Bluetooth

Bluetooth Basics

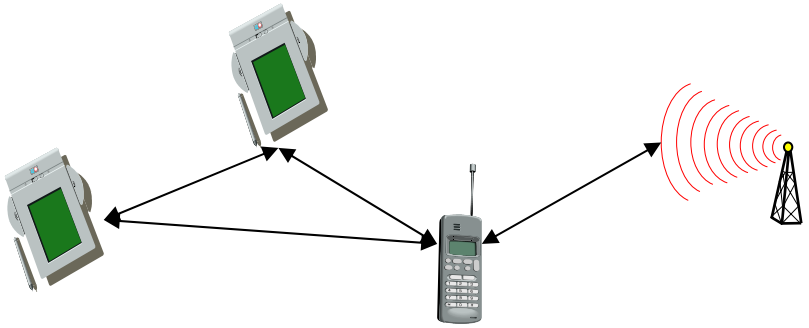
Bluetooth and Linux

Bluetooth at AG Tech

I. Bluetooth

- Idea
  - Universal radio interface for ad-hoc wireless connectivity
  - Interconnecting computer and peripherals, handheld devices, PDAs, cell phones – replacement of IrDA
  - Embedded in other devices, goal: 5€/device (2002: 50€/USB bluetooth)
  - Short range (10 m), low power consumption, license-free 2.45 GHz ISM
  - Voice and data transmission, approx. 1 Mbit/s gross data rate

One of the first modules (Ericsson)



Basics

- History
  - 1994: Ericsson (Mattison/Haartsen), “MC-link” project
  - Renaming of the project: Bluetooth according to Harald “Blåtand” Gormsen [son of Gorm], King of Denmark in the 10th century
  - 1998: foundation of Bluetooth SIG, www.bluetooth.org
  - 2001: first consumer products for mass market, spec. version 1.1 released

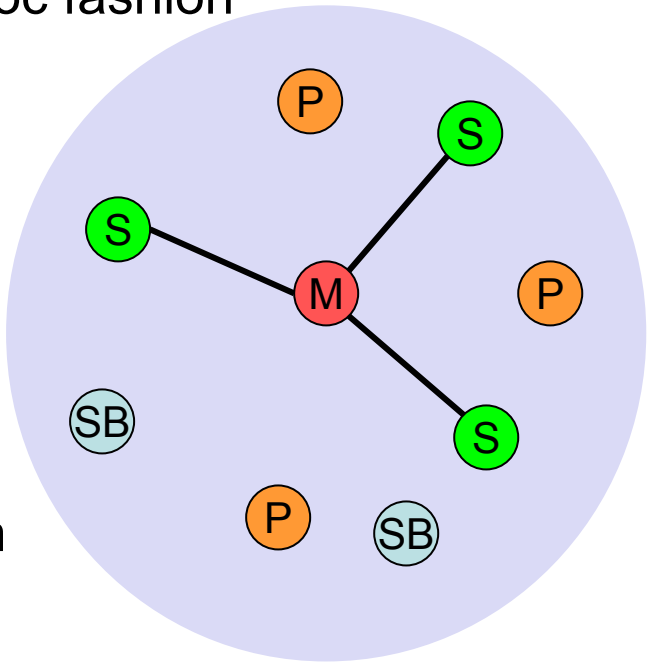
- Special Interest Group
  - Original founding members: Ericsson, Intel, IBM, Nokia, Toshiba
  - Added promoters: 3Com, Agere (was: Lucent), Microsoft, Motorola
  - > 2500 members
  - Common specification and certification of products

Characteristics

- 2.4 GHz ISM band, 79 (23) RF channels, 1 MHz carrier spacing
  - Channel 0: 2402 MHz … channel 78: 2480 MHz
  - G-FSK modulation, 1-100 mW transmit power
- FHSS and TDD
  - Frequency hopping with 1600 hops/s
  - Hopping sequence in a pseudo random fashion, determined by a master
  - Time division duplex for send/receive separation
- Voice link – SCO (Synchronous Connection Oriented)
  - FEC (forward error correction), no retransmission, 64 kbit/s duplex, point-to-point, circuit switched
- Data link – ACL (Asynchronous ConnectionLess)
  - Asynchronous, fast acknowledge, point-to-multipoint, up to 433.9 kbit/s symmetric or 723.2/57.6 kbit/s asymmetric, packet switched
- Topology
  - Overlapping piconets (stars) forming a scatternet

Piconet

- Collection of devices connected in an ad hoc fashion
- One unit acts as master and the others as slaves for the lifetime of the piconet
- Master determines hopping pattern, slaves have to synchronize
- Each piconet has a unique hopping pattern
- Participation in a piconet = synchronization to hopping sequence

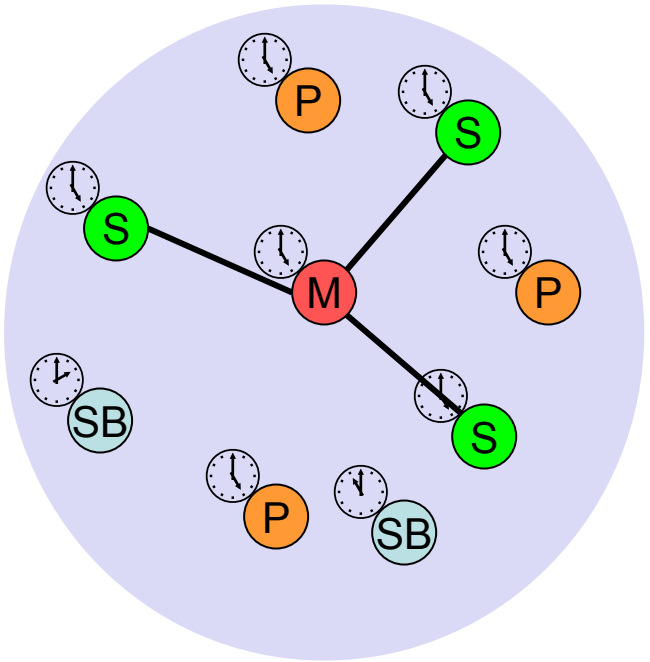


M=Master P=Parked
S=Slave SB=Standby

- Each piconet has one master and up to 7 simultaneous slaves (> 200 could be parked)

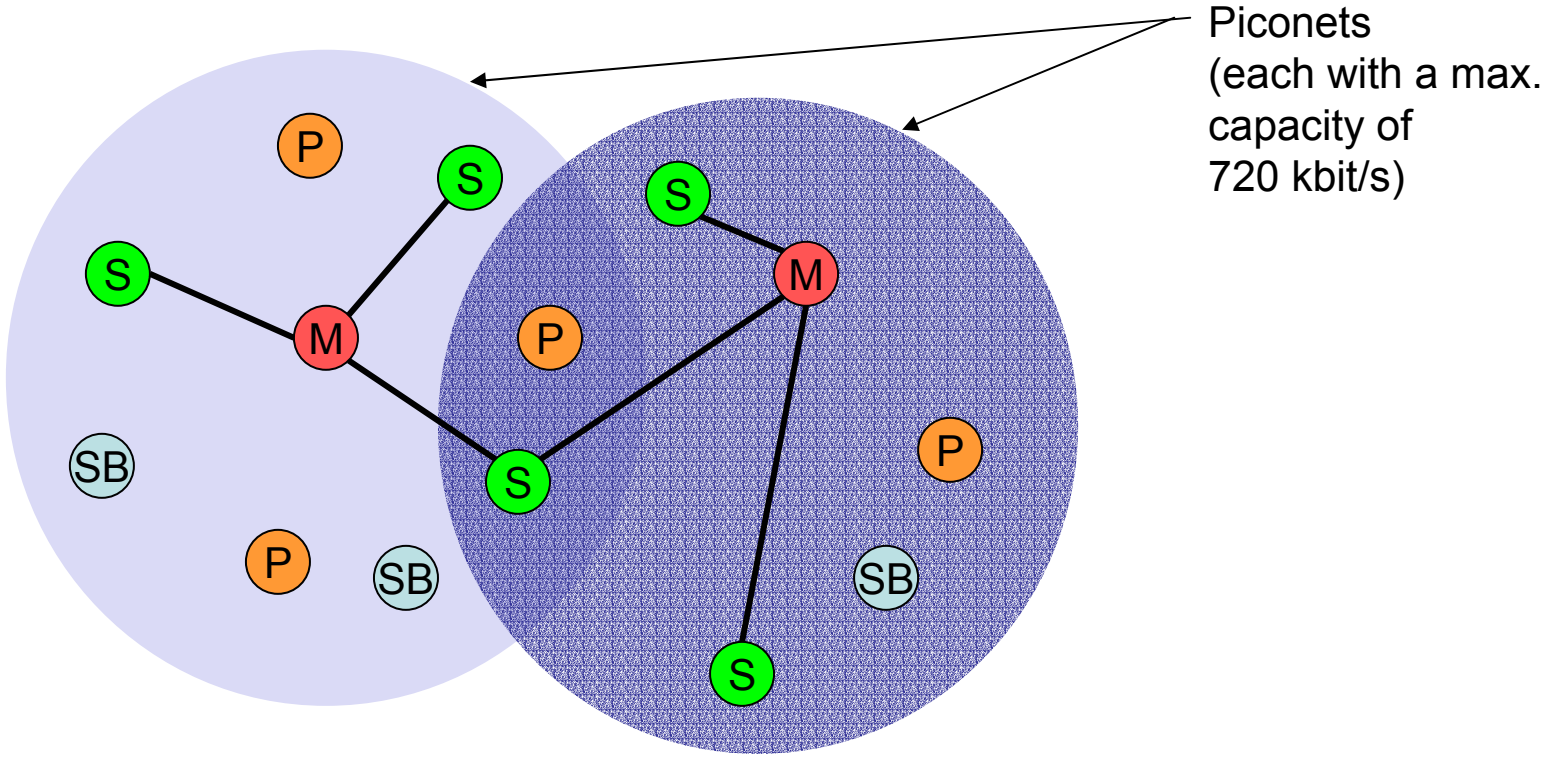
Forming a piconet

- All devices in a piconet hop together
  - Master gives slaves its clock and device ID
    - Hopping pattern: determined by device ID (48 bit, unique worldwide)
    - Phase in hopping pattern determined by clock
- Addressing
  - Active Member Address (AMA, 3 bit)
  - Parked Member Address (PMA, 8 bit)

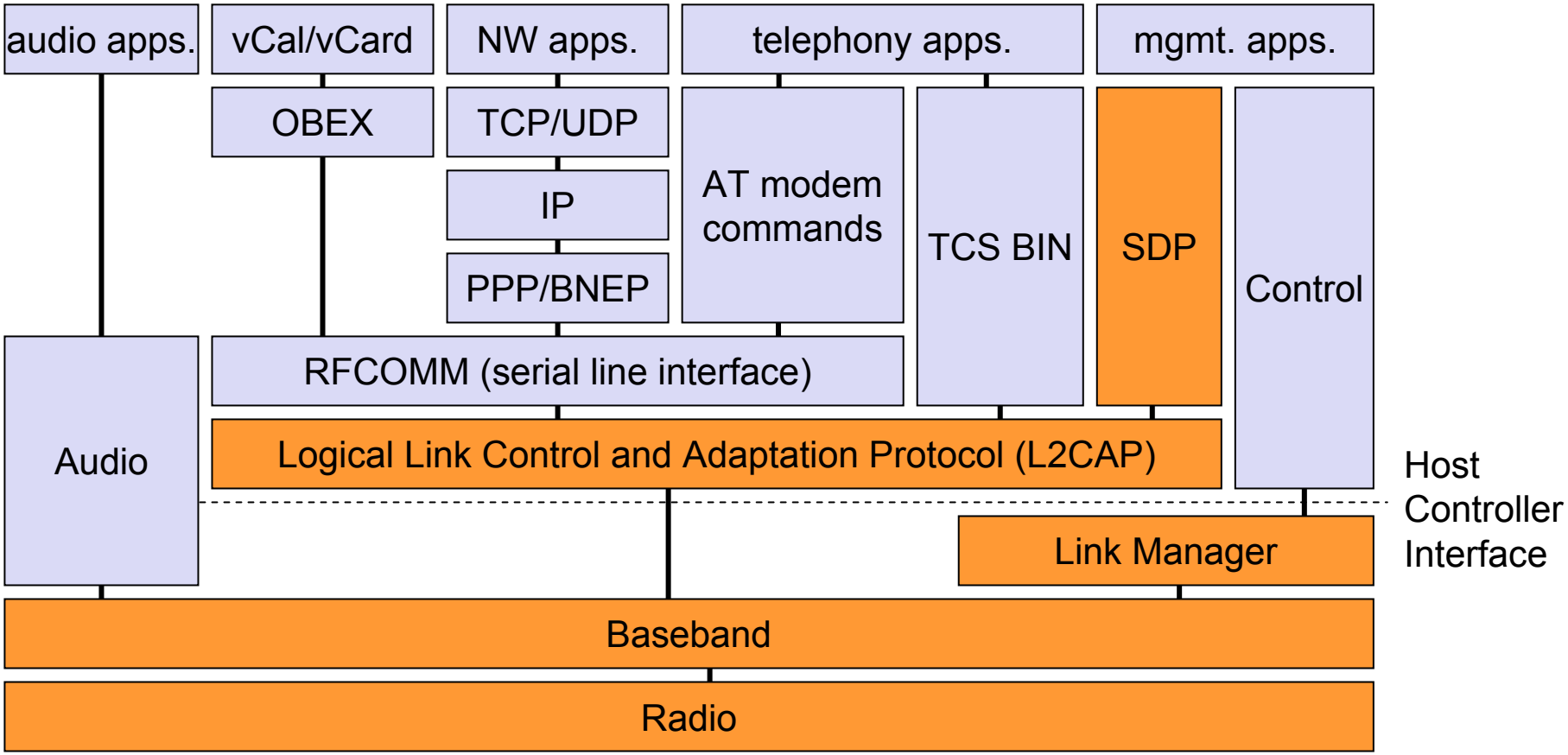


Scatternet

- Linking of multiple co-located piconets through the sharing of common master or slave devices
  - Devices can be slave in one piconet and master of another
  - Devices jumping back and forth between the piconets



Bluetooth protocol stack

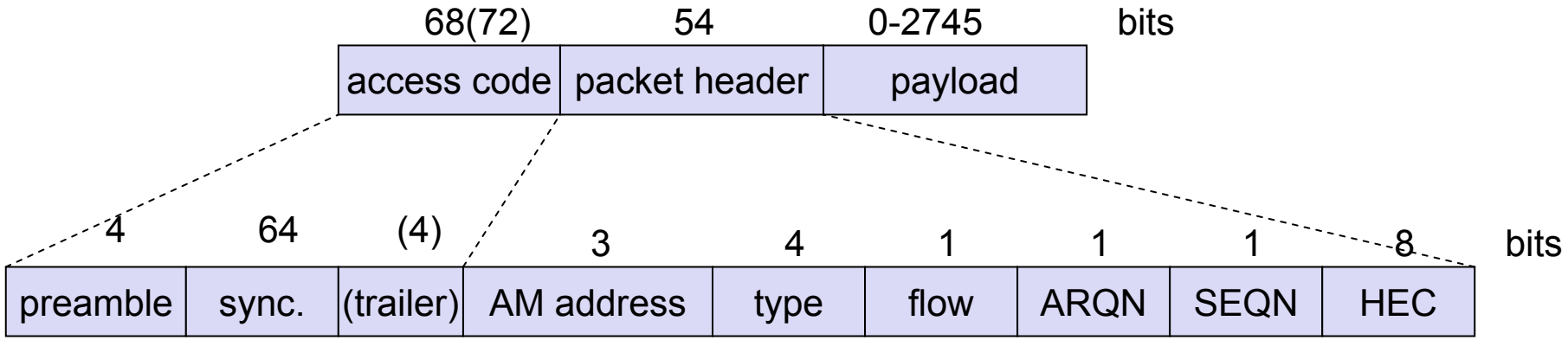


AT: attention sequence
OBEX: object exchange
TCS BIN: telephony control protocol specification – binary
BNEP: Bluetooth network encapsulation protocol

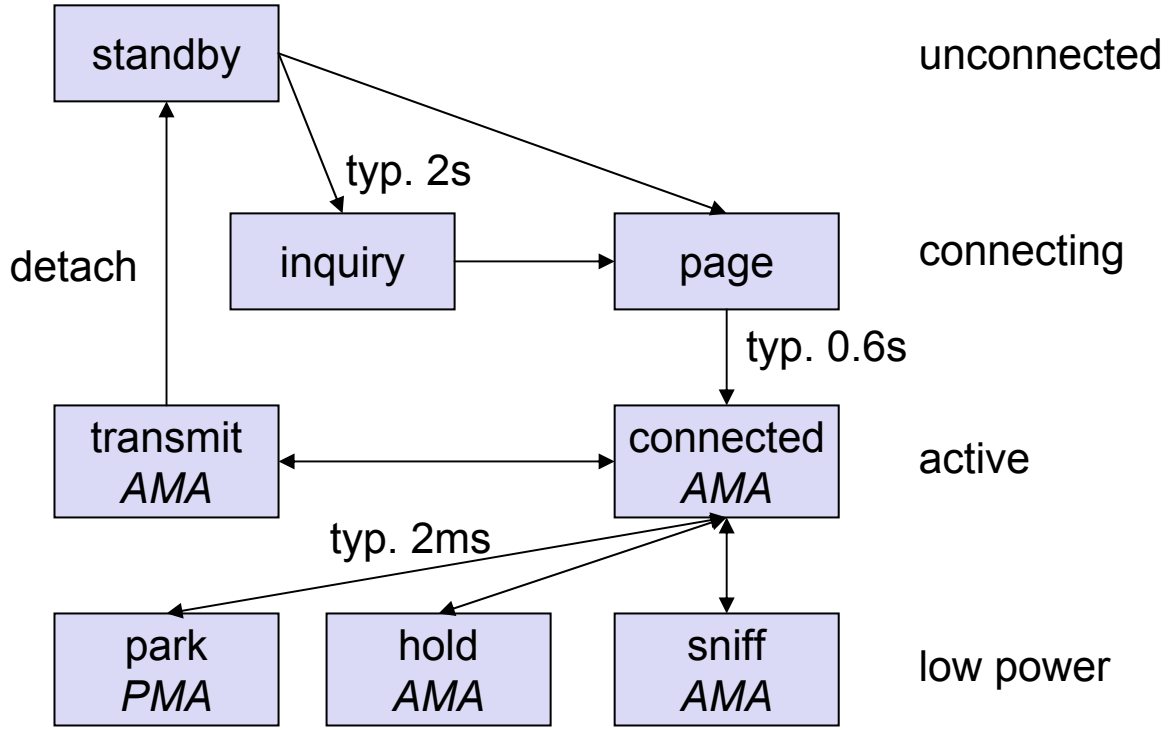
SDP: service discovery protocol
RFCOMM: radio frequency comm.

Baseband

- Piconet/channel definition
- Low-level packet definition
  - Access code
    - Channel, device access, e.g., derived from master
  - Packet header
    - 1/3-FEC, active member address (1 master, 7 slaves), link type, alternating bit ARQ/SEQ, checksum



Baseband states of a Bluetooth device

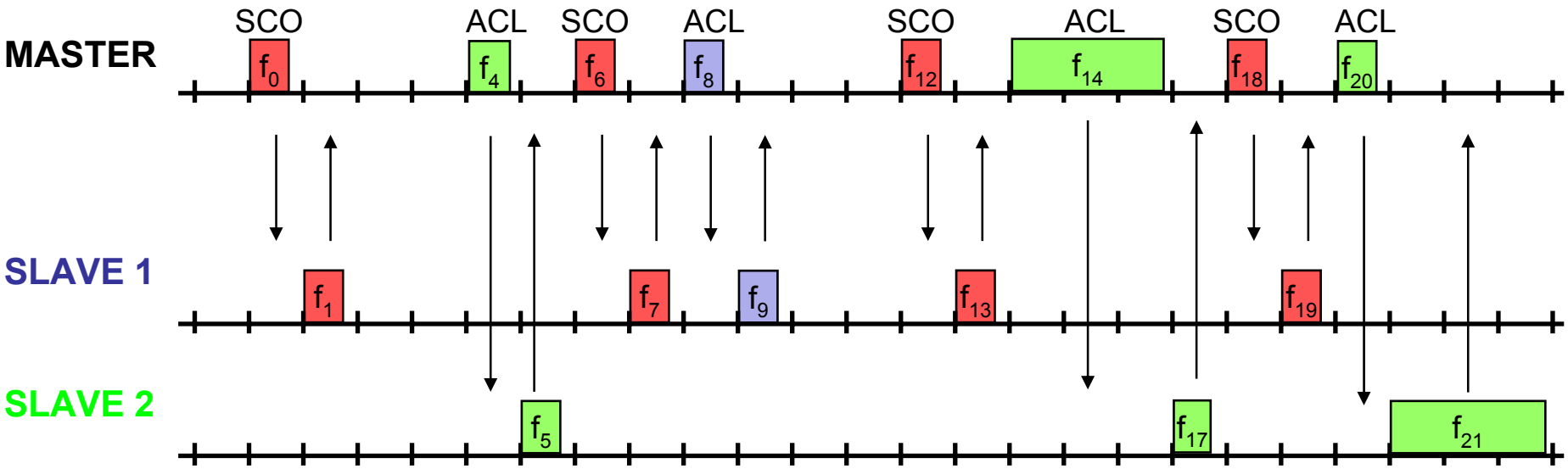


Standby: do nothing
Inquire: search for other devices
Page: connect to a specific device
Connected: participate in a piconet

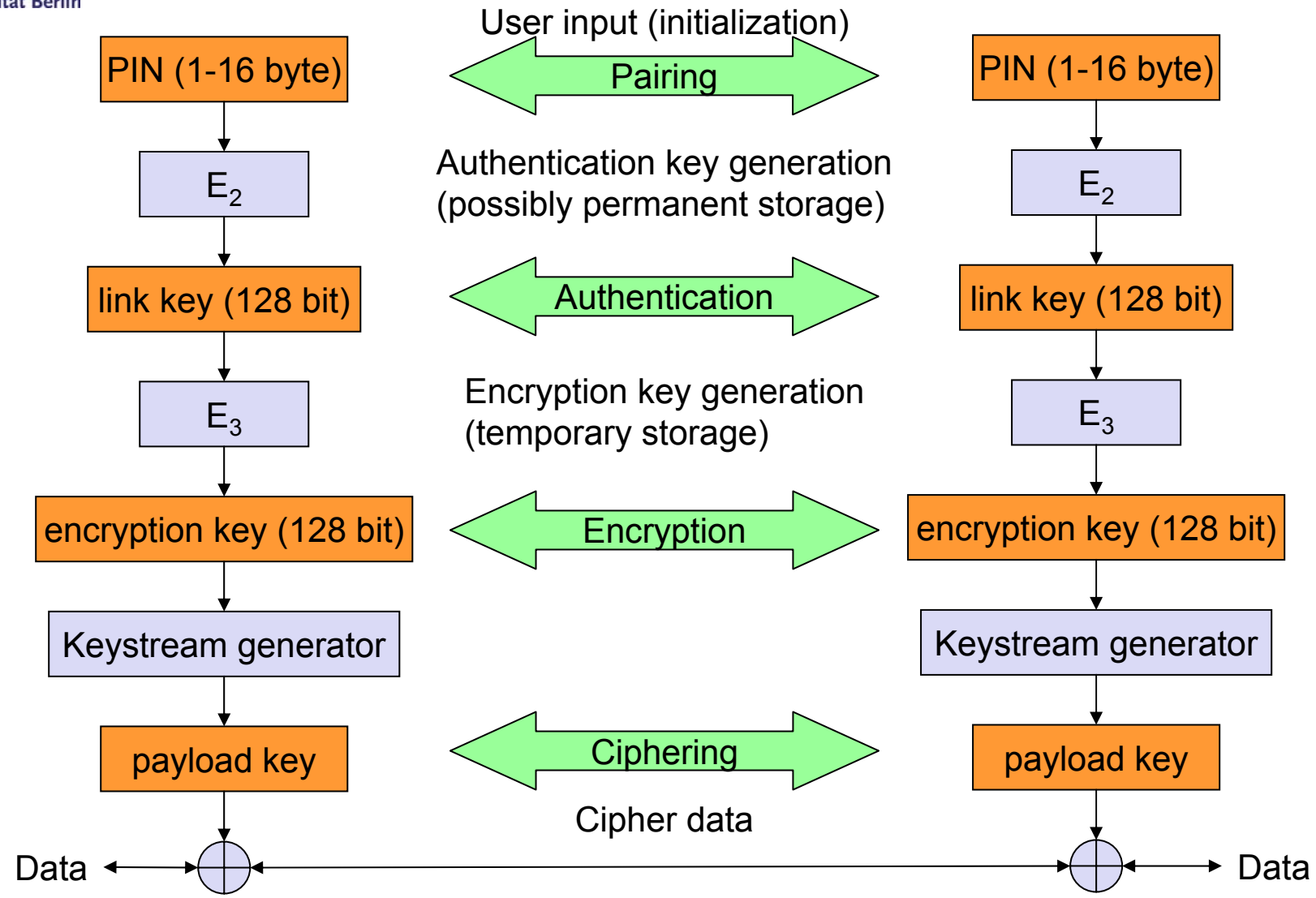
Park: release AMA, get PMA
Sniff: listen periodically, not each slot
Hold: stop ACLs, SCO still possible, possibly participate in another piconet

Baseband link types

- Polling-based TDD packet transmission
  - 625µs slots, master polls slaves
- SCO (Synchronous Connection Oriented) – Voice
  - Periodic single slot packet assignment, 64 kbit/s full-duplex, point-to-point
- ACL (Asynchronous ConnectionLess) – Data
  - Variable packet size (1,3,5 slots), asymmetric bandwidth, point-to-multipoint



Security


User input (initialization)
PIN (1-16 byte)
Pairing
PIN (1-16 byte)
E2
Authentication key generation
(possibly permanent storage)
E2
link key (128 bit)
Authentication
link key (128 bit)
E3
Encryption key generation
(temporary storage)
E3
encryption key (128 bit)
Encryption
encryption key (128 bit)
Keystream generator
Keystream generator
payload key
Ciphering
payload key
Cipher data
Data
Data

L2CAP - Logical Link Control and Adaptation Protocol

- Simple data link protocol on top of baseband
- Connection oriented, connectionless, and signalling channels
- Protocol multiplexing
  - RFCOMM, SDP, telephony control
- Segmentation & reassembly
  - Up to 64kbyte user data, 16 bit CRC
- QoS flow specification per channel
  - Follows RFC 1363, specifies delay, jitter, bursts, bandwidth
- Group abstraction
  - Create/close group, add/remove member

SDP – Service Discovery Protocol

- Inquiry/response protocol for discovering services
  - Searching for and browsing services in radio proximity
  - Adapted to the highly dynamic environment
  - Defines discovery only, not the usage of services
  - Caching of discovered services

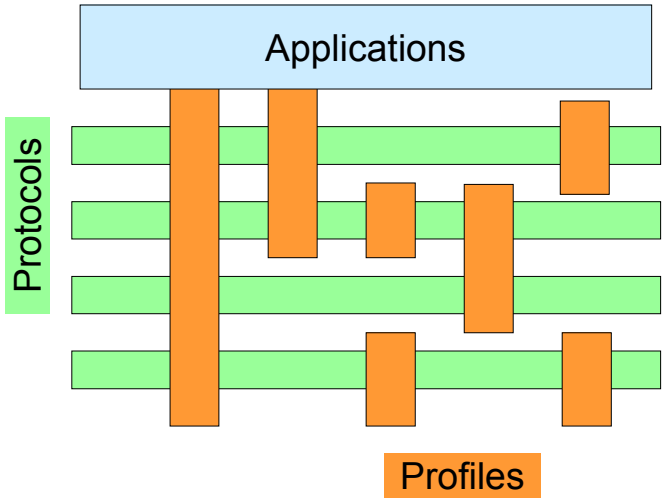
- Service record format
  - Information about services provided by attributes
  - Attributes are composed of an 16 bit ID (name) and a value
  - IDs may be derived from 128 bit Universally Unique Identifiers (UUID)

Additional protocols

- RFCOMM
  - Emulation of a serial port (supports a large base of legacy applications)
  - Allows multiple ports over a single physical channel
- Telephony Control Protocol Specification (TCS)
  - Call control (setup, release)
  - Group management
- OBEX
  - Exchange of objects, IrDA replacement

Profiles

- Represent default solutions for a certain usage model
  - Vertical slice through the protocol stack
  - Basis for interoperability
- Generic Access Profile
- Service Discovery Application Profile
- Cordless Telephony Profile
- Intercom Profile
- Serial Port Profile
- Headset Profile
- Dial-up Networking Profile
- Fax Profile
- LAN Access Profile
- Generic Object Exchange Profile
- Object Push Profile
- File Transfer Profile
- Synchronization Profile



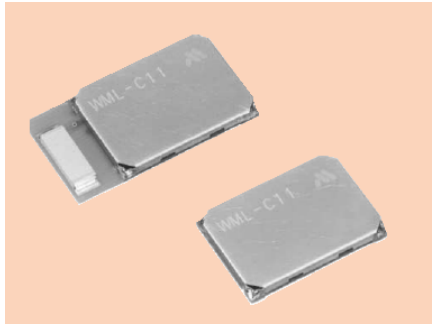
Additional Profiles
Advanced Audio Distribution
PAN
Audio Video Remote Control
Basic Printing
Basic Imaging
Extended Service Discovery
Generic Audio Video Distribution
Hands Free
Hardcopy Cable Replacement

II. Bluetooth under Linux

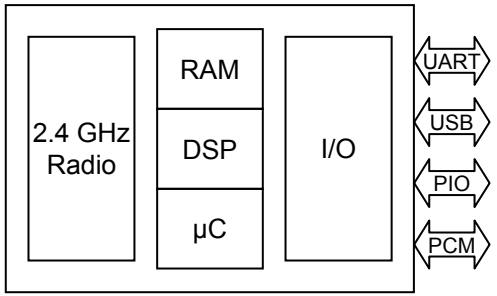
- Source for information: http://bluez.sourceforge.net/howto/
- Relevant configs:
  - */etc/hcid.conf, /etc/bluetooth/pin, /etc/rc.d/init.d/bluetooth, /etc/pcmcia/bluetooth, /etc/pcmcia/bluetooth.conf*
- USB device:
  - *modprobe hci_usb*
  - *hciconfig hci0 up*
- Tools:
  - *hcitool*
  - *l2ping*
  - *l2test, scotest*
  - *hcidump* (like *tcpdump*)
- Example: http://bluez.sourceforge.net/howto/node30.html
- Set up PPP over Bluetooth, running SDPd (SDP daemon), open bluetooth socket, deploy BNEP, …

III. Bluetooth at AG Tech

- Usage of Bluetooth modules from CSR (Cambridge Silicon Radio)
- 66% of market, Linux driver available, very good documentation



- On-board microcontroller handles basic bluetooth protocol functions
- SDK available for µC-programming
- USB and UART interfaces
  ⇒ looks like a „normal“ device



- Compaq iPAQ with bluetooth (CSR module…)
  - Bluetooth is now working (http://www.handhelds.org/projects/h3800.html)
  - NEW: dual boot iPAQs!
- Goal: Matchbox-sized web server with IP network access and display

Bluetooth Research

- Interoperability Bluetooth ⇔ WLAN
  - Same frequency, different access, no coordination
  - IEEE working group: Adaptive Frequency Hopping
- Service Discovery:
  - Using Service Discovery Protocol (SDP) for immediate access to services in a room or location
  - Integration with other service discovery architectures (ESDP, UPnP, Jini, .NET, …)
- Ad-hoc networking
  - How to build spontaneous networks
  - How to make this fast, but secure

Bluetooth/USB adapter (2002: 50€)