

Sveučilište u Zagrebu
Fakultet elektrotehnike i računarstva
Zavod za elektroniku, mikroelektroniku, računalne i inteligentne sustave

Kolegij: Mreže računala
Seminarski rad:

BLUETOOTH



Martin Žagar
36374880

U Zagrebu, 1. siječnja 2003.

Sadržaj:

1. UVOD	2
1.1. Plavozubi? Što je to?	2
1.2. Svakodnevna primjena	2
1.3. Korištenje	3
1.4. Bluetooth SIG	3
2. ARHITEKTURA BLUETOOTH-a	5
2.1. Radio	7
2.2. Baseband	8
2.3. Piconet i Sccaternet	9
2.4. Fizičke veze	11
2.5. Struktura paketa	12
2.6. Logički kanali	15
2.7. Link controller	15
2.8. Stanja link kontrolera	15
2.9. Audio	18
2.10. Link manager	19
2.11. Host controller interface (HCI)	21
2.12. L2CAP (Logical link control and adaptation protocol)	26
2.13. RFCOMM	28
2.14. SDP	29
2.15. Aplikacije – Bluetooth profili	31
2.16. Usporedba s WAP-om i IrDA	32
2.17. Usporedba s IEEE 802.11 (bežični ethernet) i 3GPP-om	33
2.18. 3GPP – što je to?	34
3. ZAKLJUČAK	35
3.1. Razvoj i primjena u budućnosti	35
4. LITERATURA	36

1. UVOD

1.1. Plavozubi? Što je to?

Harald Bluetooth je bio danski kralj koji je živio u doba kad su Vikinzi bili jedan od najmoćnijih naroda Europe. Iako je živio kratko (940. – 981.) uspio je ujediniti i pokrstiti kraljevine Dansku i Norvešku. Zato je u tim krajevima i danas slavljen kao hrabar i odlučan čovjek, simbol skandinavskog jedinstva.

Danas je Bluetooth upravo jedinstven poduhvat 9 velikih kompanija, pokretača ideje bežične komunikacije između mobilnih uređaja, spoja telekomunikacija i računala. Tih 9 pokretača danas su članovi Bluetooth SIG-a (Special Interest Group), a ukupno u svijetu trenutno postoji preko 3000 kompanija koje koriste rješenja Bluetooth-a. Prve ideje o bežičnim vezama kratkog dometa javljaju se 1994. u glavama Ericssonovih inženjera. 1995. počinju istraživanja i razvoj nove tehnologije, 1998. je osnovan Bluetooth SIG, 1999. je objavljena prva Bluetooth specifikacija, a 2000. Bluetooth nalazi primjenu u svakodnevi.



Slika 1. Devet kompanija, pokretača Bluetooth-a

1.2. Svakodnevna primjena

WIRELESSLY. POWERFULLY. EFFICIENTLY. BLUETOOTH™ WIRELESS TECHNOLOGY WORKS HOWEVER YOU WORK. IT WORKS WHENEVER YOU WORK, SEAMLESSLY CONNECTING ALL OF YOUR MOBILE DEVICES. CREATING UNPRECEDENTED PRODUCTIVITY.

Ovo je slogan koji se može naći u [1]. To zapravo znači da je Bluetooth postao (ili će tek postati, ovisno o zemlji) nešto bez čega se ne može biti. Povezan u mrežu, a bez žica – to je naša budućnost. Zapravo, može se reći da je ideja u povezanosti, poštujući slobodu koju pruža neožičenost (dakle nema nikakvog prtljanja s kabelima). Najčešće se danas koristi u povezivanju prijenosnih računala (laptopa), mobitela i prijenosnih PDA-a na Internet. Bluetooth u tom pogledu omogućuje razmjenu podataka i pruža sučelje prema krajnjem korisniku. Zato jer se pri njegovom konstruiranju razmišljalo «otvoreno» (open system), danas nalazi primjenu diljem svijeta zbog svoje otvorenosti. Uspostavljanje tih standarada

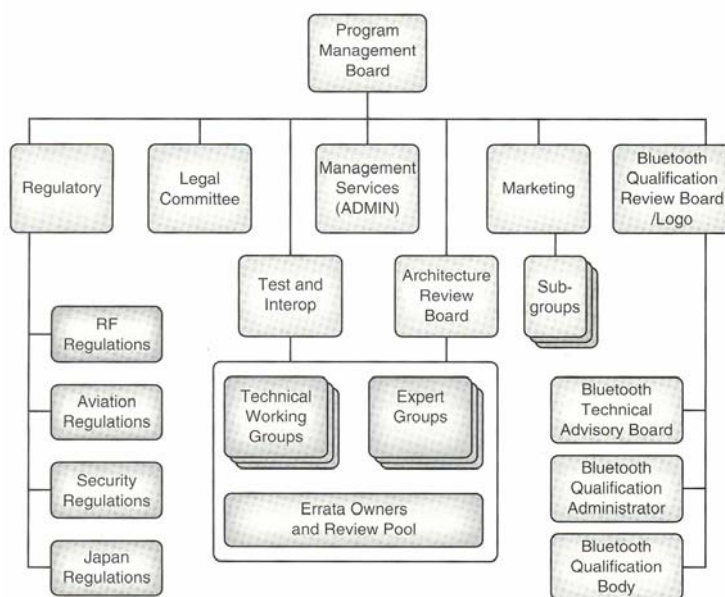
znači integriranje dobro testirane tehnologije sa učinkovitošću i malim troškovima. Bluetooth je toliko uspješan zato jer je razvijen poštujući tehnološke zakone i razumijevanje potrošača i biznisa. Standard je podržan od mnogih proizvođača koji su razvili aplikacije u širokom spektru tržišnih segmenata, od autoindustrije do elektroničke industrije.

1.3. Korištenje

AT HOME. AT WORK. ON THE ROAD. USING BLUETOOTH WIRELESS TECHNOLOGY MEANS TOTAL FREEDOM FROM THE CONSTRAINTS AND CLUTTER OF WIRES – ON YOUR DESKTOP, IN YOUR CAR...WHEREVER THE POSSIBILITIES OF WIRELESS PRODUCTIVITY MAY TAKE YOU.

Za sada je korištenje ograničeno na komunikaciju između bežičnog headset-a sa mobitelom negdje u torbi ili odjelu, ili sinkronizaciju između PDA-a i PC-a, svi oni dijele isti adresar, raspored i podsjetnik. A to je tek početak. Uskoro će svjetlo dana ugledati Bluetooth olovka ili torba, a ni Bluetooth škrinja nije daleko. Pred nama je dakle Bluetooth budućnost.

1.4. Bluetooth SIG



Slika 2. Organizacija Bluetooth SIG-a

Osim gore navedenih velikih igrača, u Bluetooth SIG se može učlaniti bilo koja kompanija. Time kompanije dobivaju besplatnu dozvolu izrade proizvoda koji koriste Bluetooth tehnologiju. Na čelu te organizacije je upravni odbor (kao što je predočeno na slici 2. koja je kao i sve ostale slike preuzeta iz [5]) koji upravlja mnogim grupama, a najvažnija su tehničke radne grupe. U te grupe koje rade na Bluetooth standardima, se mogu pridružiti članovi Bluetooth SIG te time izravno utjecati na razvoj. *Bluetooth promoters group* se brine o predstavljanju Bluetooth specifikacije i njihova odluka je da to bude otvoren standard zato jer korisnici radije prihvaćaju tehnologiju koja može biti izrađena od mnogih proizvođača, nego onu koja je limitirana na samo nekolicinu. Cilj nije samo napraviti čim više *core* (temeljnih, jezgrenih) specifikacija, nego učiniti proizvode korisnijima olakšavanjem komunikacije među

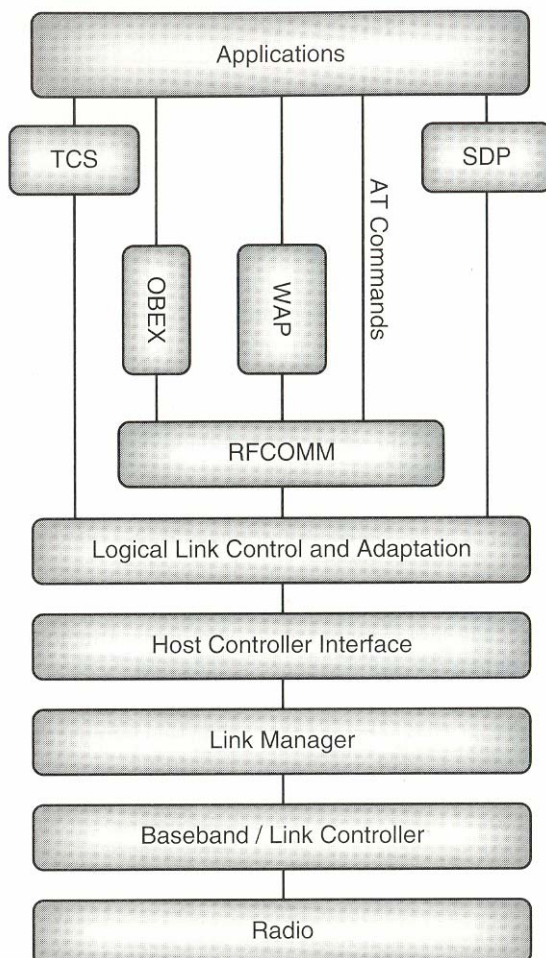
njima. Kablovi ne samo što su nespretni i lomljivi, nego se mogu i izgubiti, pa su bežična rješenja logičnija, ali i u početku skupa. Usavršavanjem tehnologije, danas to više nije slučaj (svaka jedinica košta manje od 5\$).

Zato što je Bluetooth tehnologija dizajnirana za mobilne uređaje, mora biti sposobna raditi na baterije. To znači da potrošnja energije mora biti čim je moguće manja, isto kao i dimenzije, kako bi se mogla ugrađivati u minijaturne mobitele ili PDA-e. Dakle, ukupni ciljevi su široka dostupnost, niska cijena, lakoća rukovanja, pouzdanost, sigurnost, robusnost, male dimenzije i niska potrošnja. Samim time što je frekvencijsko područje koje koristi Bluetooth besplatno u većini zemalja, te time što je kompletna specifikacija rješenja potpuno besplatna, osigurana je velika dostupnost rješenja.

Tvorci Bluetooth-a su se poslužili provjerenom metodom - raslojavanje arhitekture. Drugim riječima, Bluetooth arhitektura je temeljena na slojevima. Svaki sloj pruža neku funkciju sloju iznad skrivajući od njega implementaciju. Važna je samo komunikacija između slojeva koja se obavlja preko definiranih sučelja.

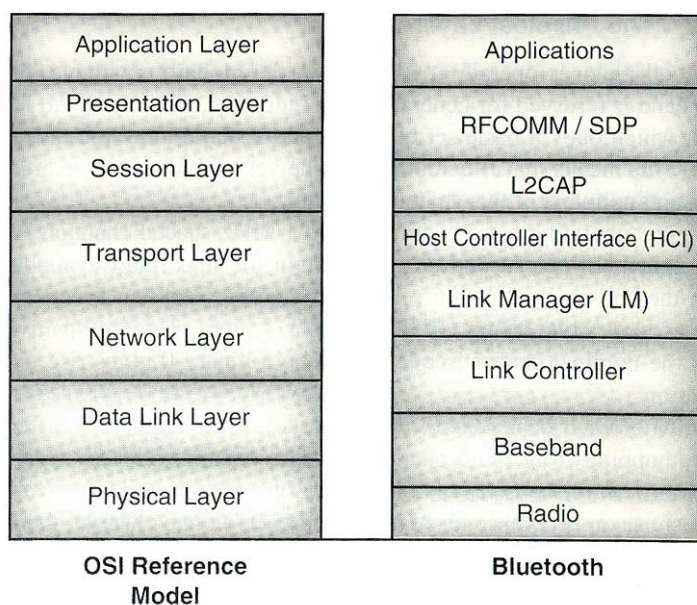
2. ARHITEKTURA BLUETOOTH-a

Bluetooth je radio veza kratkog dometa (10 – 100 m) namjenjena kao zamjena za kablove koji povezuju različite uređaje. Bluetooth radi na slobodnom ISM (*Industrial, scientific, medicinal*) frekvencijskom području od 2.4 GHz. Za komunikaciju se koristi primopredajnik koji stalno, u skokovima (*frequency hop*) mijenja radnu frekvenciju kako bi se smanjile smetnje i pogreške pri prijenosu. Kako bi se pojednostavila izvedba primopredajnika, koristi se binarna frekvencijska modulacija pri prijenosu podataka. Broj simbola koji se mogu prenijeti je 1 Mbit/sec. Koristi se dvosmjerni prijenos podataka. Kroz kanal informacija se prenosi u paketima, koji se šalju svaki na svojoj frekvenciji (svaki paket koristi jedan frekvencijski skok). Komunikacijski kanal je podijeljen u slotove (vremenske odsječke) od kojih svaki slot ima nominalnu duljinu od 625 μ s. Paket se normalno smješta u svoj slot, ali pojedini paketi mogu zauzeti do 5 slotova. Bluetooth može podržavati asinkroni kanal podataka, do 3 istovremena sinkrona glasovna kanala ili kanal koji istovremeno podržava asinkroni prijenos podataka i sinkroni prijenos glasa. Svaki glasovni kanal podržava dvosmjerni 64 kbps sinkroni glasovni prijenos. Brzina asinkronog kanala može biti maksimalno 73.2 kbps u jednom smjeru (tada je brzina u drugom smjeru 57.6 kbps) ili 433.9 kbps u oba kanala istovremeno.



Slika 3. Bluetooth protokolni stog

Iako će biti kasnije detaljnije objašnjeni, samo kratki komentar uz sliku 3. Najniži sloj, *Radio*, modulira i demodulira podatke za prijenos i primanje. *Baseband / link controller* kontrolira fizičke veze preko *radio* sloja, sastavlja pakete i kontrolira skokove frekvencije (*frequency hopping*). *Link manager* kontrolira i konfigurira veze prema drugim uređajima. *Host controller interface* komunicira između udaljenih domaćina i Bluetooth modula. *Logical link control and adaptation* multipleksira podatke iz viših slojeva i pretvara različite duljine paketa. *RFCOMM* pruža pogodnost serijskog sučelja preko RS232. *WAP* i *OBEX* predstavljaju sučelje prema višim slojevima drugih komunikacijskih protokola. *SDP* (*service discovery protocol*) daje Bluetooth uređajima mogućnost nalaženja usluga koje su podržane od drugih uređaja. *TCS* (*telephony control protocol specification*) pruža telefonske usluge. Na kraju, Bluetooth profili daju smjernice kako će aplikacije koristiti *Bluetooth protocol stack*.



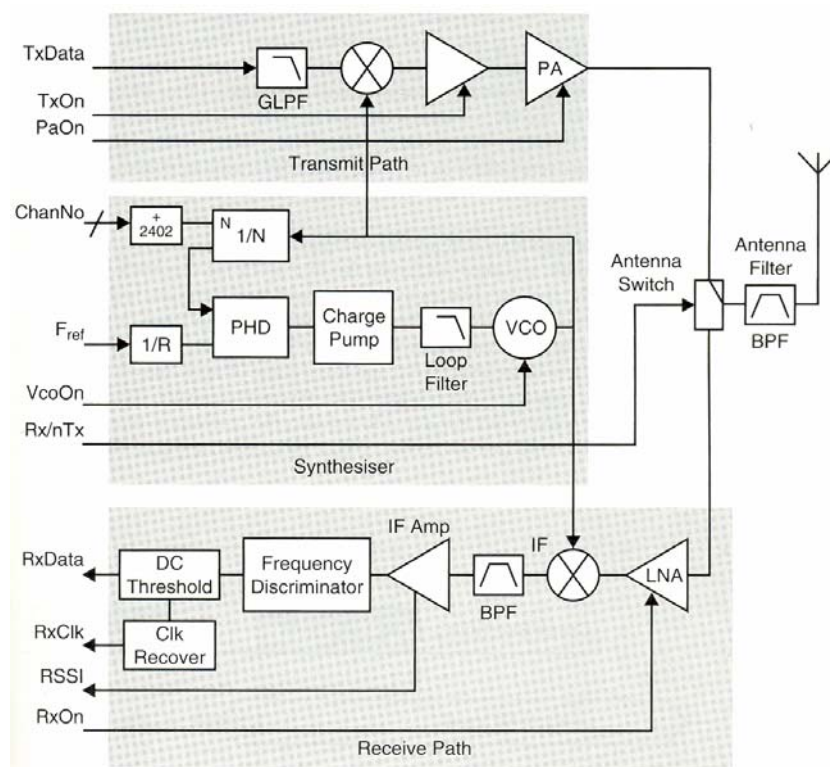
Slika 4. Usporedba Bluetooth - OSI

Važno je primijetiti da za razliku od OSI modela ili Internet modela, slojevi BT arhitekture nisu naredani jedan na drugi, tj. podaci se ne razmjenjuju kroz sve postojeće slojeve (slika 4.). Za usporedbu pogledajmo OSI model koji se sastoji od 7 slojeva. Najdonji je fizički sloj, a najgornji aplikacijski sloj. Podatak koji prolazi kroz stog prođe cijeli put od prvog do sedmog sloja prije negoli bude prosljeđen ili upotrijebljen. Zbog različitosti aplikacija i primjena, svi podaci u BT arhitekturi ne prolaze sve slojeve. Primjer tome su usluge telefonije (TCS) ili protokola za otkrivanje usluga (SDP) koji ne zahtijevaju obradu podataka od slojeva poput TCP/IP-a.

2.1. Radio

Bluetooth uređaji rade na globalno dostupnoj i besplatnoj ISM frekvenciji (*industria, scientific and medicinal*) od 2.4 GHz. Međutim, na toj frekvenciji su standardizirane i WLAN (*wireless local area networking*) aplikacije i drugi RF odašiljači što rezultira da je frekvencija od 2.4 GHz dosta zauzeta i prometna. Zato Bluetooth koristi nekoliko tehnika da bi se komunikacija odvijala regularno: frekvencijske skokove, prilagodljivu kontrolu snage i kratke pakete podataka. *Frequency hopping* je izumio Austrijanac Hedy Lamarr za vrijeme II svjetskog rata kao mehanizam za robusnu komunikaciju i sigurnost prijenosa, što je vrlo korisno i u Bluetooth-u. Uvijek je moguće da radio kanal bude trenutno blokiran od interferirajućeg izvora (kao što je već spomenuto za ISM). Iako Bluetooth podržava retransmisiju izgubljenih paketa na istom kanalu, veća je efikasnost ponovnog slanja na novom kanalu, za koji je manje vjerojatno da je blokiran. Također, nekoliko aktivnih pikomreža može biti unutar opsega od drugih pikomreža pa svaka neovisno mijenja frekvenciju pomoću *pseudo random sequence* bazirane na pojedinom *piconet ID/access* kodu, u svrhu minimiziranja kolizija. To je posebno bitno jer svi Bluetooth uređaji imaju samo 79 kanala na kojima rade. U uredu ili javnom okruženju, broj aktivnih uređaja može brzo doseći ovaj limit.

Širina frekvencijskog pojasa od 83.5 MHz je podjeljena na kanale širine 1MHz, svaki šalje signale brzinom 1M simbola/sec sa izabranom modulacijom GFSK (*Gaussian frequency shift keying*, podaci su predstavljeni kao pomaci u frekvenciji, za opis pogledati u [5]) što iznosi 1 Mb/sec. Binarno 1 predstavlja pozitivnu frekvencijsku devijaciju od nominalne noseće frekvencije, dok binarna 0 predstavlja negativnu devijaciju.



Slika 5. Jednostavni radio sistem sa direktnom vezom na baseband

Bluetooth uređaji su podijeljeni u tri razreda prema najvećoj izlaznoj snazi predajnika. Kontroler snage se koristi kako bi se ograničila i optimizirala izlazna snaga ovisno o zahtjevima uređaja:

1. 100 mW (20 dBm) – 1 mW (0 dBm) (maksimalna izlazna snaga – min. izl. snaga)
2. 2.5 mW (4 dBm) – 0.25 mW (-6 dBm)
3. 1 mW (0 dBm) - /

Bluetooth je specificiran da radi sa maksimalnom BER (*bit error rate*) od 0.1%. Na slici 5. je prikazana blok shema tipičnog Bluetooth radio sistema. Kontrolne linije (TxOn, PaOn, VcOn, RxOn) su vremenski vođene od unutarnjeg brojila zajedno sa *receive / transmit selection* signalom (Rx/nTx) koji indicira da li se vrši prijem ili slanje. Podaci se dobivaju na linijama (TxData) kontrolirani od *baseband transmit clock* (TxClk). Za slanje podataka služi (RxData) u taktu koji daje (RxClk). Frekvencijski skokovi se vrše uz pomoć (ChanNo).

Bluetooth radio je sa gledišta arhitekture sustava, najniža Bluetooth komponenta (sloj sustava) koja definira osobine i karakteristike fizičkog uređaja s kojim se uspostavlja bežična komunikacija. Sa samog fizičkog gledišta, radio se može promatrati kao radio-predajnik koji operira na određenim frekvencijskim područjima. On ima svoje karakteristike te svoje operacijske modove (načine rada).

2.2. Baseband

Na početku treba razjasniti razliku između *link controllera* i *basebanda*. Link controller (LC) je odgovoran za prenošenje razine veze kroz trajanje od nekoliko paketa podataka u cilju odgovora komandama višeg nivoa *link managera* (LM). Lokalni udaljeni LC entiteti će uspostaviti proces slanja paketa kad je veza jednom uspostavljena od strane LM. Fizički sloj u OSI standardu je ovdje predstavljen sa *radio* i *baseband* slojevima. *Radio* komunicira između *on – air* kanala i digitalnog basebanda koji formatira podatke prosljeđene od LC i služi za robustan i siguran prijenos, kao i za primanje podatka i prosljeđivanje višim dijelovima prijenosnog stoga. Baseband je odgovoran za kodiranje i dekodiranje kanala, nisku razinu vremenske kontrole i upravljanje linkom unutar domene od jednog prijenosa paketa podataka.

Svaki Bluetooth uređaj ima 48 bitnu IEE MAC adresu poznatu i kao *Bluetooth device address* (BD_ADDR). MAC adresa se sastoji od *Non – significant address part* (NAP), *Upper address part* (UAP) i *Lower address part* (LAP) kako slijedi:

- BD_ADDR [47:32] – NAP [15:0] koristi se za enkripciju LFSR (*Linear feedback shift register* – služi, specijalno kod basebanda za enkripciju, generaciju *Header error check* i generaciju pseudošuma)
- BD_ADDR [31:24] – UAP [7:0] koristi se za inicijalizaciju HEC i CRC izračuna i za frekvencijske skokove
- BD_ADDR [23:0] – LAP [23:0] koristi se za generiranje sinkronih riječi i frekvencijske skokove.

U svom radu, Bluetooth uređaji koriste generator takta radi vremenskog usklađivanja.

Native clock: To je takozvani prirodni sat koji je izveden sklopovski i u stvari je oscilator koji služi za određivanje vrijednosti ostalih satova u uređaju.

Bluetooth clock: Služi za određivanje sekvenci skokova pri komunikaciji između BT uređaja kao i definiranju *vremenskih odsječaka* radio-prijemnika. Određuje se tako da se vrijednosti prirodnog sata doda određeni pomak (offset) koji ovisi o prirodnom satu glavnog (master)

uređaja u vezi te služi za sinkronizaciju svih uređaja u mreži. Rezolucija BT sata je 312.5 us (polovina rezolucije *vremenskog odsječka L*). Vrijednost sata je predstavljena kao 28 bitni broj kojem treba oko 24 sata da bi se vratio na nulu.

Estimated clock: Koristi se tijekom *prozivanja (paging-a)*, a radi ubrzanja ostvarivanja veze. Njime se pokušava aproksimirati "otkucaj" prirodnog sata uređaja kojeg se proziva.

Master clock: Vrijednost ovog sata se šalje od *master* uređaja ostalim uređajima u pikomreži kako bi svojim prirodnim satovima dodali offset vrijednost te se sinkronizirali prema glavnom uređaju. Također, vrijednost ovog sata koristi svaki uređaj u piconetu kako bi slao ili primao podatke (dobivao slobodne vremenske odsječke).

U radu *master sata* definiraju se i sljedeća vremena: *CLK0*, *CLK1*, *CLK2*, i *CLK12* koja se događaju svakih 312.5us, 625us, 1.25ms, i 1.28 sekundi respektivno. Ova vremena su jako važna jer definiraju kada master može slati podatke podređenom uređaju (slave), kada slave može slati podatke master-u, te kako se frekvencijski skokovi mijenjaju kada uređaji obavljaju operacije **upita** (inquiry, inquiry scan) i **prozivanja** (paging, page scan).

2.3. Piconet i Scatternet



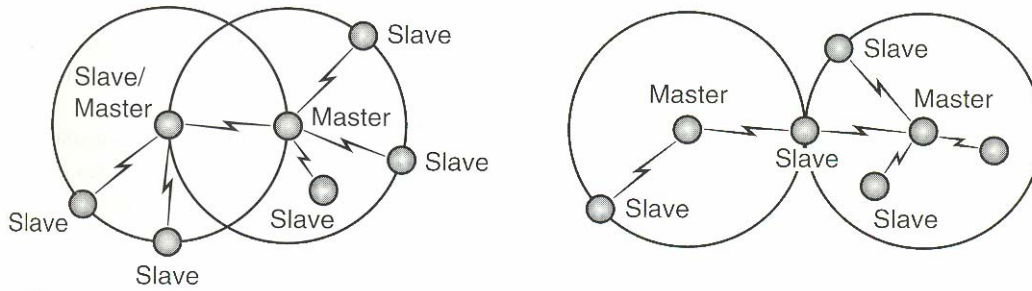
Slika 6. *Point to point* i *point to multipoint piconet*.

- **Master** – uređaj koji inicira razmjenu podataka
- **Slave** – uređaj koji odgovara na upit mastera.

Skup *slave* uređaja povezanih sa jednim određenim *masterom* naziva se *piconet*. Svi uređaji na *piconetu* slijede frekvencijske skokove i *timing* koje diktira *master*. Postoje različiti tipovi *piconeta*:

- Samo sa jednim *slave* uređajem (*point to point*)
- Sa više *slave* uređaja (*point to multipoint*).

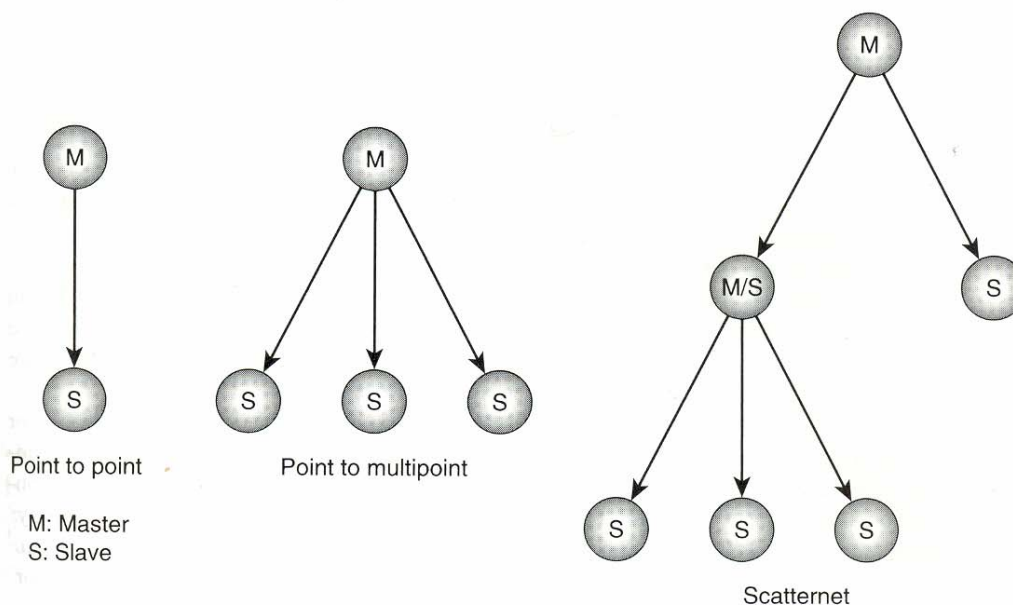
Slave uređaji u *piconetu* imaju vezu samo na *master*, odnosno, nema direktne veze između *slave*-ova. Specifikacija limitira broj *slave* uređaja na sedam. Veći broj uređaja priključenih na mrežu postiže se povezivanjem *piconeta* u *scatternet*.



Slika 7. Scatternet

Svaki uređaj može biti i master i slave, ali ne u istom trenutku. Kod komunikacije, slave se prilagođava timingu mastera i sinkronizira se. Master prvi šalje zahtjev u kojoj se nalazi adresa slave-a. Nakon 625 us će oba uređaja biti na kanalu *radio* sloja (Ch n) na kojem je emitirao master ili će skočiti na (Ch (n+1)) i sada slave-u ne samo da je dozvoljeno emitirati, već mora odgovoriti da li je «razumio» zadnji paket ili ne, te će ponovno oslušivati da li master njega proziva. Master sada može poslati nekome drugome paket ili ne mora nikome. Svaki uređaj «frekvencijski skače» jednom po paketu. To je osnovna zamisao cijelog Bluetootha i to omogućava sljedeće:

- Sigurnost – budući da su frekvencijski skokovi bazirani na pseudoslučajnoj sekvenci adrese mastera
- Pouzdanost – ako bi se zbog interferencije sa drugim uređajem paket na Ch (n) izgubio, malo je vjerojatno da će do toga doći i na Ch (n+m), gdje je n+m udaljenost od n zajamčena sa *pseudo random hopping* algoritmom.



Slika 8. Usporedba piconet - scatternet

Svi paketi imaju jednake headere i kontrolne podatke, pa su onda *multi – slot* paketi (kroz više vremenskih odsječaka) efikasniji za prijenos. Sposobnost da se uređaji mijenjaju ukrug (kao i da mijenjaju svojstvo da li su master ili slave) je posebnost piconeta i to čini Bluetooth različitim od drugih žičanih veza. Zato se takve mreže nazivaju i *ad – hoc* mrežama. LC u sebi ima ugrađene posebne operacije pomoću kojih detektira nove uređaje u dometu i lagano uspostavlja kontakt s njima. Isto tako, zbog kretanja uokolo, pojedini uređaji mogu ostati

izvan dometa i izgubiti kontakt sa piconetom. Zato svaka veza ima *supervision timeout* koji osigurava da se takve veze prekinu.

Kao i mnogi drugi komunikacijski protokoli, i Bluetooth ima sinkronizaciju većine operacija u skladu sa generatorom takta u stvarnom vremenu. Svaki uređaj ima svoj vlastiti neovisni brojač kojim kontrolira operacije (spojen je na CLKN). Ako uređaj radi u master modu, tada koristi i svoj CLKN, dok u slave modu se mora sinkronizirati sa svojim masterom.

2.4. Fizičke veze

Jednom kada je veza između mastera i slave-a uspostavljena, postoje dva osnovna tipa podatkovnih paketa kojima se može odvijati razmjena:

- ACL (*asynchronous connection-less*)
- SCO (*synchronous connection oriented*).

ACL veza postoji između mastera i slave-a čim se uspostavi kontakt. Master može imati veći broj ACL linkova prema većem broju slave uređaja u bilo kojem trenutku, ali postoji samo jedan link između bilo koja dva uređaja. Prema tome, master ne šalje uvijek prema istom slave-u. Odluku o tome kome će slati i od koga će primiti, master bazira na *slot by slot* osnovi. Slave može samo odgovoriti sa ACL paketom u sljedećem slave to master odsječku ako je prethodno bio prozvan u maser to slave odsječku. Postoje i broadcast ACL paketi koji ne adresiraju određen slave, nego se šalju svima. Ova konekcija je asinkrona te se ostvaruje između master-a i svih ostalih slave uređaja u piconetu. Podržava simetrične i asimetrične, point-to-multipoint veze te veze sa prospajanjem paketa. ACL veza se ostvaruje u onim vremenskim odsječcima koji nisu rezervirani od SCO veze (koja samim time ima prednost te osigurava određenu razinu QoS - Quality of Service, kvaliteta usluge). Vrsta konekcije koja se ovdje odvija je klasična *packet - switching* konekcija (prospajanje paketa). Brzine ovog tipa veze kreću se do max. 721 kbit/s u jednom i 57.6 kbit/s u drugom smjeru. Slijede neke važne značajke implementacije ove vrste veze kod BT-a:

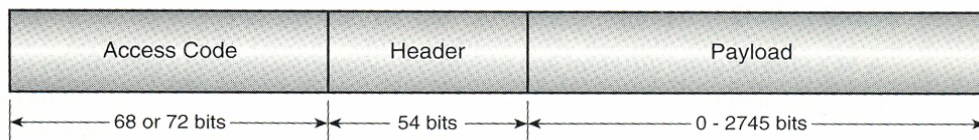
- Samo jedna ACL veza smije postojati između master-a i slave-a
- Slave može odgovoriti ACL paketom u sljedećem slave-to-master vremenskom odsječku jedino ako je bio prozvan od master-a prethodnim master-to-slave odsječkom.
- Bilo koji ACL paket koji specifično ne adresira nijedan slave uređaj, smatra se kao broadcast paket te ga čitaju svi uređaji u piconetu.
- Izgubljeni ili pogrešno primljeni paketi se mogu ponovno slati (dozvoljena retransmisija ARQ - Automatic Retransmission Query shemom)
- ACL paket se može protezati u više vremenskih odsječaka. Ako se desi da paket zauzima više vremenskih odsječaka (multi-slot paket), onda se njegov kompletni prijenos odvija na istoj frekvenciji.

SCO je simetrična veza između mastera i slave-a s periodičnom izmjenom podataka u obliku rezerviranih slotova. Time se osigurava *circuit – switched* veza koja je pogodna za prijenos podataka u stvarnom vremenu, npr. audio podataka. Master može imati do 3 linka prema jednom ili različitim slave-ovima. Zbog ovisnosti o vremenu, nema retransmisije paketa. Takt slanja paketa daje (TSCO). Slave-u je uvijek dozvoljeno odgovoriti sa SCO paketom na jednom od slotova rezerviranih upravo za odgovore. Ova je veza sinkrona i ostvaruje se između master i slave uređaja u piconetu na principu od točke-do-točke. Kada se ovdje kaže da je

veza sinkrona, to znači da master uređaj rezervira vremenske odsječke za komunikaciju u pravilnim vremenskim razmacima. Važne osobine ovog tipa veze su sljedeće:

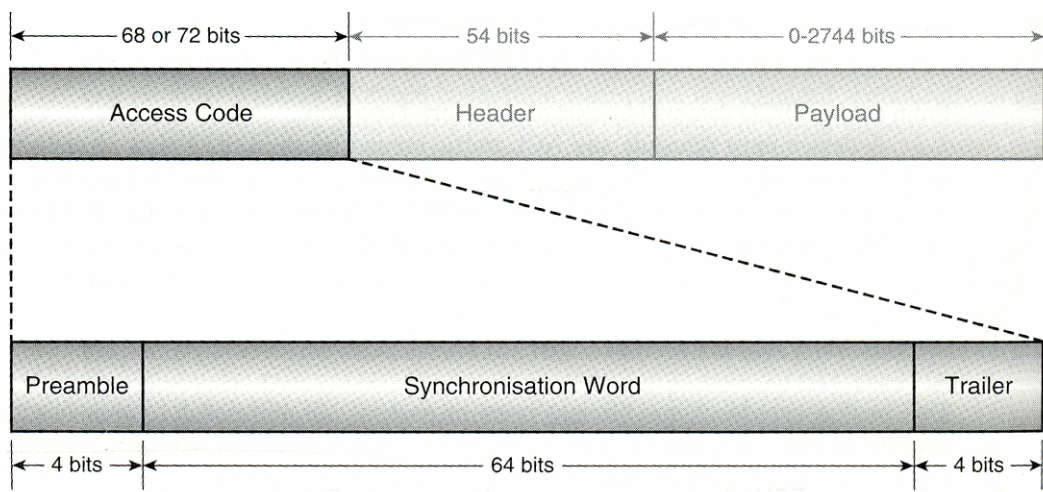
- Master uređaj može istovremeno podržavati tri SCO veze koje mogu pripasti jednom slave-u ili se mogu raspodijeliti na više njih.
- Slave može podržavati do tri SCO veze od istog mastera ili maksimalno dvije veze u slučaju više mastera.
- Paketi u SCO konekciji se nikada ne šalju ponovno (retransmisija), čak i ako su i izgubljeni ili nevaljani.
- Podaci poslani SCO konekcijom ne protežu se na više od jednog vremenskog odsječka (slot-a).

2.5. Struktura paketa



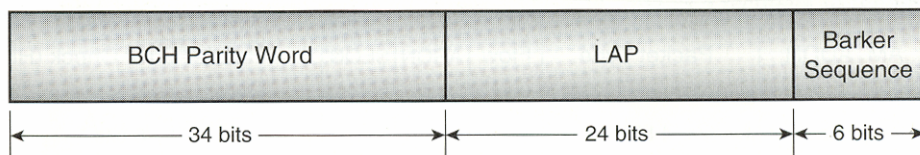
Slika 9. Struktura Bluetooth paketa

Kao što je prikazano na slici 9., paket se sastoji od polja *Access code*, *Header* i *Payload*. Access code sadrži adresu pošiljaoca, a ima i ulogu detekcije prisutnosti paketa u pojedinom uređaju. U zaglavlju se nalaze kontrolne informacije o paketu, vezi, kao i adresa odredišta. Payload sadrži poruku (informaciju) o trenutnom stanju, ako se radi o višem sloju protokola (LM ili L2CAP), ili jednostavno podatke.



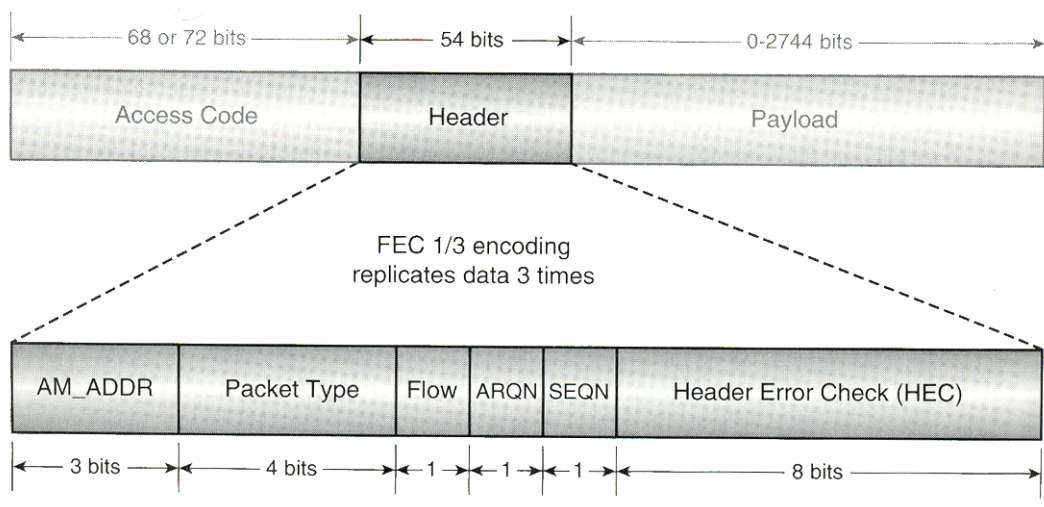
Slika 10. Struktura *access code-a*

Access code je omeđen sa 4 bita na svakoj strani koji označavaju graničnike (koristi se bitovna kombinacija 0101 ili 1010, ovisno o tome kako počinje i završava sinkronizacijska riječ – ukoliko počinje sa 0 početak se označava sa 0101, a za 1 je obrnuto; isto vrijedi i za graničnik koji označava kraj). Preostala 64 bita tvore sinkronizacijsku riječ (slika 11.). Ona se sastoji od 24 bita *Lower address part* (LAP) adrese Bluetooth uređaja. Na nju se prema posebnom algoritmu dodaje i *Barker sequence*, a BCH (Bose Chaundhuri Hocquenghem)



Slika 11. Struktura polja *synchronisation word*

služi za provjeru pariteta. Svaki prijemnik gleda da li je dotični paket namijenjen za njega, i ako je, očekuje dalje header. Na ovaj način slave će primiti samo one pakete koji su poslani od njegovog mastera unutar masterovog vlastitog piconeta. Ako su dva piconeta blizu, svaki skup slave-ova će primiti samo pakete sa njegovog piconeta, koji ima vlastitu sinkronizacijsku riječ.

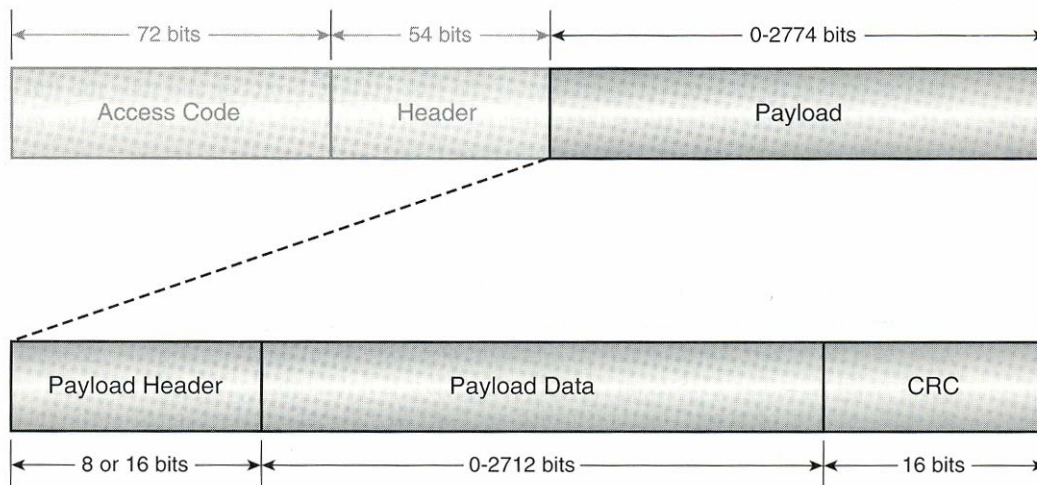


Slika 12. Struktura polja *packet header*

Packet header se sastoji redom od:

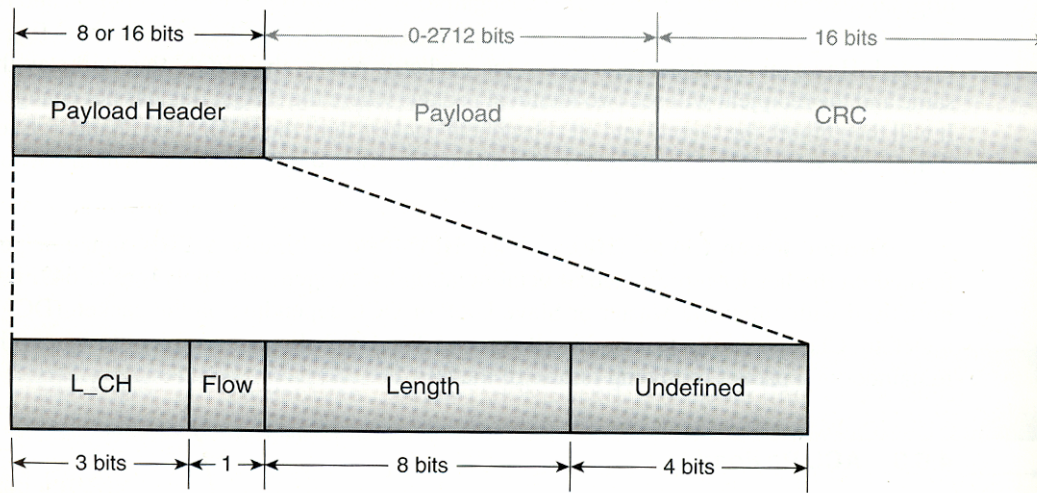
- AM_ADDR (Active member address) koji master dodjeljuje slave-u. To omogućuje masteru da razlikuje do 7 različitih slave-ova. Ukoliko su sva 3 bita na nuli, to znači da se šalje broadcast paket (znači svima).
- Packet type definira koja vrsta prometa se odvija na vezi (SCO, ACL, NULL, POLL), koliko slotova treba očekivati payload i vrstu greške (ako je došlo do nje).
- Zastavice *Flow* (označava da je buffer uređaja pun i da ne može više primiti podatke), ARQN (primatelj označava da je paket primio) i SEQN (postavlja se svaki put kada se šalje novi paket) HEC (Header error check) polje je u stvari CRC funkcija koja služi za kontrolu grešaka headera.

ACL Payload (slika 13.) se sastoji od *Payload headera*, *Payload* polja i CRC polja. Payload header (slika 14.) se sastoji od L_CH polja koje označava da li se radi o početku komunikacije ili nastavku L2CAP ili LMP poruke; *Flow* zastavice koja označava transfer na nivou L2CAP i duljinu samog polja. Istu strukturu paketa ima i SCO prijenos (slika 15.), samo što su kod njega neka polja redundantna.

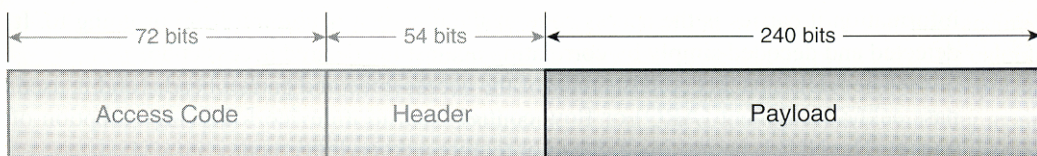


Slika 13. ACL payload polje

Postoje i specijalni paketi – ID, NULL, POLL i FHS. ID paket se sastoji samo od Access code-a i služi samo za uspostavu veze. NULL paket ima samo Access code i Packet header i služi za prijenos potvrde i to u situacijama kad prijemnik nema što odgovoriti, ali mora poslati potvrdu o regularnom primitku. NULL paket ne mora biti potvrđen. POLL ima slični izgled i funkciju s razlikom da POLL mora biti potvrđen. FHS (Frequency hop synchronisation) paket sadrži sve potrebne podatke za obavljanje frekvencijskog skoka.



Slika 14. ACL payload header



Slika 15. SCO struktura paketa

2..6. Logički kanali

Logički kanali se baziraju na fizičkim SCO i ACL vezama i koriste se za prijenos podatkovnih i kontrolnih informacija između mastera i slave-a.

LC (Link control) se prenosi u packet headeru i sastoji se od ARQ, SEQ i ostalih kontrolnih podataka. Ovaj kanal služi kao osnova za održavanje i kontroliranje veze. Koristi se na razini Link controller-a.

LM (Link manager) se nalazi u payload polju i sadrži kontrolne podatke koji se izmjenjuju između dva LM entiteta. Označava ga L_CH = 11 vrijednost. Nalazi se iznad LC nivoa.

UA / UI (User asynchronous / isochronous data) se također prenosi u ACL polju i to kada je L_CH = 10 (za pakete kojima se označava početak veze) i L_CH = 01 (ukoliko se radi o fragmentaciji kroz nekoliko baseband paketa). Prenose asinkrone i snkrone korisnike podatke.

US (User – synchronous data) služi za prijenos korisničkih informacija u SCO vezi i predstavlja jedini tip kanala koji može biti prekinut u transmisiji u slučaju podataka višeg prioriteta (sa nekog drugog kanala - LM, UA ili UI).

2.7. Link controller

Link control sloj se brine o otkrivanju upravljačkog uređaja i, jednom kada su uspostavljeni, održavanja različitih *on – air* veza. Link controller i Baseband sloj su međusobno usko povezani i stoga posebna pažnja mora biti posvećena kod dizajniranja Bluetooth uređaja da bi se omogućilo optimalno komuniciranje i razdvajanje na slojeve. Link control protokol se prenosi u LC kanalu i zadužen je za mehanizme održavanja veze. U tome najbitniju ulogu ima ARQ (Acknowledgement / request) shema koja služi za retransmisiju podatka nad kojima se dogodila pogreška. U svakom packet headeru se nalazi već spomenuta zastavica ARQN koja sadrži informaciju o prethodno primljenom paketu. Ako je postavljena (ACK), signalizira da je prethodni paket primljen i ispravno dekodiran. U protivnom (NAK) govori da je došlo do greške. Greške se javljaju uglavnom kod CRC-a ili HEC-a, ili ako se negdje zagubi odgovor slave-a masteru.

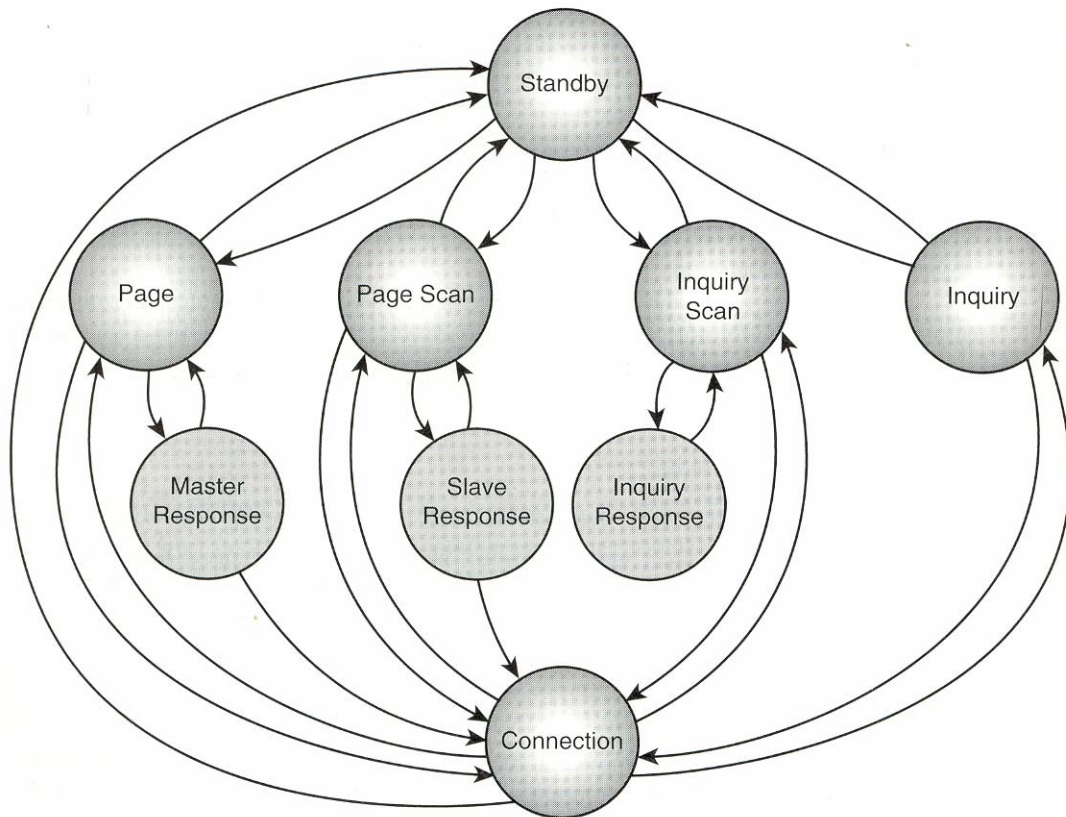
2.8. Stanja link kontrolera

U svakom trenutku se Bluetooth uređaj nalazi u jednom od brojnih stanja. Dva glavna stanja su *standby* i *connection*.

Standby - ovo je početno stanje BT uređaja. U ovom stanju uređaj zahtijeva malo energije, jer osim rada prirodnog sata, ne ostvaruju se nikakve veze sa drugim uređajima. Da bi se ostvarila veza između dva BT uređaja, potrebno je za BT uređaj koji je u standby stanju, da povremeno osluškuje prostor oko sebe u potrazi za drugim BT uređajima. U ovom stanju postoji i nekoliko podstanja koja služe za ostvarivanje veze:

- *Inquiry* - Kontroler ulazi u ovo stanje onog trenutka kada želi otkriti nove uređaje oko sebe. Da bi to postigao on svim uređajima u blizini šalje (broadcast načinom) inquiry pakete (ID pakete) koji u sebi sadrže IAC (inquiry access code). Odlazak u inquiry

state stavlja uređaj u ulogu potencijalnog mastera (za sada se naziva source). Inquiry stanje se može nazvati i stanjem prozivanja i prvi je korak pri komunikaciji dva uređaja.



Slika 16. Dijagram stanja *link controller-a*

- *Inquiry scan* - Kontroler ulazi u ovo stanje kada želi primiti inquiry pakete. Ako uređaj nije u ovom stanju dok ga source proziva, daljnja uspostava veze nije moguća. Uređaj koji odlazi u inquiry scan state definira se kao potencijalni slave uređaj (za sada samo destination).
- *Page* - Ovo podstanje je svojstveno samo source uređaju. Uređaj u ovom stanju konstantno odašilje ID pakete sa DAC-om (Device access code) nekog određižnog uređaja, a s namjerom ostvarivanja konekcije. Ako u nekom trenutku source uređaj dobije odgovor tada se prebacuje u *master response* stanje.
- *Page scan* - Dok je source uređaj u page stanju, destination uređaj mora biti u page scan stanju ne bi li se uspješno obavilo povezivanje ta dva uređaja. Destination uređaj se budi iz standby stanja svakih 11.25 ms (scan window) te prelazi u ovo stanje i osluškuje ID pakete ne bi li naišao na svoj DAC u njima. Ako uređaj dobije takav paket, odlazi u *slave response* stanje.
- *Slave Response state*
 1. Ako je uređaj u ovom stanju, znači da je dobio od source uređaja svoj AC (u ID paketu). Destination uređaj šalje svoj DAC source uređaju. Prelazi se na 1. dio master response stanja.

2. Kada destination uređaj primi FHS paket, on opet (kao odgovor da je primio ovaj paket) šalje ID paket source uređaju.
 3. Destination uređaj koristi podatke poslane u FHS paketu da se prebaci na parametre koje mu je zadao master uređaj. Nakon toga, destination uređaj se deklarira kao slave uređaj u master-slave vezi.
- *Master response state*
 1. Ako je uređaj u ovom stanju, znači da je dobio od destination uređaja odgovor (DAC destination uređaja u ID paketu). Source uređaj sada šalje FHS paket destination uređaju. Ovaj paket je važan jer sadrži informacije potrebne za sinhronizaciju dvaju uređaja. Prelazi se na 2. dio slave response stanja.
 2. Nakon primanja drugog ID paketa od strane destination uređaja, source uređaj se deklarira kao master uređaj u master-slave vezi.

Na kraju, važno je napomenuti način na koji slave uređaj osluškuje prostor oko sebe u potrazi za ID paketima (pogotovo ako se zna da se cijelo vrijeme pri komunikaciji dva uređaja koristi neka hopping frekvencijska sekvenca koju je odredio master uređaj). Svakih 1.28 sekundi, master uređaj iz page stanja prelazi u page scan stanje ne bi li oslušnuo da li mu možda slave uređaj odgovara na njegov upit za ID paketom. Važno je pri tome napomenuti da master uređaj ne zna frekvencijsku sekvencu kojom slave uređaj šalje svoj odgovor (page response stanje). Zato se koristi *page train* metoda kojom master uređaj svakih spomenutih 1.28 sekundi pretražuje određenu frekvencijsku hopping sekvencu (od moguće 32) s kojom slave uređaj šalje odgovor.

Connection -u ovo stanje uređaji ulaze ako je ostvarivanje veze uspješno obavljeno te ako na master-ov POLL paket slave odgovori bilo kojim paketom. To znači da se slave uređaj uskladio sa master uređajem. Podstanja ovog stanja mogu biti:

- *Active* - uređaj u ovom modu rada aktivno sudjeluje u komunikaciji na kanalu. Master kontrolira promet na kanalu između ostalih slave uređaja i sebe, te se brine da svi uređaji ostanu sinkronizirani. Aktivni slave uređaj, u master-slave vremenskim odsječcima osluškuje pakete, ne bi li naišao na neki koji je njemu upućen (ima njegovu adresu). Ako paket ne sadrži njegovu adresu, slave uređaj može spavati neko vrijeme (do nove transmisije master uređaja).
- *Sniff* - ovo je prvi od *power-saving* načina rada (načini rada predviđeni za štednju energije) koji ujedno i najmanje šteti energiju. Štednja energije je izvedena tako da se poveća vrijeme vremenskog perioda u kojem BT uređaj osluškuje mrežu (piconet). Vremenski period je programabilan i ovisi o aplikaciji. Povećanjem vremenskog perioda se omogućava uređaju da vrijeme koje nije aktivan na piconet-u iskoristi za eventualnu konekciju sa drugim piconet-om. U ovom stanju su podržani i SCO i ACL paketi.
- *Hold* - ovo je još jedan od power-saving načina rada u koje slave uređaj može ući ili biti stavljen od master uređaja. Uređaj u ovom modu podržava SCO pakete, no ne i ACL pakete.

- *Park* - kada uređaj odlazi u ovo stanje on se odriče svoje aktivne adrese člana (AM_ADDR), dobiva privremenu 8-bitnu *park adresu* člana (PM_ADDR), te izlazi iz piconet-a. To ipak ne znači da se on odriče konekcije sa piconet-om. Naprotiv, on ostaje sinkroniziran sa njim, no ne može slati ni primiti pakete (osim broadcast paketa). Kako je PM_ADDR 8 bitna vrijednost, dozvoljava se 255 parkiranih BT uređaja. Iako su aktivnosti uređaja u ovom stanju svedene na minimum, uređaj ipak periodički osluškuje prostor oko sebe za eventualnim broadcast paketima poslanih od master-a (npr. resinkronizacija). Za ponovno ulaženje uređaja u piconet, potrebna je slobodna AM_ADDR (ima ih 7). Ovo je najštedljiviji način rada BT uređaja.

2.9. Audio

Glavna aplikacija Bluetooth-a je prijenos audio informacije. Audio podaci se prenose preko SCO kanala. Standard specificira da bi se audio trebao prenositi putem SCO kanala, a podaci putem ACL. Najbolji način procjene kvalitete prijenosa audio podataka je subjektivno, onako kako ga čujemo, jer su različite sheme usporedbi sa ljudskim uhom, još nedovoljno razvijene. Za visokokvalitetni audio prijenos se podaci komprimiraju (najpoznatiji način MP3).

SCO veze se uspostavljaju pomoću inicijalnog ACL linka, a potom se paketi izmjenjuju pomoću SCO. Bluetooth posjeduje 3 različite tehnike kodiranja: *Log PCM A – law*, *Log PCM u – law* i CVSD. Log kodiranje kompresira ulazne podatke pomoću logaritamske funkcije (osigurava minimalne gubitke u kvaliteti zvuka, gotovo nečujne ljudskom uhu). CVSD (Continuous variable slope data modulation) je kompleksnija tehnika koja kvantizira razliku između dva uzorka, a ne cijele amplitude i zato bolje aproksimira ulazni signal.

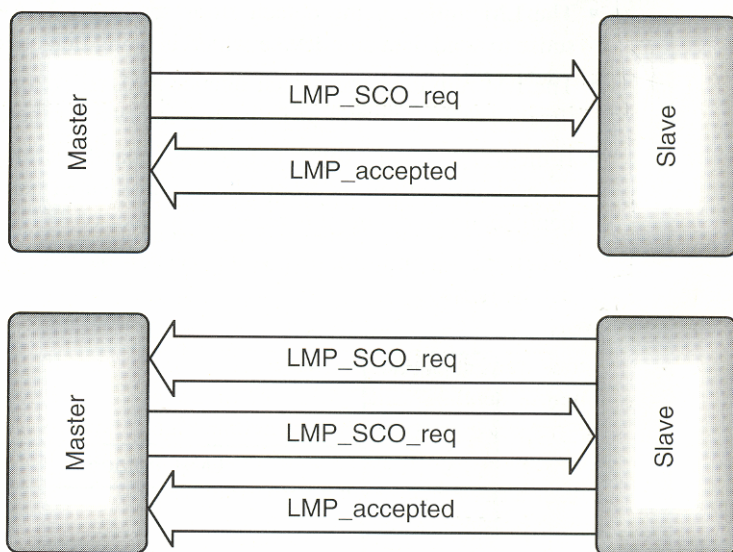
Audio promet nije zaštićen niti CRC kodom niti retransmisijom koja naravno nije ni moguća zbog rada u stvarnom vremenu. Iako je CVSD prilično robustan (otporan na greške) uvijek se može dogoditi da dođe do pogreške koja će paket učiniti neupotrebljivim. Najjednostavniji način za rješavanje greške je resetiranje razine kodeka (dobivanog CVSD-om) na njegove početne vrijednosti sve dok ne dođe novi paket. Algoritam «pokriva» sam sebe vrlo brzo i efekt na zvuk je zanemariv.

Audio je jedan od osnovnih komponenata BT uređaja koji je sposoban koristiti do 3 dvosmjerna audio kanala istovremeno. Zvukovne mogućnosti su vrlo prikladne za primjenu u aplikacijama koje zahtijevaju visoku kvalitetu zvuka, kao npr. komunikacija između mobitela i *headseta*.

2.10. Link manager

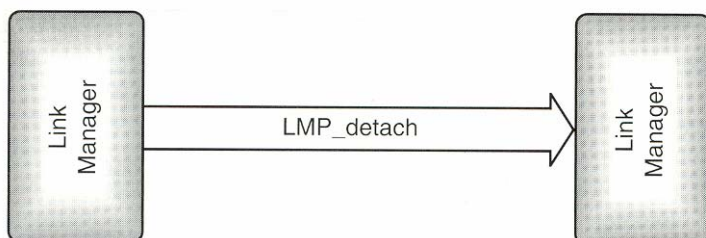
Link manager (LM) služi da bi preveo komande od HCI (host upravlja BT uređajem pomoću HCI komandi) u operacije na baseband nivou i to:

- Priključivanje slave uređaja na piconet i dodjeljivanje aktivne adrese
- Prekidanje veza za u svrhu odspajanja slave-a od piconeta
- Konfiguracija linkova uključujući zamjene master – slave
- Uspostava ACL (data) i SCO (glas) veza
- Stavljanje veza u *low – power* načine
- Kontroliranje test načina



Slika 17. Slijed LMP poruka kod uspostavljanja SCO veze

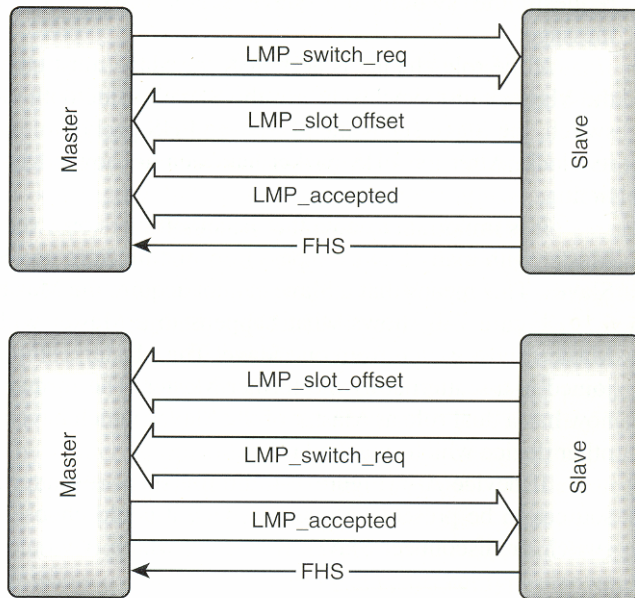
BT LM komunicira s LM na drugom BT uređaju koristeći LMP (Link management protocol). Svaka LMP poruka počinje sa TID (bit transaction identifier) koji je 0 ako je prijenos inicirao master, a 1 ako je inicirana od strane slave-a. Nakon toga slijedi 7 bitni OpCode (operacijski kod) koji opisuje tip poruke. Nakon njega slijede parametri poruke.



Slika 18. LMP prekidanje veze

LM je odgovoran za uspostavljanje baseband veza koje povezuju BT uređaje. LM uspostavlja ACL vezu kontrolirajući baseband; LMP poruke mogu biti korištene za uspostavljanje SCO veze (slika 17.) uz postojeću ACL. Link controller sloj mora uspostaviti vezu prije nego dođe

do razmjene LMP-a. LMP ima 30 sekundni vremenski istjek između slanja poruke i odgovora na nju. Ako je vrijeme isteklo, za vezu se pretpostavlja da je mrtva (slika 18.).



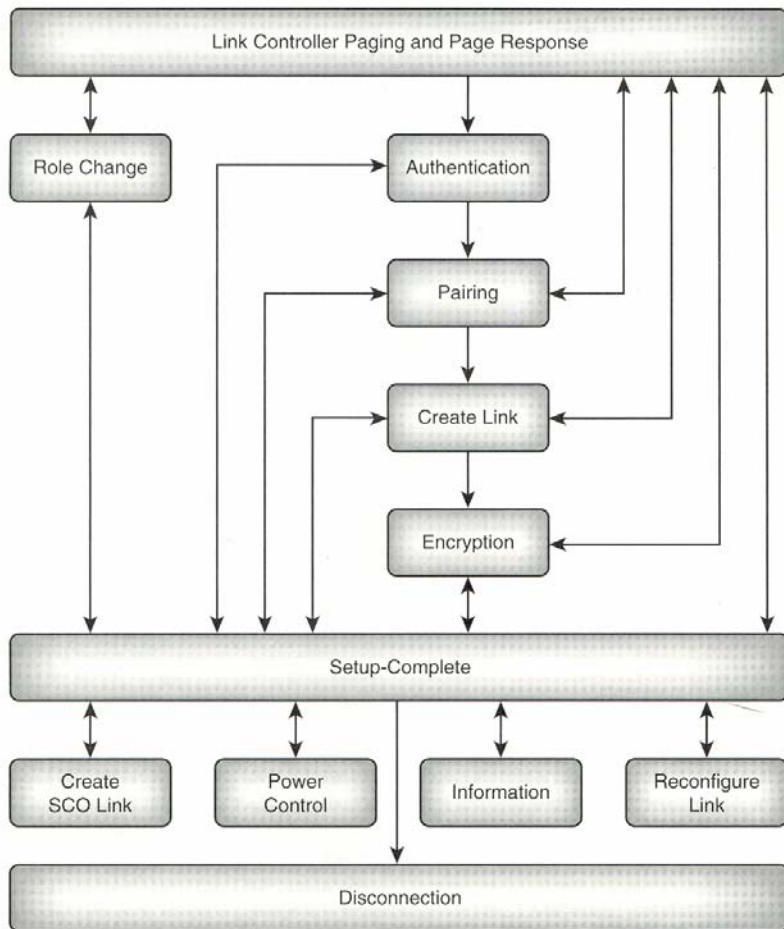
Slika 19. Slijed poruka kod izmjene uloga

U bilo kojem trenutku ako bilo master bilo slave želi prekinuti vezu šalje *LMP_detach* naredbu (kao na slici). Uzroci prekida mogu biti:

- Korisnik je prekinuo vezu
- Slab prijem
- Power off.

Kad se slave odspaja od piconeta, master može ponovno upotrijebiti slave-ov *AM_ADDR* za identifikaciju novog slave-a.

Kod zamjene uloga (slika 19.), uređaj koji želi zamjenu šalje *LMP_switch_req* signal onom drugom. Razlika je jedino u tome da ako slave želi inicirati zamjenu mora prije poslati *LMP_slot_offset* (sadrži *BD_ADDR* slave-a koja će postati adresa mastera nakon zamjene i *slot offset*). Za vrijeme zamjene šalje se i *FHS* paket koji zajedno sa *slot offsetom* služi za usklađivanje (sinkronizaciju) satova.



Slika 20. Pregled svih LMP operacija

2.11. Host controller interface (HCI)

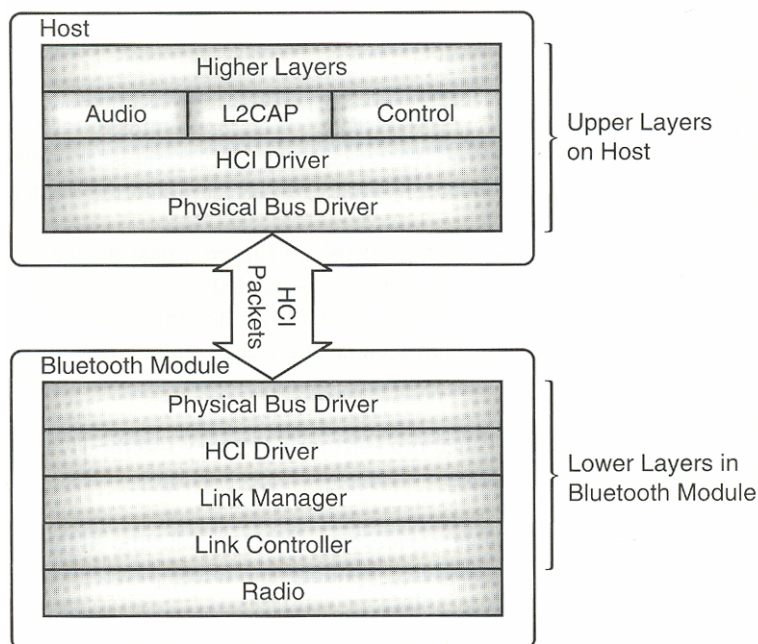
Neki BT sistemi imaju baseband i LM slojeve pokrenute na jednom procesoru koji ima više nivoa poput L2CAP, SDP, RFCOMM, a aplikacije su pokrenute na drugom procesoru domaćina (host). U sistemima kad su viši nivoi pokrenuti na procesoru od host-a, a niži nivoi na BT uređaju potrebno je sučelje između viših i nižih slojeva. BT uređaji zato imaju u sebi implementiran HCI koji omogućava pravilnu komunikaciju i povezivanje u takvim slučajevima (primjer može biti BT PCMCIA kartica za PC koja može biti implementirana sa LM i baseband slojevima na samoj kartici, a višim slojevima na PC procesoru; još jedan primjer je headset). Razdvajanje na više i niže slojeve se može vršiti zbog:

- Domaćini, poput PC-a, imaju dovoljno slobodnog kapaciteta za rukovanje višim slojevima, dopuštajući BT uređaju da ima manje memorije i manje snažan procesor ili DSP, što smanjuje troškove
- Domaćin može «spavati» i po potrebi ga «budi» nadolazeća BT veza (niži slojevi su odgovorni za uspostavljanje veze, a tek kad se ona uspostavi koriste se i viši slojevi).

BT standard definira sljedeće HCI tipove paketa:

- Paketi sa naredbama koji se koriste kod kontrole modula od strane domaćina

- *Event* paketi koje šalje modul da informira host o promjenama u nižim slojevima (događajima)
- Podatkovni paketi za prijenos podatka i glasa između domaćina i modula
- Transportni slojevi koje mogu nositi HCI paketi.



Slika 21. Pozicija HCI u Bluetooth stogu

Naredbeni paketi započinju sa poljem za operacijski kod (OpCode) koji identificira tip komande, nakon njega slijedi polje u koje je upisana duljina parametara u byte-ovima i nakon toga parametri. Podatkovni paketi koriste i SCO i ACL veze. Glavna karakteristika takvih paketa su zastavice PB (koja označava da li su podaci smješteni u više paketa - šalju se u više vremenskih odsječaka) i BC koja označava broadcast pakete. Naravno, postoje i polja koja definiraju vrstu veze, duljinu podataka i same podatke. Event paketi su po strukturi slični naredbenim paketima, samo nose informaciju o događajima (statusu) u nižim slojevima. Spomenut ću samo neke: `HCI_Set_Event_Mask`, `HCI_Command_Status`, `HCI_Command:Complete`, `HCI_Number_Of_Completed_Packets` i dr.

Kontrola toka se koristi u smjeru od računala do HC-a (Host controller) kako bi se izbjeglo prepunjavanje stoga podataka na HC-u sa ACL podacima za udaljeni uređaj koji npr. ne odgovara. Računalo upravlja podatkovnim stogovima na HC-u. Na temelju `Read_Buffer_Size` naredbe, (računalo ju šalje pri inicijalizaciji - određuje maksimalnu veličinu HCI ACL i SCO paketa koje računalo šalje HC-u kao i broj paketa koje HC još može primiti) i *Number of Complete Packets* događaja (određuje broj HCI podatkovnih paketa koji su preneseni, odbačeni ili vraćeni računalu), računalo odlučuje za koju će vezu slati HCI podatkovne pakete. Na temelju *Number of Complete Packets* događaja može se odrediti koliko još paketa stane u podatkovni stog. HCI paketi moraju biti poslani HC-u u poretku u kojem su kreirani na računalu, a u tom poretku ih HC mora prenijeti nižim slojevima. Isto je i s poretkom primljenih HCI paketa od donjih slojeva. Smjer prijenosa može biti i od HC-a do računala, što

određuje naredba `Set_Host_Controller_To_Host_Flow_Control`. Mehanizam kontrole toka je isti uz zamjenu `Read_Buffer_Size` sa `Host_Buffer_Size` naredbom i `Number_of_Complete_Packets` događaja sa `Host_Number_of_Complete_Packets` naredbom. Na naredbu se ne primjenjuje mehanizam kontrole toka. Kada računalo primi *Disconnection complete* događaj, on pretpostavlja da su svi podaci poslani HC-u odbačeni i podatkovni stogovi ispraznjeni. Isto tako ako je omogućen drugi smjer prijenosa, HC će pretpostaviti da će računalo nakon što on pošalje *Disconnection Complete* događaj isprazniti svoje stogove.

HCI je sučelje koje osigurava skup naredbi za pristup BT sklopovlju. U HCI-ju su definirane sljedeće grupe naredbi.

- HCI Link naredbe (naredbe veze) omogućuju host-u da kontrolira veze link sloja sa link slojem drugih Bluetooth uređaj. Obično se to odnosi na naredbe koje Link Manager (LM) izmjenjuje sa drugim LM-om, a koristeći LMP protokol (upravljanje organizacijom piconeta i scatterneteta).
- HCI Policy naredbe utječu na ponašanje lokalnog i udaljenog LM-a. Te naredbe omogućuju host-u da utječe na način kako LM upravlja piconetom. S druge strane, HC omogućuje prilagođavanje mehanizma za podršku više različitih policy implementacija. Na taj način BT modul može podržavati više modela korištenja i isti BT modul može biti integriran u različite BT uređaje.
- Host Controller i Baseband naredbe, Informational naredbe, te Status naredbe omogućuju host-u pristup različitim registrima u Host Controller-u. Informativni parametri su utvrđeni od strane proizvođača BT sklopovlja te su "upisani" u njemu i ne mogu se mijenjati. Ove naredbe pružaju uvid u ove parametre te daju informaciju o BT uređaju i mogućnostima HC-a, LM-a i baseband-a. Parametri statusa daju informaciju o trenutnom stanju HC-a, LM-a i baseband-a. HC može mijenjati sve status parametre.
- Testing naredbe omogućavaju testiranje funkcionalnosti Bluetooth sklopovlja na način da se njima simuliraju određeni scenariji stvarne primjene.

Ovdje ću navesti samo Link control i Policy naredbe, jer su one najvažnije.

NAREDBA		OPIS
□	Inquiry	uzrokuje ulaz BT uređaja u Inquiry mod, kako bi otkrio obližnje BT uređaje
□	Inquiry_Cancel	uzrokuje izlaz BT uređaja i Inquiry moda.
□	Periodic_Inquiry_Mode	konfigurira BT uređaj na automatski inquiry baziran na specifičnom vremenskom periodu.
□	Exit_Periodic_Inquiry_Mode	izlaz iz gornjeg moda.
□	Create_Connection	uzrokuje stvaranje ACL veze sa BT uređajem na temelju BD_ADDR koji je specificiran kao parametar.
□	Disconnect	prekidanje postojeće veze.

□	Add_SCO_Connection	LM stvara SCO vezu koristeći ACL vezu specificiranu parametrom Pokazivač veze (Connection handle).
□	Accept_Connection_Request	za prihvaćanje dolazećeg zahtjeva za vezu.
□	Reject_Connection_Request	za odbijanje zahtjeva za vezom
□	Link_Key_Request_Reply	odgovor na Link key request događaj od HC-a i specificira Link key pohranjen na računalo kao ključ veze za vezu sa BT uređajima specificiranim BD_ADDR-om.
□	Link_Key_Request_Negative_Reply	koristi se za odgovor na Link_Key_Request događaj od HC-a ako računalo nema Link Key za vezu sa drugim BT uređajem.
□	PIN_Code_Request_Reply	odgovor na PIN Code Request događaj od HC-a i specificira PIN kod koji će se koristiti za vezu.
□	PIN_Code_Request_Negative_Reply	kada računalo ne može odrediti PIN za vezu.
□	Change_Connection_Packet_Type	koji paketi se koriste za vezu koja je trenutno uspostavljena.
□	Authentication_Requested	za uspostavu autentikacije između 2 BT uređaja određenih pokazivačem veze.
□	Set_Connection_Encryption	Za omogućavanje/onemogućavanje enkripcije na nivou veze.
□	Change_Connection_Link_Key	oba BT uređaja moraju kreirati novi ključ veze
□	Master_Link_Key	oba BT uređaja povezana pokazivačem veze neka koriste privremene ključeve veze glavnog (master) uređaja ili regularne ključeve veze.
□	Remote_Name_Request	za dobivanje user-friendly imena 2 BT uređaja.
□	Read_Remote_Supported_Features	zahtjeva listu podržanih mogućnosti udaljenog uređaja.
□	Read_Remote_Version_Information	naredba će pročitati informaciju o verziji za udaljeni BT uređaj.
□	Read_clock_offset	dozvoljava računalo da pročita pomak sata udaljenog BT uređaja.

Tablica 1. Link control naredbe

NAREDBA		OPIS
□	Hold_Mode	LM stavlja lokalni ili udaljeni uređaj u hold mode.
□	Sniff_Mode	LM stavlja lokalni ili udaljeni uređaj u sniff mode.
□	Exit_Sniff_Mode	izlazak iz sniff moda za pokazivač veze koja je trenutno u sniff modu.
□	Park_Mode	LM stavlja lokalni ili udaljeni uređaj u Park mode.
□	Exit_park_Mode	
□	QoS_Setup	specificiranje QoS parametara za pokazivač veze.
□	Role_Discovery	za određivanje koju ulogu uređaj ima za određeni pokazivač veze.
□	Switch_Role	zamjena uloge koju trenutno BT uređaj ima za pokazivač veze.
□	Read_Link_Policy_Settings	naredba će pročitati postavke politike veze za pokazivač veze.
□	Write_Link_Policy_Settings	naredba će upisati postavke politike veze za pokazivač veze.

Tablica 2. Policy naredbe

2.12. L2CAP (Logical link control and adaptation protocol)

L2CAP šalje podatke iz viših slojeva BT stoga ili aplikacija, prema HCI, odnosno u slučaju sistema bez domaćina, direktno u LM. Spada u sloj podatkovne veze, gdje omogućuje uspostavljanje spojnih i bespojnih podatkovnih usluga prema protokolima viših slojeva. Funkcije L2CAP-a su:

- dogovaranje između protokola u višim slojevima (i njihovo međusobno usklađivanje), dopuštajući im tim da dijele niže slojne veze
- segmentacija i reasembliranje u cilju prijenosa većih paketa, nego što to dopuštaju niži slojevi
- *group management* koji podržava jednosmjerni prijenos prema grupi drugih BT uređaja
- *QoS (Quality of service)* pružanje informacija o kvaliteti usluga za više slojeve.

Posluživanje gornjih slojeva od strane L2CAP-a se vrši preko L2CAP kanala. Definirana su tri tipa L2CAP kanala: bi-direkcijski signalni kanal koji se koristi za prijenos naredbi; spojna veza za dvosmjernu *point-to-point* komunikaciju; te jednosmjerna bespojna veza koji podržava *point-to-multipoint* komunikaciju omogućavajući BT jedinici da bude u komunikaciji sa grupom drugih uređaja. Za razliku od Baseband layera (na koji se L2CAP nadovezuje) koji podržava SCO i ACL tipove veza, L2CAP specifikacija definirana je samo za ACL veze.

Multipleksiranje protokola - L2CAP mora podržavati multipleksiranje protokola jer baseband sloj ne podržava (nema u formatu svog paketa) polje koje bi određivalo "tip protokola" više razine koji je multipleksiran. Tipovi protokola koje L2CAP mora razlikovati su SDP, RFCOMM, TC (Telephony control).

Segmentacija i ponovno spajanje - ova funkcionalnost L2CAP sloja apsolutno je potrebna za podršku protokola više razine koji koriste pakete veće dužine nego je definirano baseband slojem. U usporedbi sa drugim žičanim fizičkim medijima, podatkovni paketi definirani baseband protokolom ograničeni su u veličini. MTU (Maximum Transmission Unit) koji je povezan sa najvećim payloadom koji baseband podržava, ograničava efikasnu iskoristivost prijenosa za protokole viših slojeva koji su dizajnirani da koriste veće pakete. To znači da "preveliki" L2CAP paket mora biti segmentiran u više manjih baseband paketa prije prijenosa (transmisije). Jednako tako, primljeni baseband paketi moraju biti ponovno spojeni u jedan veći L2CAP paket, držeći se pri tome provjere integriteta.

Kvaliteta usluge (QoS) - L2CAP proces uspostavljanja spoja omogućuje razmjenu informacije koja se odnosi na kvalitetu usluge očekivane između dvije BT jedinice. Svaka L2CAP implementacija mora nadzirati resurse uređaja koje protokol koristi i osigurati da je QoS dogovor poštivan.

Grupe - baseband protokol podržava koncept piconeta, grupe sinkroniziranih uređaja koji međusobno komuniciraju. Također, mnogi drugi protokoli na višim slojevima od baseband sloja omogućavaju rad uređaja u grupama, no ne na isti način kao što je to izvedeno baseband protokolom. L2CAP sloj ovdje uskače i omogućuje efikasno mapiranje grupa viših slojeva na baseband piconet. Takvo mapiranje se naziva *group abstraction*.

Neophodni zahtjevi za L2CAP sloj uključuju:

- jednostavnu i malu nadogradnju (overhead)
- mogućnost korištenja na ograničenim računalnim resursima
- veliku iskoristivost snage (power efficiency)
- zahtjevi za memorijom moraju biti svedeni na minimum
- postizanje visoke propusnosti i prijenosa podataka.

I L2CAP sloj je baziran na konceptu kanala (channels). Svaka krajnja točka (end-point) predstavlja identifikator kanala (Channel Identifier - CID). Identifikatori kanala su lokalna imena koja predstavljaju kraj logičkog kanala na BT jedinici. Dodjela CID-a povezana je sa pojedinim BT uređajem i uređaj može dodijeliti CID nekom kanalu neovisno o ostalim uređajima koji to mogu isto učiniti za svoje ostvarena kanale. Jedina iznimka u dodjeljivanju CID-a kanalima su neki od rezerviranih identifikatora (npr. signalizacijski kanal).

Za spojno orijentirane kanale definira se automat (state machine), sa definiranim stanjima, prijelazima, i akcijama kao odgovorima na prijelaze. Taj automat koristi se samo za dvosmjerne CID-ove i ne uključuje signalni kanal ni jednosmjerni kanal. Za spojno orijentirane kanale, kraj kanala (endpoint) identificiran preko CID-a, može biti u jednom od nekoliko mogućih stanja. Početno stanje kanala (prije stvaranja konekcije) je CLOSED (zatvoren) koje indicira da nijednom kanalu nije pridružen CID. To je stanje u koje će kraj kanala prijeći ako veza nije prisutna (ujedno i jedino stanje koje ne zahtjeva komunikaciju sa baseband slojem). Prijenos podataka će biti omogućen jedino ako je endpoint u stanju OPEN.

Konekcija - da bi se mogao otvoriti kanal, potrebno je napraviti određene predradnje. To su konekcija i konfiguracija. Konekcija može nastati u dva slučaja:

- ako lokalni uređaj (L2CAP sloj gledamo sada kao entitet) zatraži konekciju na udaljeni uređaj - zahtjev je potekao iz gornjih slojeva BT protokola. U ovom slučaju lokalni uređaj prelazi u stanje W4_L2CAP_Connect_RSP u kojem čeka odgovor od uređaja kojeg je prozvaao
- ako je lokalni uređaj dobio zahtjev (request) za konekcijom od nekog drugog uređaja - ovaj zahtjev se prosljeđuje gornjim slojevima. U ovom slučaju lokalni uređaj prelazi u stanje W4_L2CA_Connect_RSP u kojem čeka odgovor.

U oba slučaja, nakon primljenog odgovora, lokalni uređaj odlazi u CONFIG stanje.

Konfiguracija - za konekcijski orijentirane kanale sljedeći korak u uspostavljanju veze je konfiguracija kanala koja uključuje razmjenu informacija između obje strane sve dok se svi parametri ne usuglase (MTU, QOS, *flush-timeout*). Način na koji je to omogućeno je preko *Configuration_Request* i *Configuration_Response* naredbi. Nakon završene konfiguracije, oba L2CAP entiteta prelaze u stanje OPEN te prijenos podataka može početi.

Prekid veze - prekid veze se vrši na način da jedna strana pošalje *disconnect* zahtjev drugoj strani. Razlikujemo dva slučaja:

- ako lokalni uređaj (L2CAP sloj gledamo sada kao entitet) zatraži raskid veze - zahtjev je potekao iz gornjih slojeva BT protokola. U ovom slučaju lokalni uređaj prelazi u stanje *W4_L2CAP_Disconnect_RSP* u kojem čeka odgovor od uređaja kojeg je prozvaao
- ako je lokalni uređaj dobio zahtjev *disconnect request* od nekog drugog uređaja - ovaj zahtjev se proslijeđuje gornjim slojevima. U ovom slučaju lokalni uređaj prelazi u stanje *W4_L2CA_Disconnect_RSP* u kojem čeka odgovor.

U oba slučaja, kada L2CAP entitet primi odgovor, lokalni uređaj odlazi u CLOSE stanje.

2.13. RFCOMM

RFCOMM se zasniva na baseband sloju i zadužen je za sigurnu dostavu paketa podataka. Može emulirati serijsku vezu statusa i postavki za RS-232 serijski port. Nema mogućnost ispravaka pogrešaka. Ima mogućnosti uokviravanja, multipleksiranja, a na njegovo djelovanje utječu i :

- status modema – RTS/CTS, DSR/DTR, DCD, *ring*
- status udaljenih linija – break, overrun, parity
- *remote port settings* – broj podatkovnih bitova, *baud rate*
- veličina okvira
- kontrola toka

RFCOMM podržava dva tipa uređaja:

- *type1* - unutarne emulirani serijski port
- *type2* - uređaji sa fizičkim serijskim portom.

RFCOMM komunicira pomoću okvira. RFCOMM okviri postaju *data payload* polje u L2CAP paketima. Postoji pet različitih tipova okvira:

- SABM – *Start asynchronous balanced mode* – start instrukcija
- UA – Unnumbered acknowledgment – odgovor nakon što je uspostavljena veza
- DISC – Disconnect command
- DM – disconnected mode – odgovor na zahtjev za prekidom veze
- UIH – Unnumbered information with header check.

Zbog toga što se RFCOMM okviri prenose u payload polju L2CAP paketa, L2CAP veza mora biti uspostavljena prije RFCOMM veze. RFCOMM ima rezerviranu PSM (Protocol and service multiplexer) vrijednost koju koristi L2CAP za prepoznavanje RFCOMM prometa. Prvi okvir koji se prenosi RFCOMM kanalom je SABM. Ukoliko uređaj s kojim se želi uspostaviti veza odgovara potvrdno (sa UA), tada prelazi u ASM (Asynchronous balanced mode). U drugom slučaju šalje paket DM. RFCOMM ima 60 s vremenski istjek koji starta kada je naredba poslana. Ako u tom vremenu ne stigne odgovor, veza se prekida.

RFCOMM okvir počinje sa adresnim poljem koje identificira kojem od multipleksiranih kanala poruka pripada. U tom polju se nalazi i zastavica koja signalizira da li se radi o naredbi ili o odgovoru. Nakon njega slijedi kontrolno polje koje služi za identifikaciju tipa okvira.

Potom dolazi duljina polja i sami podaci. Na kraju se nalazi FCS (Frame check sequence) koji se računa posebnim algoritmom u koji ovdje neću ulaziti.

2.14. SDP

SDP (Service Discovery Protocol) je protokol koji omogućuje BT uređajima da dinamički reagiraju na ponuđene usluge u BT okruženju. To konkretno znači da uređaji mogu otkriti koje usluge su njima dostupne i odrediti njihove karakteristike. SDP u BT okruženju drugačiji je nego u klasičnim mrežnim okruženjima jer se skup usluga koje su dostupne mijenja dinamički (ovisno o RF blizini uređaja u pokretu). U takvom jednom okruženju BT uređaj može imati ulogu SDP klijenta koji zahtjeva uslugu, SDP servera koji daje uslugu ili može imati obje uloge istovremeno (s tim da jedan BT uređaj ne može imati ulogu dva ili više SDP servera dok može biti klijent za više usluga).

Bitna je i prilagodba konstantnoj promjeni broja povezanih uređaja. Usluge mreže mogu biti uobičajene, kao što su ispisivanje na mrežnim printerima, slanje faksimila, kao i razne vrste pristupa informacijama, npr. telekonferencije, mrežne pristupne točke i e-commerce primjene. Protokol za otkrivanje usluge treba udovoljavati slijedećim zahtjevima:

- Treba omogućiti klijentima pronalaženje potrebnih usluga prema atributima tih usluga
- Dozvoliti uslugama da budu otkrivene na temelju klase usluge
- Omogućiti potragu za uslugama bez prethodnog znanja o karakteristikama tih usluga
- Omogućiti pronalaženje usluga kada uređaj koji daje uslugu uđe u BT kruženje, kao i pronalaženje usluga koje postaju raspoložive za vrijeme boravka uređaja u BT okruženju
- Otkriti kada je usluga prestala biti dostupna, bez obzira da li je davatelj usluge napustio BT okruženje ili je davatelj usluge prekinuo uslugu i ostao u BT okruženju
- Omogućiti jedinstveno označavanje usluge, klase usluge i atributa
- Dopustiti klijentu na jednom uređaju da pronade uslugu koja se nalazi na drugom uređaju, bez da se kontaktira treći uređaj
- Mora biti pogodan za implementaciju na uređaju limitirane kompleksnosti
- Omogućiti postupno pronalaženje informacija o usluzi, kako bi se što više smanjila količina podataka koja se mora prenijeti prije nego se ustanovi da promatrana usluga nije potrebna

SDP je paketni protokol koji koristi model zahtjev - odgovor (*request - response*) gdje se svaka transakcija sastoji od jednog podatkovnog zahtjeva za protokolom (PDU - Protocol Data Unit) te jednog odgovora (također PDU). Svaki PDU se sastoji od zaglavlja nakon kojeg slijede parametri. U specifičnim slučajevima kada SDP koristi L2CAP prijenosni protokol, višestruki PDU-i mogu biti poslani u jednom L2CAP paketu, ali je moguće poslati samo jedan paket po vezi u danom trenutku (prema SDP poslužitelju).

PDU je jedinica sastavljena od paketa koje pripremaju viši slojevi BT arhitekture. Svaki PDU sastoji se od zaglavlja iza kojeg slijede specifični PDU parametri. Dužina parametra je specificirana u zaglavlju paketa. Neki SDP zahtjevi traže odgovore duže od jednostrukog *response* PDU-a, pa SDP poslužitelj generira parcijalni odgovor koji sadrži *parametar stanja*

neprestanog trajanja. Parametar je polje varijabilne duljine i sadrži broj okteta koji slijede u polju. Parametar nije standardiziran i od značenja je jedino poslužitelju koji ga je stvorio. Kada SDP klijent primi parcijalni odgovor i prateće parametre neprestanog trajanja on može ponovno izdati originalni zahtjev sa uključenim stanjem neprestanog trajanja obavještavajući poslužitelj da želi ostatak odgovora.

Protokol za otkrivanje usluga omogućava aplikaciji na klijentu da pronade usluge koje nudi serverska aplikacija kao i atribute tih usluga. Atributi usluga uključuju tip ili klasu usluge koja se nudi, te mehanizam kojim se koristi ta usluga. Ako server odredi da je zahtjev pogrešno formatiran, ili iz nekog razloga server ne može odgovoriti prikladnim PDU-om, odgovoriti će sa *SDP_ErrorResponse* PDU-om.

Service (usluga) je entitet koji može pružiti informaciju, izvesti akciju, ili upravljati resursom u korist drugog entiteta. Usluga može biti implementirana kao hardver, softver ili kao kombinacija obojega. Sve informacije o usluzi koju pruža neki uređaj su kataloški pohranjene na SDP serveru u vidu zapisa koji se nazivaju *service records* (zapisu o usluzi). Taj zapis se u cijelosti sastoji od liste atributa usluge. Svaki atribut usluge opisuje i definira jednu karakteristiku usluge. Atribut se sastoji od 16-bitnog ID-a te polja vrijednosti varijabilne dužine (*attribute value*). Ovo polje vrijednosti nije ograničeno na određeni tip podatka. Sastoji se od zaglavlja i podatkovnog polja. U zaglavlju je naveden tip podatkovnog polja, koje može biti *null*, *unsigned integer*, *signed twos-complement integer*, *Universally Unique Identifier (UUID)*, *string*, *Boolean*, *data element sequence (set)*, *data element alternative* i URL tipa. Interpretacija polja vrijednosti ovisi o *attribute ID* polju te o klasi servisa kojoj određeni servis pripada.

Transakcija traženja usluge dozvoljava da klijent dobavi pokazivač na zapis o usluzi, na temelju vrijednosti atributa u tom zapisu, s tim da ta vrijednost mora biti UUID. Uzorak traženja usluge predstavljen je listom UUID-a koji se koriste za lociranje odgovarajućeg zapisa o usluzi. Uzorak traženja usluge odgovara zapisu usluge ako UUID-i koje on sadrži čine podskup UUID-ova u vrijednostima atributa u zapisu usluge. Ukoliko samo jednog UUID-a nema u zapisu uzorak ne odgovara. *Searching* mehanizam je koristan kada klijent želi naći određene specifične usluge, no kada klijent želi vidjeti koje su mu *sve* usluge dostupne, tada se koristi druga metoda pretraživanja - *browsing*.

Pregledavanje usluga odgovara izlistavanju svih usluga koje su opisane u *zapisima o usluzi* SDP poslužitelja. SDP mehanizam pregleda temelji se na atributu koji dijele sve klase usluge, *BrowseGroupList* i sadrži listu UUID-a od kojih svaki predstavlja grupu s kojom usluga može biti povezana u svrhu pregleda. Kada klijent želi ispis usluga koje SDP poslužitelj nudi, kreira uzorak traženja koji sadrži UUID koji predstavlja root preglednu grupu. Tada se prikažu sve usluge s najvišeg nivoa tj. one koje imaju root ID u atributu *BrowseGroupList*. Ako neki SDP poslužitelj ima malo usluga sve će biti stavljene u root preglednu grupu. U slučaju većeg broja usluga pregledne grupe biti će organizirane hijerarhijski, što se opisuje zapisom o usluzi u klasi *BrowseGroupDescriptor*. Na taj način se omogućuje pregled korak-po-korak što je vrlo korisno kad server sadrži puno zapisa o usluzi.

2.15. Aplikacije – Bluetooth profili

Primjena BT-a je predviđena preko različitih korisničkih modela. Oni obuhvaćaju korisničke scenarije u kojima Bluetooth vrši radio prijenos. Da bi se opisalo kako implementirati ove korisničke modele, definira se određen broj profila. Profili specificiraju kako se aplikacije i uređaji mapiraju na BT koncept. Nadalje, definiraju koje poruke i procedure iz BT specifikacije koristiti za određeni model primjene, a daju i nedvosmislen opis zračnog sučelja (*air interfacea*) za pojedine usluge i slučajeve. Rad jednog profila ovisi o drugom (nadređenom) profilu ako je njegova pozicija unutar područja drugog profila. Ovisnost među profilima može biti direktna ili indirektna.

Definiraju su četiri glavna profila:

- Profil osnovnog pristupa - *Generic Access profile* (GAP)
- Profil serijskog porta - *Serial Port Profile*
- Profil aplikacije za otkrivanje usluge - *Service Discovery Application Profile* (SDAP)
- Profil osnovne zamjene objekata - *Generic Object Exchange Profile* (GOEP).

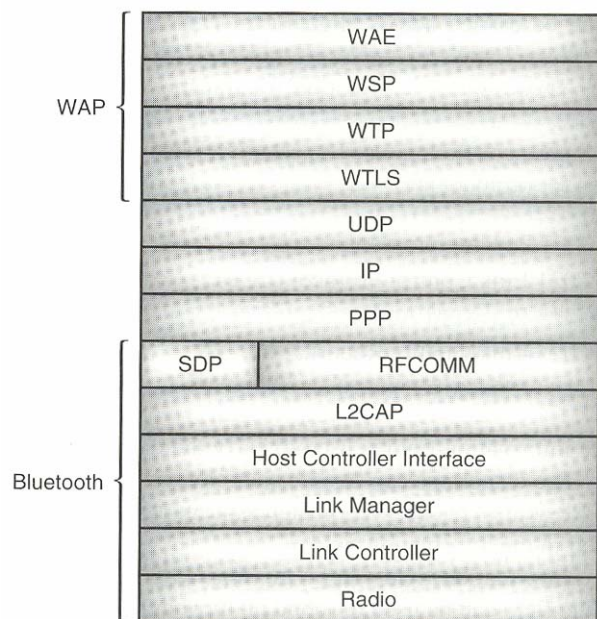
GAP definira kako se dvije nepovezane BT jedinice pronalaze i uspostavljaju vezu. Ovaj profil definira osnovne operacije kojima se služe svi ostali profili koji koriste GAP. GAP se ujedno i brine da bilo koje dvije BT jedinice, bez obzira na proizvođača, izmjene informacije kako bi se otkrilo koju vrstu aplikacije jedinica podržava. BT jedinice koje se ne oslanjaju ni na jedan drugi profil moraju se osloniti na GAP kako bi se osigurala osnovna interoperabilnost i međusobna interakcija.

SDAP definira otkrivanje usluga dostupnih BT jedinici. Profil traži znane i specifične usluge kao i općenite (general) usluge. SDAP sadrži aplikaciju, *Service Discovery User Application*, koja je potrebna u BT jedinici za pronalaženje usluga. Ova aplikacija komunicira sa protokolom za otkrivanje usluga (Service Discovery Protokolom) koji šalje i prima upite prema i od ostalih BT jedinica. SDAP, dakle, opisuje aplikaciju koja komunicira sa specifičnim BT protokolom kako bi ga u potpunosti iskoristila, a sve radi pružanja bolje usluge krajnjem korisniku. SDAP ovisi o GAP-u, tj. koristi dijelove GAP-a.

Serial Port Profile definira način kako ostvariti virtualni serijski port na dvije jedinice (uređaja) i povezati ih preko Bluetooth-a. Koristeći ovaj profil stvaramo BT jedinicu sa emulacijom serijskog kabla koji koristi RS232 signalizaciju. Profil osigurava prijenos podataka do 128 kbit/s. I ovaj profil ovisi o GAP-u.

GOEP definira skup protokola i procedura koje koriste aplikacije koje se bave slanjem, primanjem i zamjenom objekata. Veliki broj korisničkih modela temelji se na ovom profilu (npr. *File Transfer and Synchronisation*). Tipične BT jedinice koje koriste ovaj profil su prijenosna računala, PDA-ovi, mobilni telefoni). Aplikacije koje koriste GOEP pretpostavljaju da su veza i kanali uspostavljeni, kako je definirano u GAP-u. GOEP opisuje procedure za guranje podataka kao i kako povući podatke sa jedne BT jedinice prema drugoj. GOEP ovisi o Serial Port Profilu.

2.16. Usporedba s WAP-om i IrDA



Slika 22. WAP i Bluetooth protokolni stog

Wireless application protocol (WAP) je bežični protokol koji dozvoljava mobilnim uređajima korištenje podatkovnih usluga i pristup Internetu. WAP može raditi na temelju različitih bežičnih tehnologija, od kojih svaka mora sadržavati dno WAP protokolnog stoga kao nosač. U tom pogledu, BT predstavlja još jedan potencijalni nosač za WAP. WAP podržava arhitekturu server – klijent. Klijent komunicira sa serverom (ili proxy-em) pomoću WAP-a. WAP – omogućeni klijenti mogu koristiti mikro - preglednike, koji su specijalni dizajnirani Web preglednici (browseri) namijenjeni ugradnji u male mobilne uređaje. Mikro – preglednik je dizajniran da radi sa malim ekranom i da koristi manje memorije nego preglednik pokrenut na PC-u. WAP također podržava WML (Wireless markup language), derivaciju HTML-a (odnosno SGML-a). Najnovije specifikacije podržavaju i XML, kao i njegovu derivaciju GML (Geographic markup language), što otvara neslućene mogućnosti primjene. Komponente WAP protokolnog stoga su:

- WAE (Wireless application environment) – pruža korisničko sučelje koje je manja verzija Web browsera
- WSP (Wireless session protocol) – ostvaruje komunikaciju između WAP klijenta i WAP servera
- WTP (Wireless transport protocol) – pruža pouzdan transportni sloj za WSP. Ukoliko drugi slojevi ispod WSP-a pružaju također pouzdanu uslugu, nema potrebe za WTP-om. Kod BT-a, upravo baseband sloj pruža pouzdan prijenos, pa u slučaju da se kao nosilac za WAP koristi BT, WTP može biti izostavljen
- WTLS – jamči sigurnost.

WAP slojevi su udruženi sa 3 Internet protokol sloja koji dopuštaju slanje datagrama kroz BT RFCOMM serijski ulaz:

- UDP (User datagram protocol) – nepouzdan bespojni prijenos datagrama
- IP (Internet protocol) služi za adresiranje, usmjeravanje i segmentaciju paketa

- PPP (Point to point protocol) – klijent / server orijentiran prijenos paketa koji služi za uspostavu veze i enkapsulaciju paketa mrežnog sloja po serijskim vezama

IrDA (Infrared data organisation) i BT oboje podržavaju ad – hoc veze kratkog dometa. IrDA podržava brži prijenos podataka, ali ima i više slojeve u protokolnom slogu nego BT. U tom pogledu su IrDA i OBEX (Object exchange protocol) izvršno prilagođeni za prijenos podataka između BT uređaja. IrDA predstavlja komunikacijski sistem koji se temelji na infracrvenom svjetlu. Da bi se veza uspostavila, potrebna je izravna vidljivost između uređaja (IR svjetlo ne prodire kroz namještaj ili zidove kao što može RF uređaj – Bluetooth). Također, Bluetooth ima puno potpuniju mrežnu arhitekturu dok IrDA je više namijenjen kontroliranoj privatnoj razmjeni podataka između samo dva uređaja. U tablici 3. je dana usporedba.

Parametar	IrDA	Bluetooth
Broj uređaja u mreži	2	može biti vrlo velik
Medij	optički / direktan	RF
Brzina prijenosa podataka (Mb/s)	1.152, 4, 16	1
Domet	20 cm / 1.2 m	10 m / 100 m
Podaci i glas?	samo podaci	podaci i glas

Tablica 3. Usporedba IrDA i Bluetooth-a

2.17. Usporedba s IEEE 802.11 (bežični ethernet) i 3GPP-om

IEEE 802.11 radna grupa je zadužena za definiranje i održavanje WLAN (Wireless local area networks) specifikacija. Originalno, opisuje tri fizička sloja i prihvaćen MAC (Medium access control). Postoje i novije specifikacije IEEE 802.11a i IEEE 802.11b. MAC pruža dvije konfiguracije mreže: nezavisnu (direktna komunikacija među stanicama) i infrastrukturnu (komunikacija preko pristupnih točaka). MAC pruža i osnovne mehanizme sinkronizacije kanala i izbjegavanja kolizije pomoću CSMA/CA (Carrier sense multiple access), ali i prepoznavanje usluge (slično kao kod BT inquiry i scan), fragmentaciju podataka, enkripciju i dr. Od tri fizička sloja koja definira specifikacija, jedan je sličan BT-u:

- Frequency hop spread spectrum (FHSS)
2.4 Ghz ISM band, 1 – 2 Mb/s
2 i 4 – razinska GFSK modulacija
hopping 50 hops/s na 79 kanala

FHSS ima puno sporiju izmjenu frekvencija od BT-a, a to je donijelo lakšu implementaciju RF tehnologije. Međutim, korištenjem 4 – razinske GFSK modulacije, brzina prijenosa se može lako povećati. U tom pogledu je napredak donio i IEEE 802.11b koji podržava 5.5 i 11 Mb/s uporabom CCK (Complementary code keying) kao modulacijske sheme. Familija standarda IEEE 802.11 je vrlo raširena u svijetu. Trenutno se radi na povećanju brzine prijenosa, a ograničenje predstavlja frekvencija nosioca koja je ista kao i kod BT-a (ISM 2.4 Ghz) i koja polako postaje prenapučena. To BT rješava brzim frekvencijskim skokovima, a na 802.11 se još radi na tome. Problem je jedino u cijeni: 802.11 bazirani produkti koštaju od 100 \$ pa na dalje (skuplji imaju visoke brzine prijenosa – HRDSSS – High rate direct sequence spread spectrum), što je izrazito više od BT-a. Zbog toga što su BT i 802.11 slične,

ali ne i identične specifikacije, postoji potražnja za opremom temeljenom na oba standarda. U novije vrijeme postoje i proizvodi koji rade istodobno na BT piconetu i 802.11 mreži.

2.18. 3GPP – što je to?



3GPP (3rd Generation Partnership Project) je sporazum nastao krajem 1998. koji je standardizirao brojne telekomunikacijske usluge, poznate kao organizacijski partneri (engl. Organizational Partners). Originalna zadaća mu je bila proizvodnja globalnih aplikacija za tehničke specifikacije 3. generacije mobilnih sustava baziranih na GSM standardu. S vremenom su došle prelazne faze GPRS i EDGE, prema UMTS-u. Postoji organizacija u Francuskoj koja se brine o održavanju tog standarda (Mobile Competence Centre – MCC). Osnovna razlika između BT-a i 3GPP-a je u tome što je 3GPP orijentiran prvenstveno prema mobilnim telefonima, dok BT nalazi primjenu u svim mogućim mobilnim uređajima. 3GPP razvija specifikacije za IMT 2000. 3GPP je kooperacija od 6 partnera (ARIB, CWTS, T1, TTA I TTC) koji su prepoznati kao vodeća tijela standardizacije na području Japana, Kine, Europe, SAD-a i Koreje.

3. ZAKLJUČAK

3.1. Razvoj i primjena u budućnosti

Značenje slobode danas se promijenilo. To više ne označava stanje duha, nego našu fizičku pokretnost i mobilnost. Više nisu u modi gomile kablova za povezivanje. Danas se sve radi elegantnije, a razvoj tehnologija ide naprijed, tako da uskoro možemo očekivati pojavu PAN (Personal area network) i HAN (Home area network), baziranih na BT. Cijene se sa povećanjem primjene spuštaju, što dodatno doprinosi popularnosti. U budućnosti ćemo imati primjenu i BT auto, peglu? Zašto ne. Bitno je da nema žica.

Za razliku od bežičnog ethernet-a koji je prvenstveno statički usmjeren jer svaki uređaj ima svoju adresu, BT je puno pogodniji za dinamičke uvjete jer se adrese uređaja konstantno mijenjaju. Također, tu je i velika prednost u cijeni za BT pa se može očekivati da će vrlo skoro nadjačati bežični ethernet.

4. LITERATURA:

- [1] <http://www.bluetooth.com>
- [2] <http://www.ericsson.com/bluetooth>
- [3] <http://www.palowireless.com/bluetooth>
- [4] <http://www.3gpp.org/Default.htm>
- [5] J. Bray, C. Sturman: Bluetooth 1.1. Connect Without Cables, Prentice Hall, 2002.
- [6] V. Semenčić, F. Miloš: Seminarski rad Bluetooth, Zagreb, 2001.
- [7] H. Sertić: Seminarski rad Bluetooth, Zagreb, 2001.