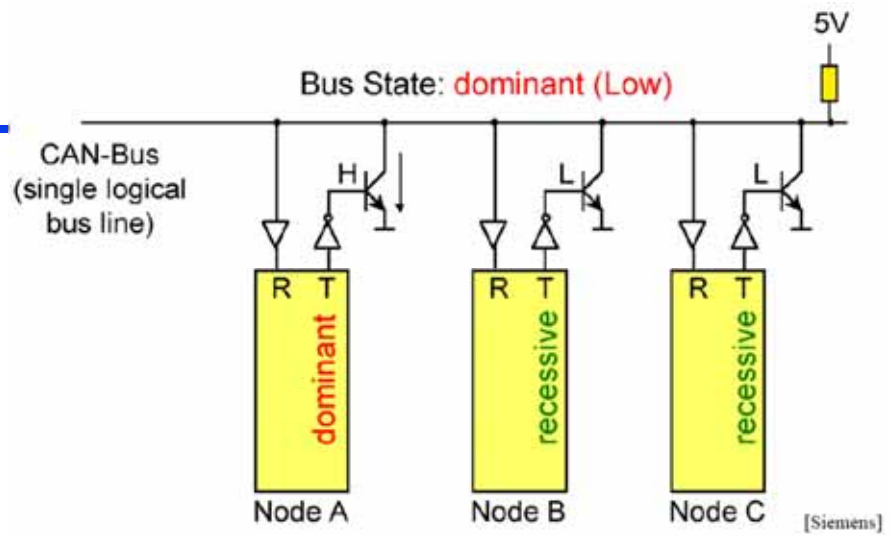


CSMA/CR

- ▶ Carrier Sense Multiple Access / Collision Resolution
 - ▶ Operation:
 - Before any message transmission, there is a global arbitration
- | | | | |
|-------------|---------|-------------|---------|
| arbitration | message | arbitration | message |
|-------------|---------|-------------|---------|

→
- Each node is assigned a unique identification number
 - All nodes wishing to transmit compete by transmitting a binary signal based on their identification value
 - A node drops out the competition if it detects a dominant state while transmitting a passive state
 - Thus, the node with the lowest identification value wins
- ▶ Example: CAN Bus

CSMA/CR



FlexRay

FlexRay:

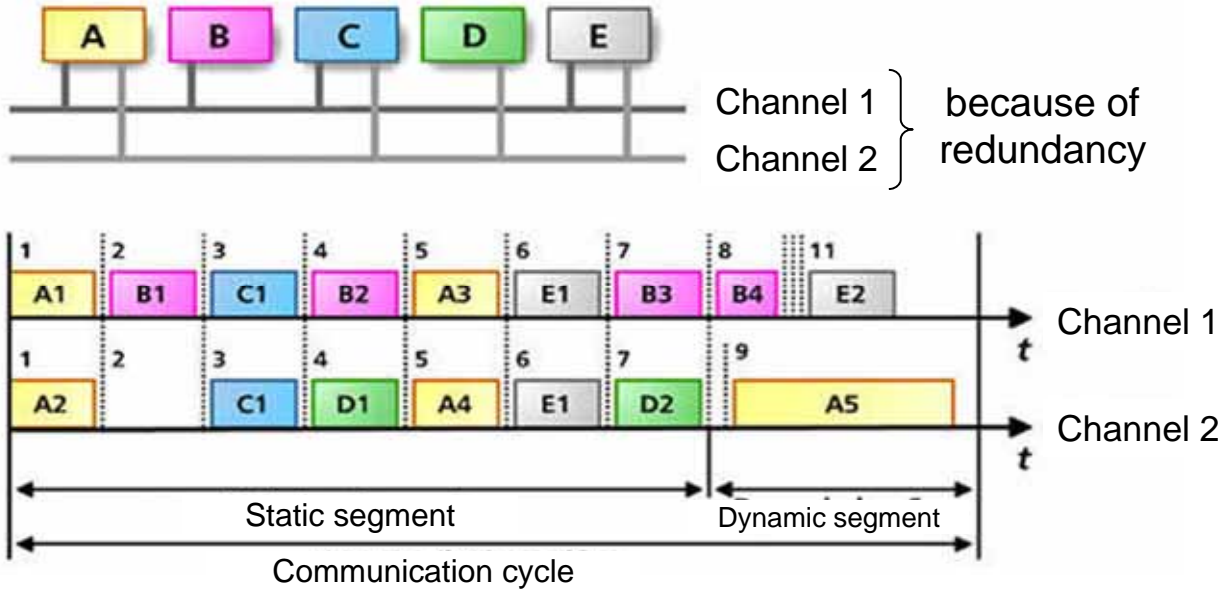
- Developed by the FlexRay consortium (BMW, Ford, Bosch, DaimlerChrysler, General Motors, Motorola, Philips).
- Combination of a TDMA and the Byteflight [Byteflight Consortium, 2003] (Flexible TDMA, close to CSMA/CA) protocol.
- High data rates can be achieved:
 - initially targeted for ~ 10Mbit/sec;
 - design allows much higher data rates

Operation principle:

- Cycle is subdivided into a static and a dynamic segment.
- Static segment is based on a fixed allocation of time slots to nodes.
- Dynamic segment for transmission of ad-hoc communication with variable bandwidth requirements.

FlexRay

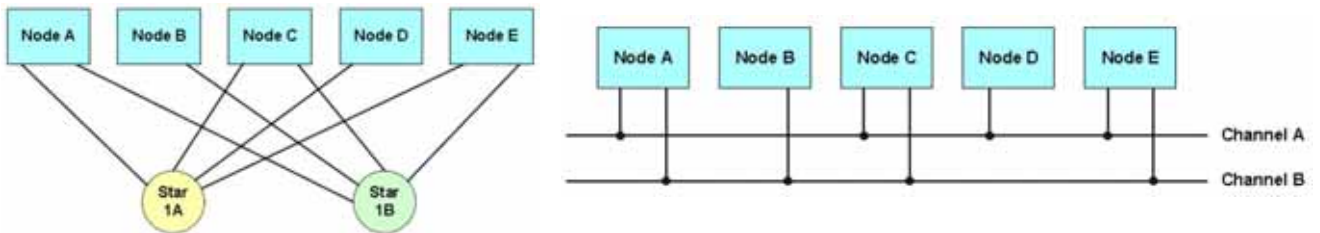
- Use of two independent channels to eliminate single-point failures



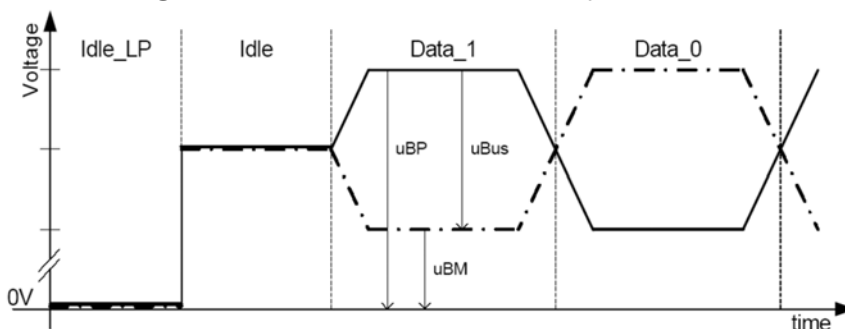
[http://www.ixxat.de/english/produkte/flexray/flexray_introduction.shtml]

FlexRay

- Basic topologies (any combination also possible):



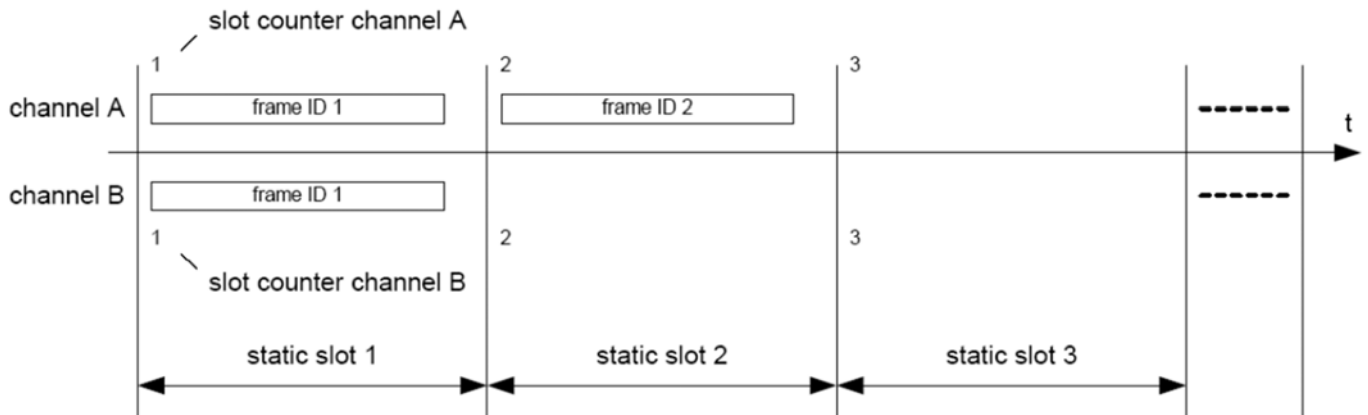
- Typical physical layer (twisted pairs and differential encoding to reduce sensitivity to electromagnetic coupling):



FlexRay

▶ TDMA

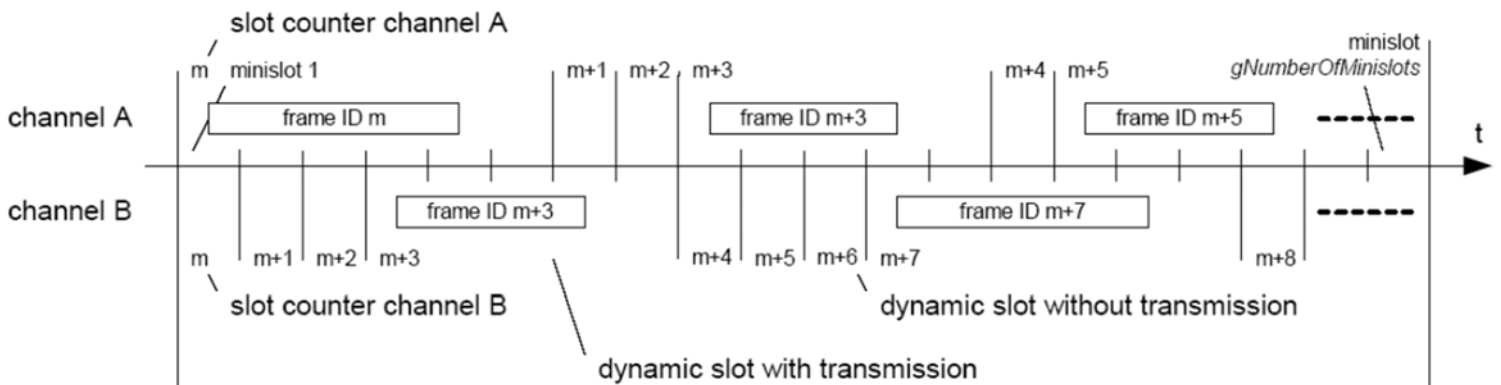
- all static slots are the same length whether used or not
- all slots are repeated in order every communication cycle
- slots are lock-stepped in order on both channels



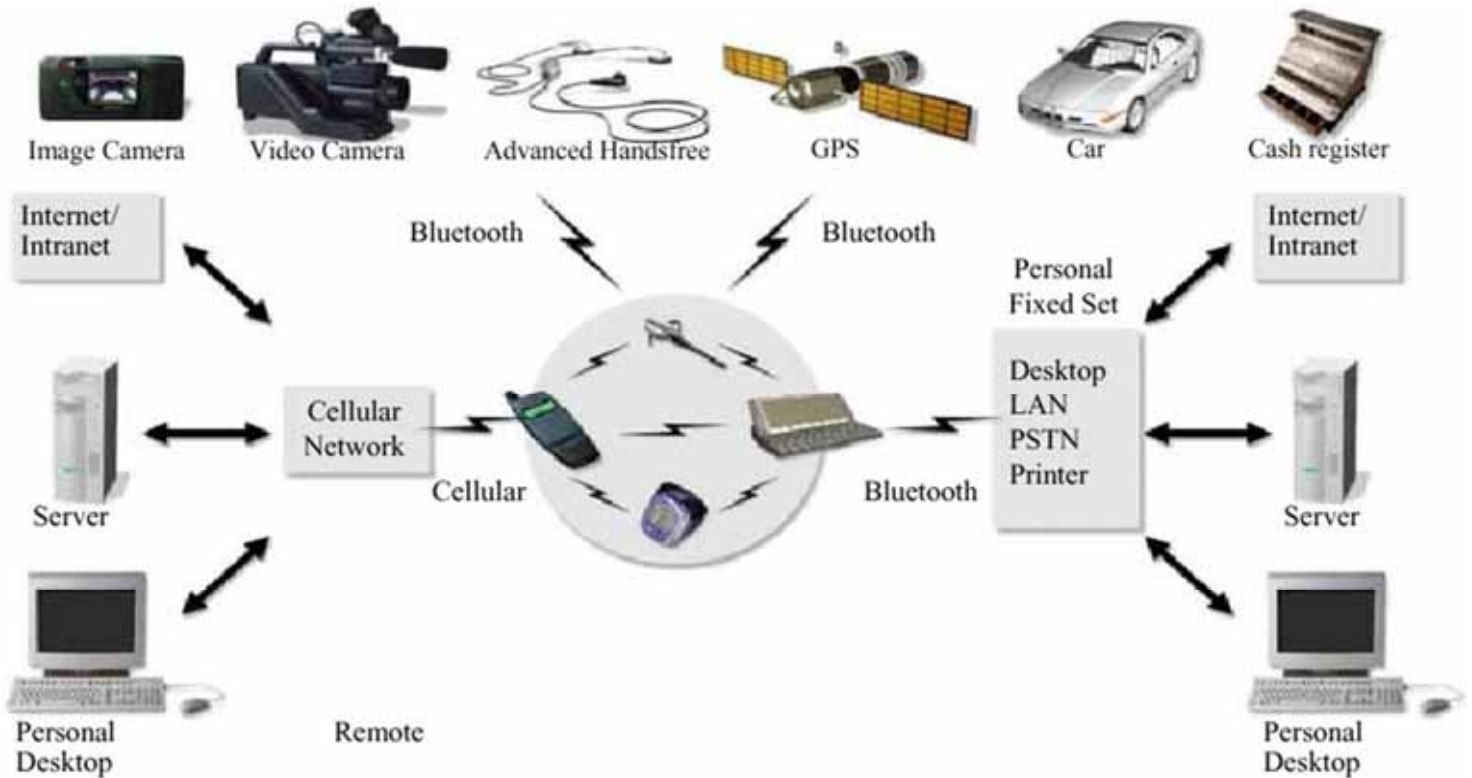
FlexRay

▶ Flexible TDMA:

- each minislot is an **opportunity** to send a message
- if message isn't sent, minislot elapses unused (short idle period)
- all nodes watch whether a message is sent so they can count minislots



Example Bluetooth



Who was Bluetooth?

- ▶ Wiking
- ▶ King of Denmark 940-981
- ▶ Christianized, unified and controlled Denmark and Norway



Bluetooth Overview

► *Design Goals*

- small size, low cost, low energy
- secure transmission (cryptography, authentication)
- robust transmission (interference with wireless LAN)

► *Technical Data*

- 2.4 GHz Band (open band, spectral bandwidth 79 MHz, frequency hopping and time multiplex)
- 10-100 m transmission range, 1 Mbit/s bandwidth for each connection
- simultaneous transmission of multimedia streams (synchronous) and data (asynchronous)
- ad hoc network (spontaneous connections to neighbor nodes, dynamic network topologies, no centralized coordination, multi-hop communication)

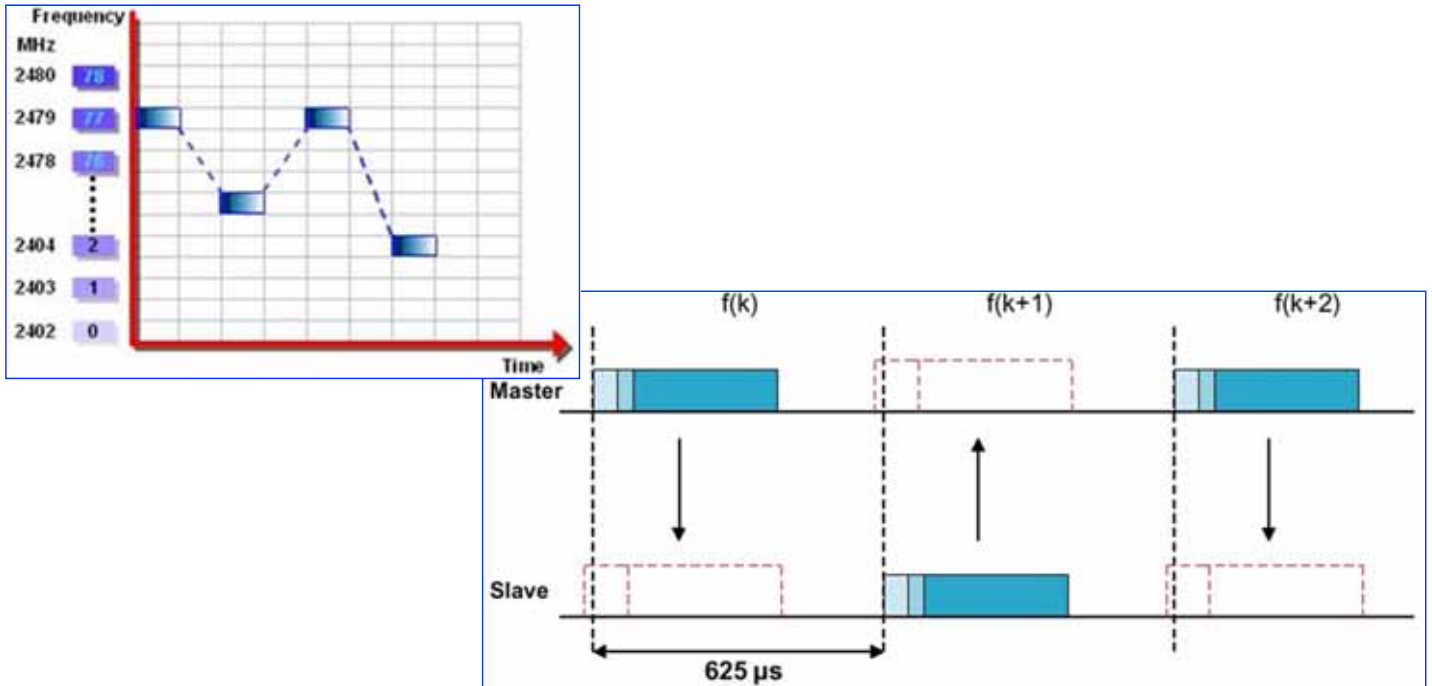
Bluetooth Overview

► *Frequency Hopping*

- Transmitter jumps from one frequency to another with a fixed rate (1600 hops/s). The ordering (channel sequence) is determined by a pseudo random sequence of length $2^{27}-1$.
- Frequency range (2402 + k) MHz, k = 0 ...78 .
- The data transmission is partitioned into time windows of length 0.625 ms; each packet is transmitted by means of a different frequency.

Bluetooth Overview

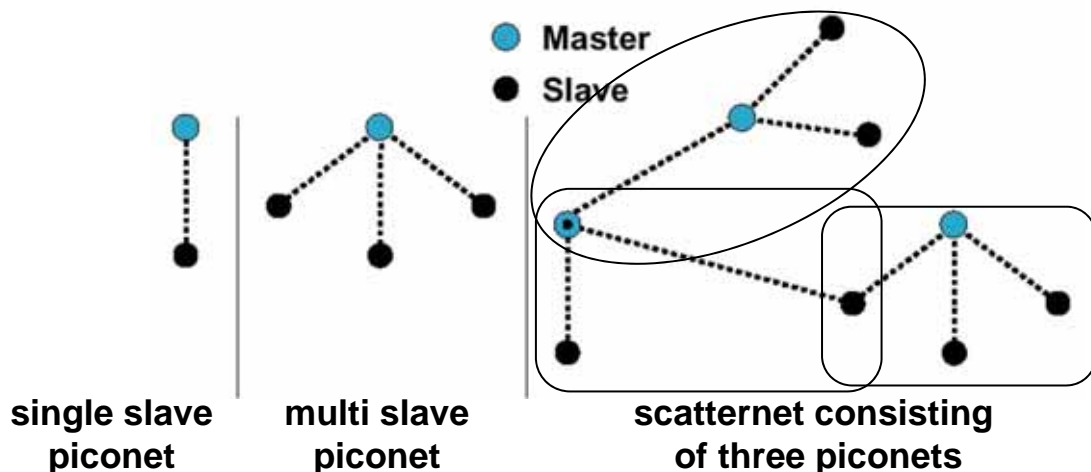
Example frequency hopping:



Network Topologies

► Ad-hoc networks

- all nodes are potentially mobile
- dynamic emergence of connections
- hierarchical structure (scatternet) of small nets (piconet)



TECHNOLOGY

'Master' and 'slave' computer labels unacceptable, officials say

Wednesday, November 26, 2003 Posted: 3:24 PM EST (2024 GMT)

LOS ANGELES, California (Reuters) -- Los Angeles officials have asked that manufacturers, suppliers and contractors stop using the terms "master" and "slave" on computer equipment, saying such terms are unacceptable and offensive.

The request -- which has some suppliers furious and others busy re-labeling components -- came after an unidentified worker spotted a videotape machine carrying devices labeled "master" and "slave" and filed a discrimination complaint with the county's Office of Affirmative Action Compliance.

In the computer industry, "master" and "slave" are used to refer to primary and secondary hard disk drives. The terms are also used in other industries.

"Based on the cultural diversity and sensitivity of Los Angeles County, this is not an acceptable identification label," Joe Sandoval, division manager of purchasing and contract services, said in a memo sent to County vendors.

"We would request that each manufacturer, supplier and contractor review, identify and remove/change any identification or labeling of equipment components that could be interpreted as discriminatory or offensive in nature," Sandoval said in the memo, which was distributed last week and made available to Reuters.

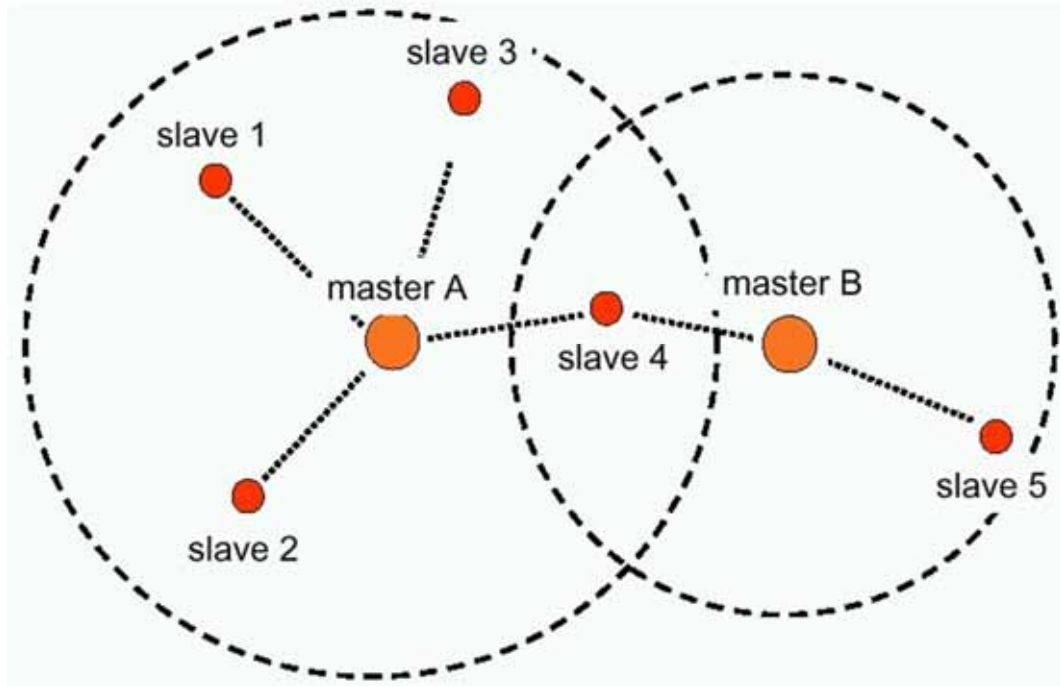
Network Topologies

► Piconet

- A piconet contains 1 master and maximally 7 slaves
- All nodes in a piconet use the same frequency hopping scheme (channel sequence) which is determined by
 - the device address of the master BD_ADDR and
 - phase which is determined by the system clock of the master.
- Connections are either one-to-one or between the master and all slaves (broadcast).
- The following connection types are possible:
 - 432 kBit/s (duplex) or 721/56 kBit/s (asymmetric) or
 - 3 audio channels or
 - a combination of data and audio.

Netzwerktopologien

► Scatternet

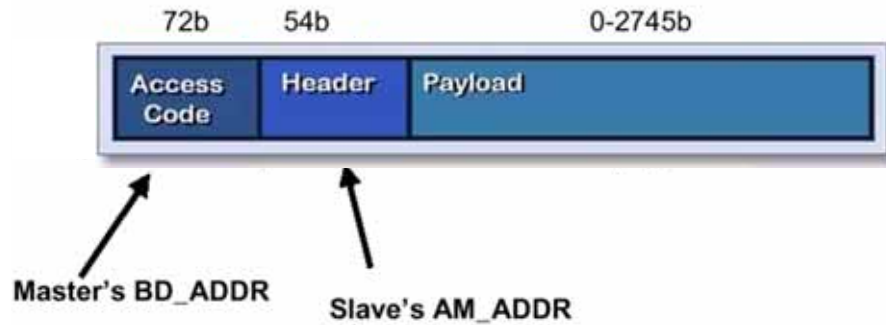


Network Topologies

► Scatternet

- Several piconets with overlapping nodes form a scatternet.
- A node can simultaneously have the roles of slaves in several piconets and the role of a master in at most one piconet.
- The channel sequences of the different piconets are not synchronized.
- As a result, large network structures can emerge and multi-hop communication is possible.

Packet Format



The access code identifies all packets between Bluetooth devices.

Packet Header identifies and characterizes connection between master and slave.

Addressing

- ▶ **Bluetooth Device Address: BD_ADDR**
 - 48 Bit
 - Unique address for each device

- ▶ **Active Member Address AM_ADDR**
 - 3 Bit for maximally 7 active Slaves in a piconet.
 - Address "Null" is a broadcast to all slaves.

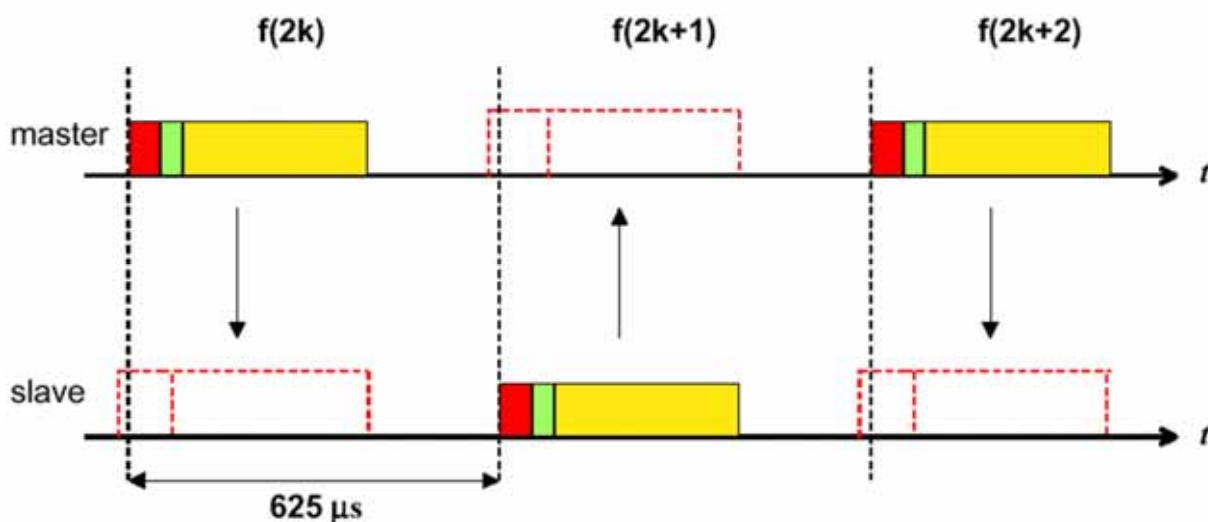
- ▶ **Parked Member Address PM_ADDR**
 - 8 Bit for parked slaves.

Connection Types

- ▶ Mixed transmission of data and audio.
- ▶ **Synchronous Connection-Oriented (SCO)**
 - Point to point full duplex connection between master & slaves
 - Master reserves slots to allow transmission of packets in regular intervals.
- ▶ **Asynchronous Connection-Less (ACL)**
 - Asynchronous service
 - No reservation of slots
 - The master transmits spontaneously, the addressed slave answers in the following interval.

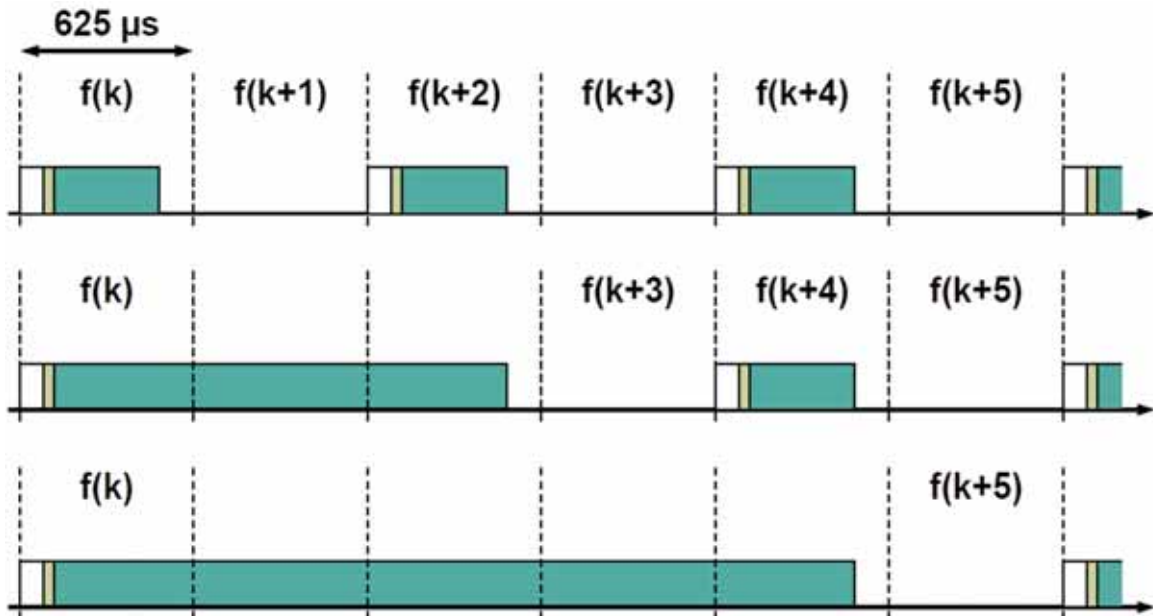
Frequency Hopping Time Multiplex

- ▶ A packet of the master is followed by a slave packet.
- ▶ After each packet, the channel (frequency) is switched.

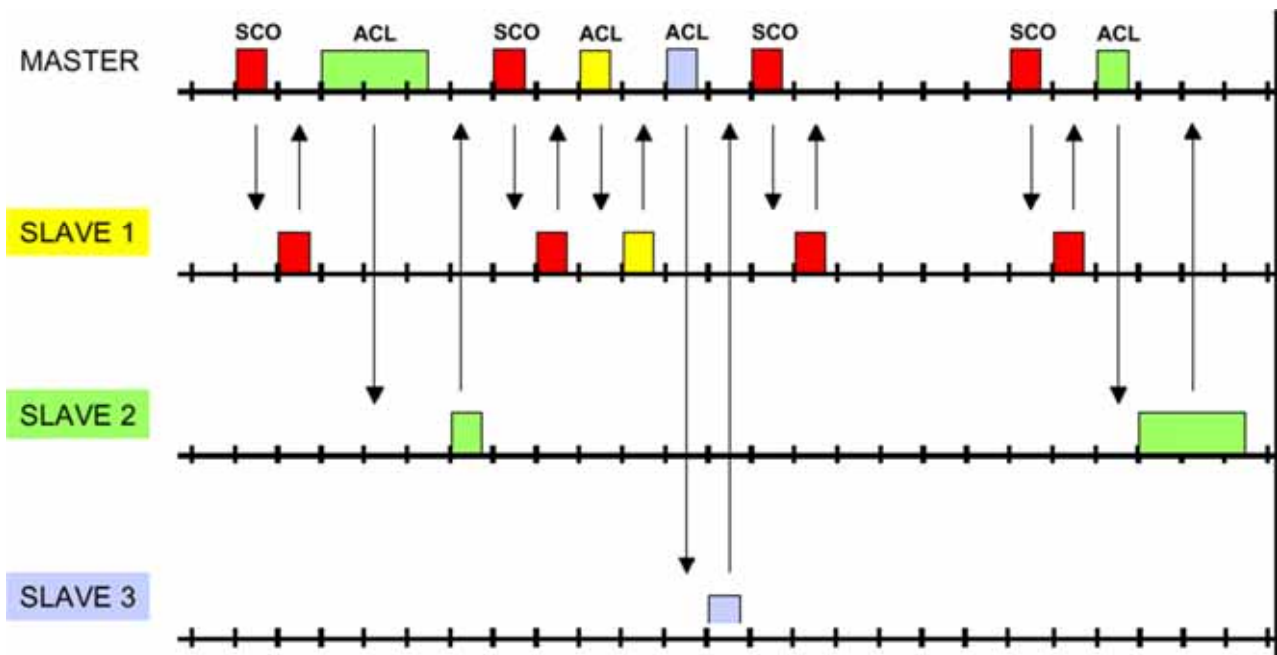


Multi-Slot Communication

- ▶ Master can only start sending in even slot numbers.
- ▶ Packets from master or slave have length of 1, 3 or 5 slots.



ACL and SCO Connections



Modes and States

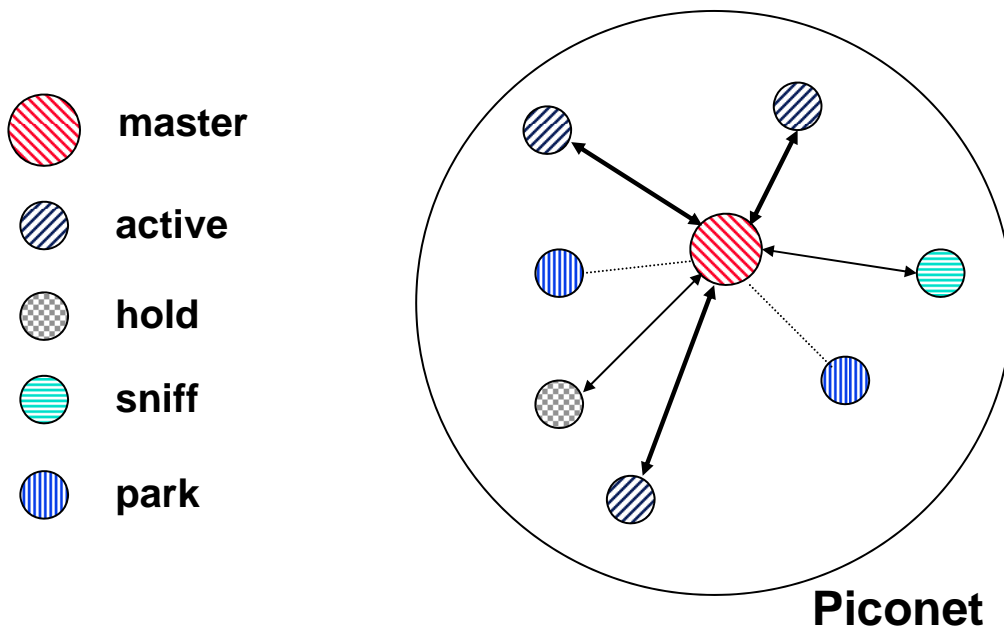
► Modes of operation:

- **Inquiry** (master identifies addresses of neighboring nodes)
- **Page** (master attempts connection to a slave whose address BD_ADDR is known)
- **Connected** (connection between master and slave is established)

► States in connection mode (sorted in decreasing order of power consumption)

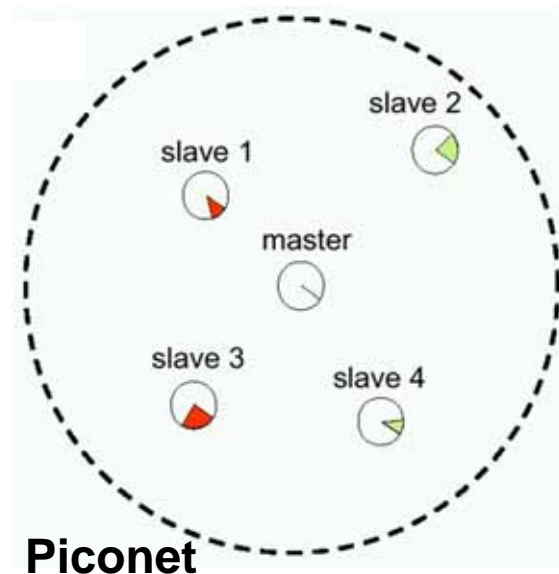
- **active** (active in a connection to a master)
- **hold** (does not process data packets)
- **sniff** (awakens in regular time intervals)
- **park** (passive, in *no connection* with master but still synchronized)

States in Connection Mode

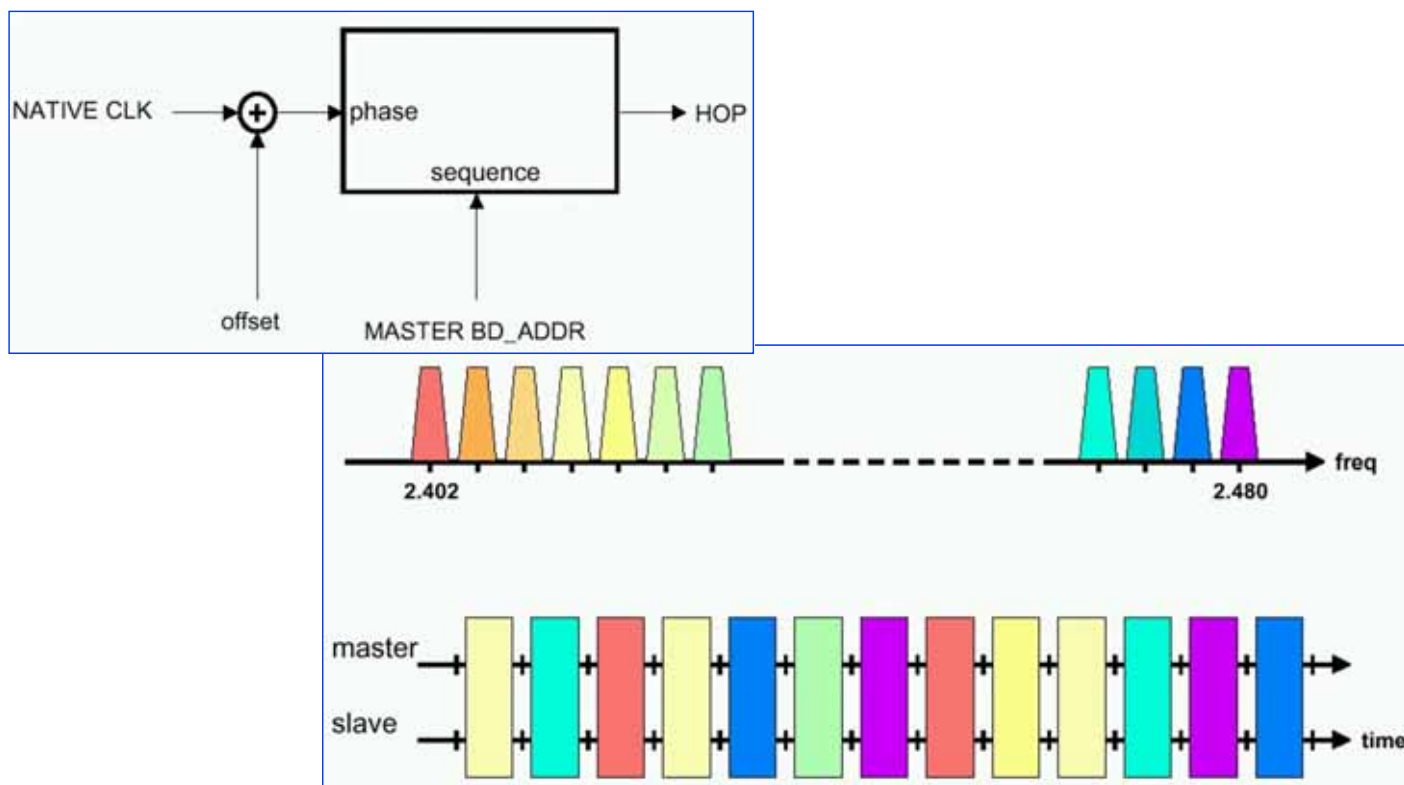


Synchronization in Connection Mode

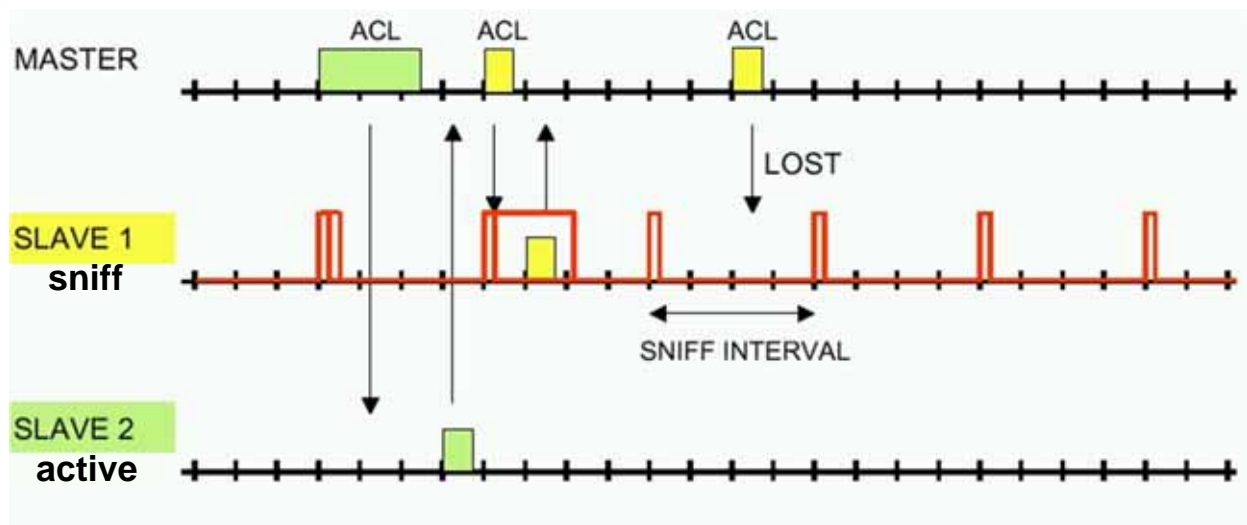
- ▶ The **channel sequence** of a piconet is determined by the BD_ADDR of the master.
- ▶ The **phase** within the sequence is also determined by the master; all slaves follow.



Synchronization in Connection Mode

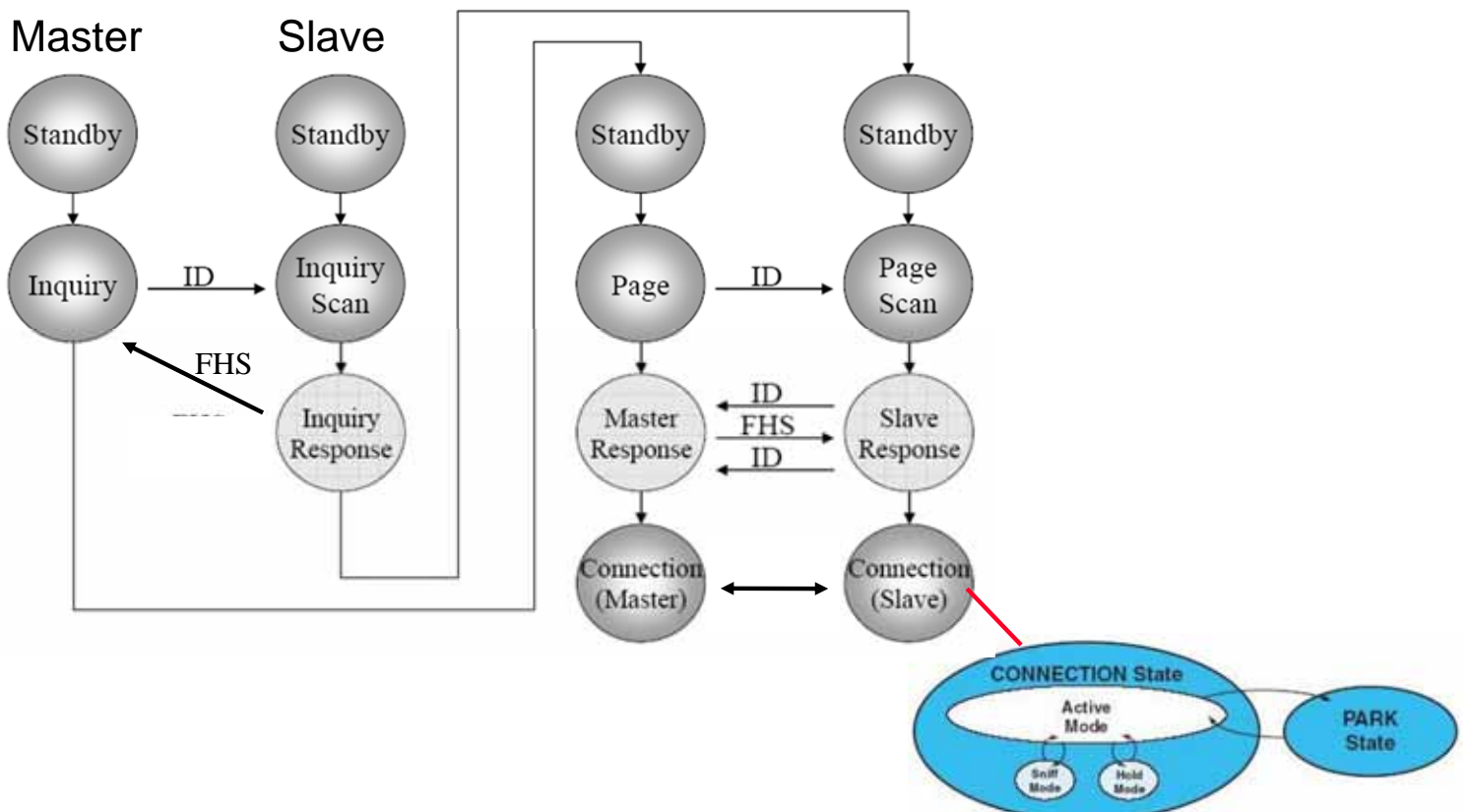


The Sniff State



A slave in sniff state listens in regular time intervals whether there is a packet with its address. If yes, it answers.

From Standby to Connection



The Page Mode

Synchronization between master and slave.
It is a prerequisite for establishing a connection.

page

Master transmits its address to slave (it uses a special channel sequence)

Problem:
Synchronization

page scan

Slave listens, whether its own address is sent from a master.

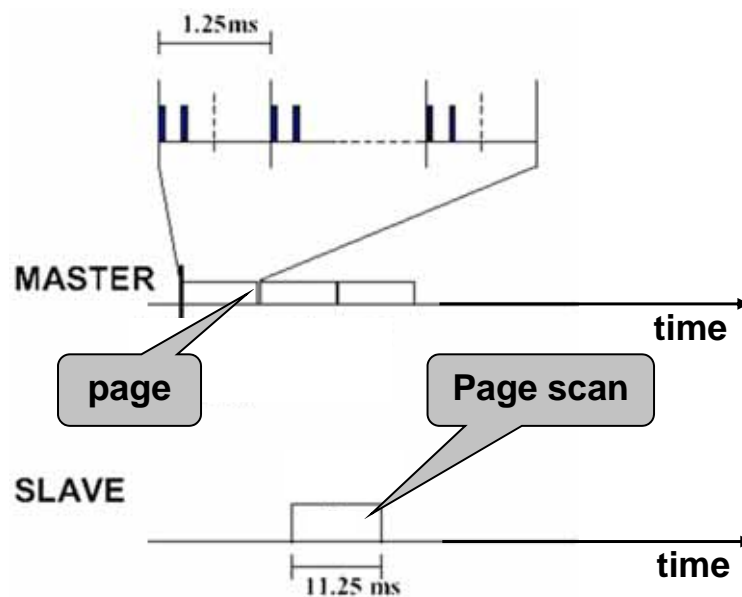
master page response

Slave answers the master with its own address.

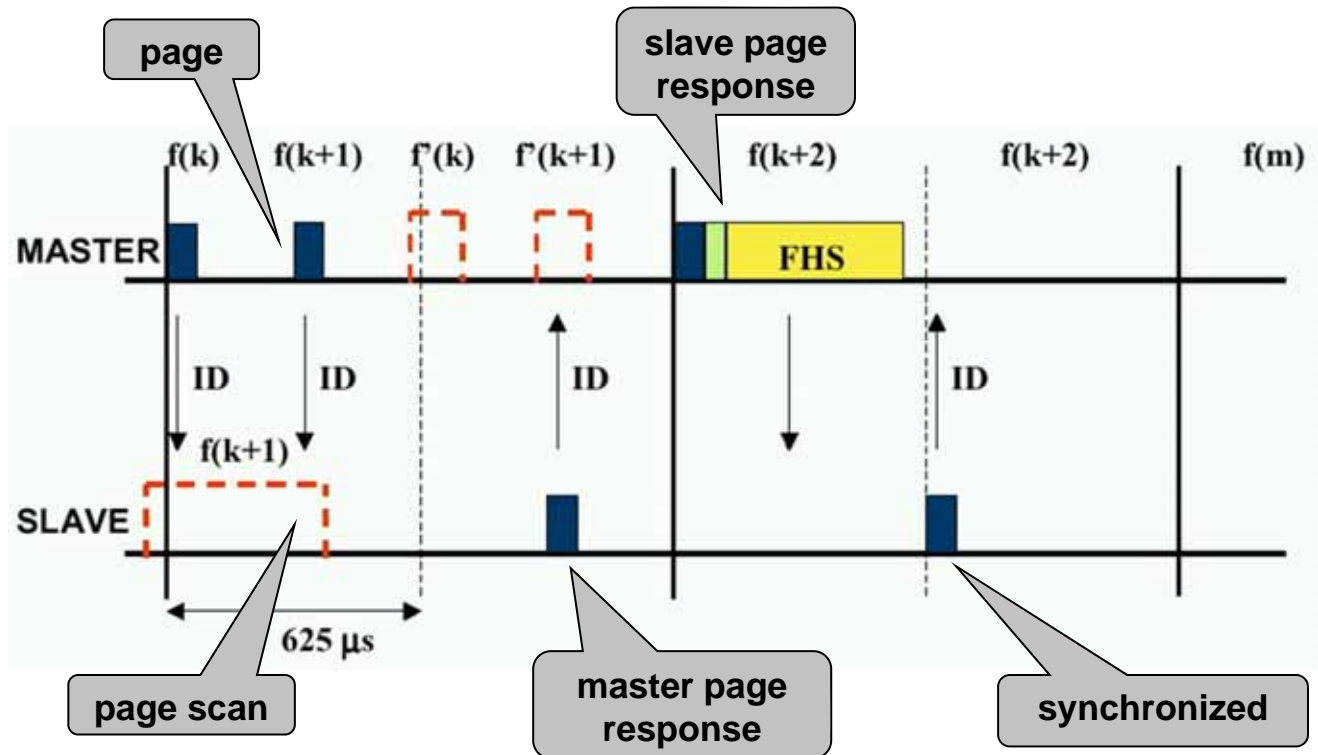
slave page response

Master sends FHS-packet (frequency hop synchronization) to slave. It contains the channel sequence and the phase of the piconet.

The Page Mode



The Page Mode



Protocol Hierarchy

- ▶ The **baseband specification** defines the packet formats, the physical and logical channels, the error correction, the synchronization between receiver and transmitter, and the different modes of operation and states that allow the transmission of data and audio.
- ▶ The **audio specification** defines the transmission of audio signals, in particular the coding and decoding methods.
- ▶ The **link manager** (LM) covers the authentication of a connection and the encryption, the management of a piconet (synchronous/asynchronous connection), the initiation of a connection (asynchronous/synchronous packet types, exchange of name and ID) and the transition between different modes of operation and states.



Protocol Hierarchy

- ▶ The **host controller interface (HCI)** defines a common standardized interface between a host and a bluetooth node; it is specified for several physical interconnections (USB, RS232, PCI, ...).
- ▶ The **link layer control and adaptation layer (L2CAP)** provides an abstract interface for data communication. It segments packets (up to 64kByte) and assembles them again, it allows the multiplexing of connections (simultaneous use of several protocols and connections) and allows the exchange of quality of service information between two nodes (packet rate, packet size, latency, delay variations, maximal rate).



Protocol Hierarchy

- ▶ **RFCOMM** is a simple transport protocol that simulates a serial connection (~RS 232).
- ▶ There are several other protocols that are defined such as the **telephony control protocol specification (TCS)**, the **service discovery protocol (SDP)**, the **OBEX** (Object Exchange Protocol), and **TCP/IP**.
- ▶ Finally, the **application** can use the top layers of the protocol stack.

