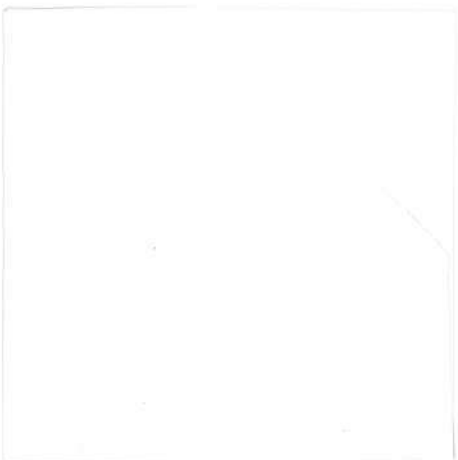


52057



UNIVERZA V LJUBLJANI
FAKULTETA ZA ELEKTROTEHNIKO

LAURA KLANČNIK, univ. dipl. ing. el.

**POVEZLJIVE NAPRAVE IN OMREŽNI SISTEMI
S PODROČJA HIŠNE AVTOMATIZACIJE**

MAGISTRSKO DELO

Mentor: prof. dr. Andrej Žemva

Ljubljana, 2005

ma 1131



O 52057/8.6.2005

Številka naloge: M-1032/2005
Datum: 5. 5. 2005

Fakulteta za elektrotehniko Univerze v Ljubljani izdaja naslednjo nalogo:

Kandidatka: **Laura KLANČNIK**, univ. dipl. inž. el.

Naslov: ***POVEZLJIVE NAPRAVE IN OMREŽNI SISTEMI S
PODROČJA HIŠNE AVTOMATIZACIJE***

Vrsta naloge: Magistrsko delo

Tematika naloge:

V zadnjih letih se je razvilo novo tržno področje, ki se imenuje hišna avtomatizacija. Zdaj to področje že vpliva na naslednjo generacijo naprav za široko porabo, bele tehnike in drugih naprav za hišo in dom. Osnovna dodana vrednost pri hišni avtomatizaciji je v integraciji proizvodov in storitev. Sisteme hišne avtomatizacije trenutno tržijo predvsem manjša podjetja, velika podjetja in institucije pa ga še raziskujejo in skušajo ugotoviti tržni potencial.

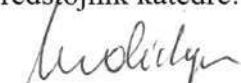
Analizirajte in predstavite možnosti razvoja inteligentnega hišnega sistema. Podajte bistvene standarde, namenjene za to področje, in opišite njihove najpomembnejše lastnosti. Zasnujte in izvedite prototip sistema, ki povezuje gospodinjski aparat prek omrežnega vozlišča do hišnega strežnika in nato prek internetnega omrežja do ponudnika storitev. Primerjajte zasnovo odprtega sistema z lastniško usmerjenim sistemom.

Mentor:


prof. dr. Andrej Žemva



Predstojnik katedre:


prof. dr. Igor Medič

Dekan:


prof. dr. Tomaž Slivnik

ZAHVALA

Zahvaljujem se mentorju prof. dr. Andreju Žemvi za pomoč in vodenje pri opravljanju magistrskega dela.

Zahvaljujem se tudi podjetju Gorenje, da mi je omogočilo študij, še posebej pa bi se zahvalila vodji razvoja, mag. Konradu Steblovniku za motivacijo pri študiju in vsakodnevnem delu, ter za nesebično razdajanje svojega znanja in izkušenj.

Posebna zahvala velja moji družini, ki mi je ves čas stala ob strani in me podpirala.

Povzetek

Magistrska naloga obravnava različne pristope k razvoju inteligentnih povezljivih naprav ter omrežij za potrebe hišne avtomatizacije.

Nekaj poglavij je posvečenih razlagi standardov, ki so bistven del razvoja omrežnih sistemov. Opisana sta dva komunikacijska standarda, ki izstopata na področju hišne avtomatizacije: LonWorks (de facto standard z 20 letno tradicijo), Konnex (novejši vseevropski standard, ki je še v procesu nastajanja).

Osnova za to nalogo je delo pri projektu 'Povezljivi aparati Gorenje', zato si posebno pozornost zasluži tudi CECED specifikacija, ki pokriva standardizacijo sporočil pri komunikaciji v omrežju gospodinjskih naprav.

Vedno bolj se poudarja pomembnost razvoja odprtih sistemov in omrežij. Uporaba OSGi standarda na strani hišnega strežnika omogoča ravno to - koeksistenco in interoperabilnost različnih standardov in komunikacijskih medijev v enem omrežju.

Na koncu naloge so vsa pridobljena znanja združena v različne možne izvedbe sistemov.

To magistrsko delo je ustvarjeno tudi z namenom njegove uporabe kot izhodiščne literature za pridobivanje osnovnih znanj s področja hišne avtomatizacije in lažje razumevanje dejanskih specifikacij omenjenih standardov, saj trenutno za to področje še ne obstaja literatura v slovenskem jeziku.

Ključne besede: Konnex, LonWorks, OSGi, CECED, Power line, inteligentni dom, povezljivi gospodinjski aparati, omrežje, komunikacijski protokol, hišni strežnik, storitve

1	UVOD	1
2	HIŠNA AVTOMATIZACIJA.....	3
2.1	APARATI ZA HIŠNO AVTOMATIZACIJO.....	3
2.2	IZHODIŠČA PROJEKTA INTELIGENTNI DOM GORENJE.....	5
2.3	CILJI PROJEKTA INTELIGENTNI DOM GORENJE	7
2.4	PROJEKT: POVEZLJIVI APARATI GORENJE	8
3	KONNEX	13
3.1	NASTANEK KONNEX ZDRUŽENJA IN OSNOVNE LASTNOSTI	13
3.1.1	<i>Osnovne lastnosti</i>	14
3.2	KNX SISTEM Z ORODJI.....	16
3.3	ELEMENTI KNX ARHITEKTURE	16
3.3.1	<i>Aplikacije, interoperabilnost in povezovanje</i>	18
3.3.2	<i>Osnovne nastavitvene sheme</i>	19
3.3.3	<i>Management omrežja in virov</i>	19
3.3.4	<i>Komunikacija :Fizični nivo</i>	20
3.3.5	<i>Komunikacija: Skupno jedro in sporočilni protokol</i>	21
3.3.6	<i>KNX viri</i>	22
3.3.7	<i>Modeli naprav</i>	22
3.3.8	<i>Identifikacija naprav</i>	23
3.4	SISTEM, KOMUNIKACIJA IN MODELI NASLAVLJANJA	23
3.4.1	<i>Logična topologija in individualni naslovni prostor</i>	24
3.4.2	<i>Omrežni management</i>	25
3.4.3	<i>Okvirji KNX sporočila</i>	26
3.5	APLIKACIJSKI MODELI, PODATKOVNE TOČKE IN POVEZAVE	27
3.5.1	<i>Skupinski objekti</i>	28
3.5.2	<i>Lastnosti vmesniških objektov kot PT</i>	29
3.5.3	<i>Prosto ali strukturirano povezovanje</i>	30
3.5.4	<i>Etiketirano povezovanje</i>	31
3.6	MODEL MEDSEBOJNEGA DELOVANJA.....	32
3.6.1	<i>Aplikacija: Oblika PT in funkcionalni bloki</i>	32
3.6.2	<i>Parametri PT</i>	33
3.7	NASTAVITVENI NAČINI	34
3.7.1	<i>Sistemske nastavitveni način(S-Mode)</i>	34
3.7.2	<i>CTRL nastavitven način</i>	36
3.7.3	<i>Način 'Pritisni gumb' (Push-button Mode)</i>	37
3.7.4	<i>LT (Logical Tag) nastavitveni način</i>	37
3.7.5	<i>LTE (Logical Tag Extended)- razširjen LT nastavitveni način</i>	38

3.7.6 A-način nastavitve	39
3.8 PROFILI	40
3.8.1 Opis profilov.....	40
4 LONWORKS	42
4.1 KAJ JE LONWORKS?	42
4.2 RAZLIKA MED LAN IN LON	44
4.3 KAJ JE LONWORKS CONTROL NETWORK?	44
4.4 LONWORKS KOMPONENTE.....	45
4.4.1 Neuron čip	46
4.4.2 PL3120 in PL3150 Power Line Smart Transceiver.....	50
4.4.3 Dokazana tehnologija in razvojni viri.....	52
4.5 LONTALK PROTOKOL	53
4.6 INTEROPERABILNOST – MEDSEBOJNO DELOVANJE	55
4.7 LONMARK ORGANIZACIJA	56
4.7.1 Standardizacija fizičnih kanalov in transceiverjev.....	57
4.7.2 Standardizacija aplikacijskih programov.....	58
4.7.3 Standardni tipi omrežnih spremenljivk (SNVT).....	58
4.7.4 Nastavitvene lastnosti.....	59
4.7.5 LONMARK objekti in funkcionalni profili.....	59
4.8 LNS PROGRAMSKO ORODJE	61
5 CECED	63
5.1 CECED SPECIFIKACIJA	65
5.1.1 Funkcionalna specifikacija (AIS Vol.1).....	67
5.1.1.1 Pristop k instalaciji gospodinjskih naprav	67
5.1.1.2 Specifikacija ključnih dogodkov in uporabniških funkcij.....	68
5.1.1.3 Osnovni opis aplikacijskega področja hišnih naprav	70
5.1.1.4 Razumevanje interoperabilnosti	72
5.1.1.5 Funkcionalni bloki.....	73
5.1.1.6 Pravila medsebojnega delovanja aplikacij.....	74
5.1.2 Struktura podatkov (AIS Vol.2)	75
5.1.3 Testna garnitura (AIS Vol.3)	76
5.1.4 Preslikave na komunikacijske protokole (AIS Vol.4)	79
6 OSGI	80
6.1 STORITVENA PLATFORMA OSGI	81
6.1.1 Vrste storitev in njihove koristi.....	82
6.1.2 Lastnosti storitvene platforme	83
6.1.3 OSGi - Interoperabilnost standardov.....	84
6.2 OSGI TEHNOLOGIJA	84

6.2.1 OSGi okvir (<i>Framework</i>).....	85
6.2.2 Storitve.....	87
7 UPORABA STANDARDOV V PRAKSI	90
7.1 SESTAVE RAZLIČNIH SISTEMOV	91
7.2 IZVEDBE OMREŽNIH VMESNIKOV POVEZLJIVIH NAPRAV	93
7.2.1 'Power Line' komunikacija.....	94
7.2.2 Konnex (EHS)omrežni vmesnik.....	95
7.2.3 LonWorks omrežni vmesnik.....	98
7.2.4 Upravljanje naprav in sistema.....	101
7.2.4.1 Hišni strežnik na osnovi OSGi ogrodja	101
7.2.4.2 Hišni strežnik na osnovi .Net Framework-a.....	103
7.2.5 Primer uporabe storitve vzdrževanja	105
7.3 PRIMERJAVA LASTNIŠKO USMERJENEGA SISTEMA Z ODPRTIM	106
7.4 IZDELAVA DEMO PRIMERA	106
8 SKLEP	109
9 LITERATURA	110

KAZALO SLIK

stran

SLIKA 2.1 : NAPRAVE IN PODSISTEMI, KI JIH NAJDEMO V SISTEMU INTELIGENTNE HIŠE	3
SLIKA 2.2 : GRADNIKI REŠITVE SISTEMA INTELIGENTNEGA DOMA.....	10
SLIKA 2.3 : SISTEM INTELIGENTNEGA DOMA S STORITVAMI	12
SLIKA 2.4 : KONCEPT RAZVOJA SISTEMA V GORENJU	12
SLIKA 3.1 : LOGOTIP KONNEX ZDRUŽENJA.....	13
SLIKA 3.2: KONFIGURACIJSKI NAČINI KNX	15
SLIKA 3.3: KNX MODEL	17
SLIKA 3.4: LOGIČNA TOPOLOGIJA KNX OMREŽJA.....	24
SLIKA 3.5: KNX STANDARDNA STRUKTURA SPOROČILA	26
SLIKA 4.1: OSNOVNA ZGRADBA LONWORKS VOZLIŠČA.....	46
SLIKA 4.2: CPU NEVRONSKEGA ČIPA	46
SLIKA 4.3: PROCESNI VMESNIK NEURON ČIPA	47
SLIKA 4.4: I/O ENOTE NEURON ČIPA	48
SLIKA 4.5: BLOK DIAGRAM TIPIČNE PL 31XX 'SMART TRANSCEIVER' NAPRAVE	51
SLIKA 4.7: ODPRT SISTEM	56
SLIKA 4.8: FUNKCIONALNI PROFIL	60

SLIKA 5.1: ODVISNOSTI OD SPECIFIKACIJE.....	66
SLIKA 5.2 : PRIMER APLIKACIJE OMREŽJA.....	70
SLIKA 5.3: DIAGRAM STANJ ZA PRALNI STROJ	72
SLIKA 5.4 STRUKTURA FUNKCIONALNEGA BLOKA ZA GOSPODINJSKE NAPRAVE.....	73
SLIKA 5.6: PRIMER OPISA TESTNEGA PRIMERA.....	78
SLIKA 5.7: FORMAT OKVIRJA ZA EHS	79
SLIKA 6.1: ARHITEKTURA HIŠNEGA STREŽNIKA Z UPORABO OSGi OKVIRJA.....	85
SLIKA 6.2: OSGi OGRODJE	86
SLIKA 6.3: OSGi SERVISI/STORITVE.....	87
SLIKA 7.1: SCENARIJ UPORABE SISTEMA S HIŠNIM TERMINALOM.....	92
SLIKA 7.2: SCENARIJ UPORABE SISTEMA Z INTELIGENTNIM HIŠNIM STREŽNIKOM	92
SLIKA 7.3: SCENARIJ UPORABE SISTEMA Z 'OSIROMAŠENIM' HIŠNIM STREŽNIKOM	93
SLIKA 7.4 : IZBIRA FREKVENČNEGA PASU	95
SLIKA 7.5 : BLOK SHEMA NA ZA UPORABO KONNEX(EHS) STANDARDA	96
SLIKA 7.6: PRIMER NA ZA KONNEX/EHS KOMUNIKACIJO	96
SLIKA 7.7 : BLOK SHEMA NA ZA UPORABO LONWORKS TEHNOLOGIJE	98
SLIKA 7.8: PRIMERJAVA STARE IN NOVE IZVEDBE NA.....	99
SLIKA 7.9: REFERENČNI DIZAJNI ZA UPORABO PL31XX.....	99
SLIKA 7.10: ZGRADBA OSGi HIŠNEGA STREŽNIKA	101
SLIKA 7.11: ZGRADBA STORITVENE PLATFORME OSGi OKVIRJA.....	102
SLIKA 7.12: PRIMER GUI ZA UPRAVLJANJE NAPRAV V OMREŽJU.....	103
SLIKA 7.13: ZGRADBA HIŠNEGA STREŽNIKA V WINDOWS OKOLJU	104
SLIKA 7.14: GUI - WINDOWS MEDIA CENTER.....	104
SLIKA 7.15: PRIMER STORITVE SERVISIRANJA NAPRAVE.....	105
SLIKA 7.16: PRIMER ZA DEMONSTRACIJO UPORABE WIN MEDIA CENTRA	107
SLIKA 7.17: PRIMER GUI ZA UPRAVLJANJE APARATOV GORENJE	108

KAZALO TABEL

stran

TABELA 4.1: TIPI PRENOSNIH KANALOV (MEDIJEV)	53
TABELA 5.1 : KLJUČNI DOGODKI PRI INSTALACIJI.....	69

1 UVOD

Danes se s področjem uvedbe inteligence v domače okolje ukvarja mnogo podjetij, raziskovalnih ustanov, fakultet,...itd. V oglasih in na spletu najdemo ogromno ponudb, izdelanih predstavitev, simulacij in podobnih projektov, vendar je resničnih delujočih primerkov še zelo malo. Gorenje se je na tem področju lotilo razvoja z vso resnostjo in v prvi fazi izvedlo kar nekaj projektov, ki so bili le demonstracijske narave, z namenom pridobivanja informacij od končnih uporabnikov. Raziskave so pokazale pomembnost razmišljanja o sistemu z vidika končnega uporabnika. V teh sistemih smo ponudili vpogled v dodatne funkcije aparatov, ki jih le ti pridobijo s povezavo na neko hišno omrežje.

Pod pojmi inteligentni dom (*Smart Home*) in okolje inteligentnega doma danes razumemo domače okolje, oplemeniteno in razširjeno s tehnologijo in storitvami. Gre za različna opravila, ki se lahko samostojno izvajajo z namenom povečanja udobnosti življenja (varnost in zavarovanje, komunikacija, udobje, varčevanje energije in varovanje okolja, ...).

Osnovne zahteve za izvedbo 'Inteligentnega doma' so omogočanje povezljivosti naprav (v omrežje) in prilagoditev uporabnikovim zahtevam.

V tej nalogi je nekaj poglavij namenjenih razlagi standardov, ki so bistven del razvoja takšnega sistema. Opisana sta dva komunikacijska standarda, ki izstopata na področju hišne avtomatizacije: LonWorks in Konnex.

LonWorks je de facto standard, ki ima že skoraj 20 letno tradicijo in svoje korenine v Ameriškem podjetju Echelon. Pokriva velik del komunikacij na tem področju. Najbolj razširjena uporaba tega standarda je v Italiji, Skandinaviji, Koreji in Ameriki, pokriva pa tudi velik del objektov v preostalih delih Evrope. Standard ima močno podporo v organizaciji LonMark, ki je predana razvoju standardov za omogočanje interoperabilnosti, certificiranje produktov glede na te standarde, ter promocijo koristi takšnih sistemov.

Konnex je novejši standard, ki je še vedno v procesu nastajanja. Cilj je razvoj vseevropskega komunikacijskega standarda za področje hišne avtomatizacije. Konnex združenje podpirajo pomembne Evropske organizacije in podjetja, kar da temu standardu tudi veliko težo.

Ker mi kot osnova za to nalogo služi moje delo pri projektu povezljivih aparatov Gorenje, si posebno pozornost zasluži tudi CECED organizacija, kateri je namenjeno 5. poglavje, ki pokriva standardizacijo sporočil v omrežju gospodinjskih naprav.

V 6. poglavju naloge se dotaknem tudi OSGi programskega okvirja, ki predstavlja pomemben gradnik v sistemu. Je del programske opreme hišnega strežnika, ki ima glavno vlogo pri omogočanju koeksistence in interoperabilnosti različnih standardov in komunikacijskih medijev v enem omrežju.

V zadnjem poglavju pa so opisane različne možne izvedbe sistemov z uporabo vseh prej navedenih standardov.

To magistrsko delo je ustvarjeno tudi z namenom uporabe za literaturo pri pridobivanju osnovnih znanj s področja hišne avtomatizacije in za lažje razumevanje dejanskih specifikacij omenjenih standardov, saj trenutno za to področje še ne obstaja literatura v slovenskem jeziku.

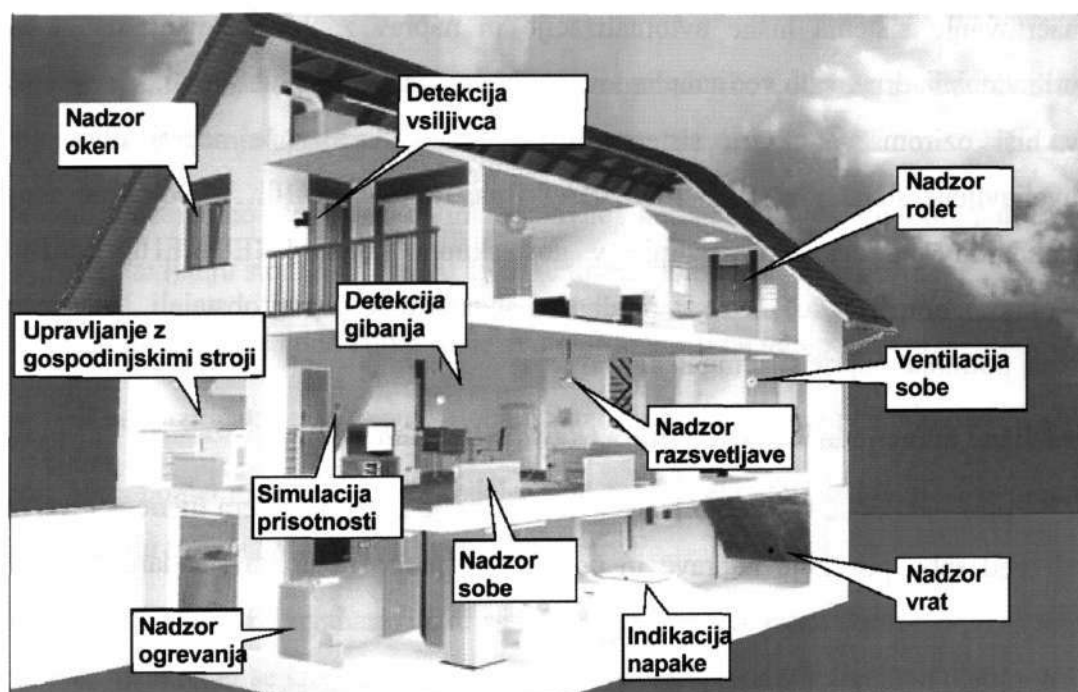
2 HIŠNA AVTOMATIZACIJA

V zadnjih letih se je razvilo novo tržno področje ali kar industrija, imenovana hišna avtomatizacija. Ta bo, oziroma že kreira naslednjo generacijo naprav s področja široke potrošnje, bele tehnike ter ostalih naprav za hišo in dom. Osnovna dodana vrednost pri hišni avtomatizaciji je v integraciji proizvodov in še pomembneje, v integraciji storitev.

Komunikacijsko omrežje zagotavlja infrastrukturo za povezovanje aparatov, senzorjev, krmilnikov in upravljalnih enot, oziroma hišnih strežnikov. Sisteme hišne avtomatizacije trenutno tržijo predvsem manjša podjetja, velika podjetja in institucije pa raziskujejo to novo vejo industrije in skušajo ugotoviti oziroma določiti potenciale tega trga.

2.1 Aparati za hišno avtomatizacijo

V okviru sistema hišne avtomatizacije se izraz aparat ne nanaša samo na gospodinjske aparate s področja bele tehnike, avdio/video aparate in prenosne naprave, ampak vključuje tudi komponente za ogrevanje in hlajenje, varnostni sistem, osvetlitev... [1] (slika 2.1).



Slika 2.1 : Naprave in podsistemi, ki jih najdemo v sistemu inteligentne hiše

Hišna avtomatizacija pokriva široko paleto proizvodov in storitev, ki so namenjeni za široko potrošnjo. Te naprave si delijo nekaj skupnih atributov:

- **poudarek je na podsistemih:**

Večina aparatov v hiši je samostojnih in v lastnem ohišju. Vsaka naprava deluje neodvisno od drugih in ima svoj nabor uporabniških ukazov. Naprave v okolju sistema hišne avtomatizacije pa so sposobne med seboj izmenjavati podatke. To omogoča, da lahko naprave združimo v podsistem. Primere takšnih podsistemov najdemo na sliki 1: sistem varovanja, avdio/video sistem, visoko razviti sistem razsvetljave s pred-nastavljenimi nivoji osvetlitve po skupinah luči. Nek bodoči podsistem bo na primer omogočil pralnemu ali pomivalnemu stroju, da ta zahteva od grelnika vode, da se v nekem trenutku izklopi in s tem zmanjša trenutno porabo električne energije.

- **koeksistenca komunikacijskih standardov:**

Nekaj omenjenih podsistemov že obstaja. Večina komponent je povezanih na osnovi posebnih tehnologij in ožičenj narejenih po naročilu. Standardizacija hišne avtomatizacije, ki je še vedno v razvoju (npr. Konnex), bo rešila proizvajalce potrebe po razvoju "ad hoc" komunikacijskih protokolov in ožičenj. Kljub temu moramo pri načrtovanju sistema hišne avtomatizacije in naprav zanj upoštevati, da se bo v prihodnosti združevalo več standardov, saj bodo posamezni podsistemi, ki se srečujejo v hiši oziroma v okviru sistema hišne avtomatizacije, še naprej ohranjali že uveljavljene standarde (EIB, EHS, BATIBus, LonTalk, X10...). Določene skupine standardov pa celo konvergirajo v nov skupni standard (EIB+EHS+BatiBus = Konnex), ampak tako, da bodo predhodni standardi še naprej obstajali. Načrtovati je torej potrebno odprte sisteme in naprave.

- **različne lokacije hišnega strežnika (*Residential Gateway*):**

Poleg naprav se v sistemu hišne avtomatizacije pojavlja pojem **hišnega strežnika**, ki med sabo povezuje naprave in podsisteme preko različnih standardnih mrežnih povezav, hkrati pa omogoča dostop do našega inteligentnega hišnega sistema od zunaj s pomočjo različnih fizičnih povezav.

Pri razvoju sistema moramo upoštevati dva osnovna kriterija:

- kje bo lokacija hišnega strežnika?
- kateri komunikacijski standard bomo implementirali?

Niti lokacija niti komunikacijski standard(i) niso vnaprej določeni, zato moramo načrtovati odprt sistem in prepustiti uporabniku izbiro in določitev glede teh dveh parametrov glede na želje in ideje, ki jih ima, oziroma glede na sistem inteligentne hiše, za katerega se je odločil (inženiring hišne avtomatizacije).

Ravno zaradi teh ugotovitev je pri razvoju aparatov potrebno upoštevati primerne standarde, ki omogočajo izgradnjo odprtega sistema, česar se v Gorenju zelo dobro zavedamo.

2.2 Izhodišča projekta Intelligentni dom Gorenje

Kot že omenjeno, nastaja v Evropi in v svetu novo tržno področje inteligentnega doma in inteligentnih aparatov. Tržni delež tega področja bo po nekaterih ocenah v roku petih let po obsegu presegel tržni delež, ki ga danes pokriva področje PC računalnikov. Pri tem se pojavlja cel kup vprašanj, na katere je potrebno poiskati odgovore [1]:

- kaj je potrebno upoštevati pri načrtovanju inteligentnega doma oziroma aparatov zanj?
- perspektive "povezanega" načina življenja uporabnikov, ki ga ponuja inteligentni dom, in pričakovanja uporabnikov?
- ali razumemo tehnološke in poslovne koristi sodelovanja in partnerstva pri zagotavljanju storitev v okviru inteligentnega doma?
- katere industrije bodo lahko našle svoje priložnosti na tem področju?
- ali lahko pričakujemo enotne globalne standarde na področju hišnih mrež?
- kako pomembno je prečno sodelovanje med različnimi industrijskimi področji pri razvoju in proizvodnji inteligentnih aparatov?

Seveda odgovori na ta vprašanja niso enostavni in enotni. Postavlja pa se še eno ključno vprašanje: »Kako naj se Gorenje kot proizvajalec velikih aparatov bele tehnike pripravi na to novo tržno področje, ki neizbežno prihaja na trge, kjer je danes prisotno?« Možen odgovor

Gorenja je: »Najti je potrebno razvojno pot, da bomo šli v korak s časom, da bomo pripravljeni in, da bomo znali pravočasno integrirati naše aparate v novo informacijsko in Internetno okolje, ki se vse bolj in bolj seli v okolje našega doma.«

Pravzaprav je to novo informacijsko okolje že prisotno, kar pomeni, da je potrebno razviti nove družine gospodinjskih aparatov na enotni platformi, ki bo omogočila njihovo integracijo v sodobno okolje. Naj navedemo nekatera izhodišča, ki jih moramo upoštevati pri načrtovanju nove generacije aparatov z elektronskim upravljanjem, inteligentnih aparatov in povezljivih aparatov (v njih lahko najdemo tudi nekaj odgovorov na zgornja vprašanja):

Uporabniki v osnovi ne želijo povezljivih aparatov, ampak želijo imeti boljše aparate, ki jim bodo predvsem:

- olajšali življenje v domačem okolju,
- ponudili več razvedrila,
- omogočali enostavno uporabo, za kar so uporabniki pripravljeni odšteti več denarja (raziskave sta opravila Electrolux in Whirlpool).

Skupaj s povezljivimi aparati moramo uporabnikom ponuditi še idejo, kje in na kakšen način lahko to lastnost aparata koristno uporabijo.

Ne smemo jim ponuditi več funkcij kot jih bodo potem pripravljeni oziroma jih bodo sploh znali uporabljati. Tipični primer je mobilni telefon - veliko uporabnikov kupuje aparate z več funkcijami in ravno zaradi teh, kljub temu da jih kasneje sploh ne uporabljajo! Razmišljati je potrebno o funkcijah in rešitvah, ne pa o tehnologiji, ki te omogoča.

Ljudje iščejo aparate, ki jim bodo prevzeli čim več in predvsem težja opravila v hiši s čimer pridobijo več prostega časa, ki je danes najbolj pogrešana 'dobrina'. Uporabniki v prvi vrsti iščejo aparate, ki so enostavni za uporabo, ne pa aparate, ki jih je mogoče povezati v mrežo.

Vsekakor bo v bližnji prihodnosti v hiši veliko aparatov in podsistemov, ki bodo povezljivi po različnih standardih. Uporabnik se bo odločil za nakup aparata glede na lastnosti, ki jih pričakuje in želi in ne glede na komunikacijski standard. Vsekakor bo hišni sistem z različnimi standardi potreboval posebni hišni strežnik, ki bo omogočal medsebojno povezavo aparatov in hkrati omogočal dostop do hišnega sistema od zunaj (preko Interneta) in seveda koriščenje cele vrste storitev (npr. *'pay-per-use'* = plačaj kolikor uporabljaš).

Pri pripravi aparatov za novo tržno področje so zelo pomembna partnerstva z različnimi podjetji, s katerimi je mogoče kreirati sisteme primerne za okolje.

Predlagana izhodišča za iskanje partnerstva so naslednja:

- Iskanje naravnih partnerjev za sodelovanje na tem področju (arhitekti, stavbeniki, lastniki stanovanj in njihovi agentje, ponudniki uslug, proizvajalci opreme, ...).
- Analiza faktorjev, ki so bistveni za snovanje partnerstva.
- Prepoznavanje tehnološke in finančne prednosti sodelovanja v poslovnem svetu inteligentnega doma.
- Raziskava sodelovanja in infrastrukture, potrebne za storitve v okolju inteligentnega doma.

Gorenje se je na koncu odločilo za dva pomembnejša partnerja pri projektu Povezljivih aparatov Gorenje, ki ustrezata najpomembnejšim zahtevam in nudita primerno tehnologijo in znanje potrebno za razvoj. To sta podjetji IBM s svojimi poslovnimi partnerji in Echelon (LonWorks).

2.3 Cilji projekta inteligentni dom Gorenje

Osnovni cilj projekta Inteligentni dom Gorenje je **implementacija višjega nivoja upravljanja v gospodinjske aparate**, ki jih bo mogoče med sabo tudi povezati v hišni sistem in sicer na osnovi lastnega znanja. To pomeni, da Gorenje poleg lastnega razvoja gospodinjskih strojev po posameznih programih (HZA – Hladilno-zamrzovalni aparati, PPA – Pralno-pomivalni aparati in KA – Kuhalni aparati), vključuje tudi razvoj mikrokrmilniških elektronskih sklopov. Pri tem gre za razvoj in konstrukcijo aparature in programske opreme. Z implementacijo višjega (inteligentnega) nivoja upravljanja v gospodinjske aparate želi podjetje doseči predvsem naslednje tržne cilje [1]:

- Odpreti novo perspektivno poslovno področje.
- Doseči večjo dodano vrednost proizvodov Gorenja.

- Povečati navezanost kupca na Gorenje kot ponudnika (elektronski servis, digitalni aparati, tele-diagnostika ...).
- Ponuditi oziroma omogočiti kupcem infrastrukturo za dodatne elektronske storitve (nabava hrane, bančne storitve, pomožne storitve ...).
- Razviti nove vrste tržnih proizvodov.

Z lastnim znanjem in celovitim obvladovanjem novih tehnologij bo dosežena veliko večja fleksibilnost v vseh fazah življenjske dobe proizvoda:

- v **fazi načrtovanja** - hitrejši odziv na zahteve trga;
- v **fazi implementacije** - kar najhitreje osvajanje proizvodnje novih sklopov in s tem aparatov z novimi funkcijami;
- v **fazi eksploatacije** – lažje in hitrejšo prilagajanje že uveljavljenih aparatov posebnim zahtevam kupcev ;

Fleksibilnost pri razvoju in osvajanju novih izdelkov postaja v današnjem času vse pomembnejša, saj je eksploatacijska doba proizvoda tudi na področju bele tehnike vse krajša. S čedalje večjim vstopanjem novih tehnologij: elektronike na osnovi mikrokontrolerov z vgrajenimi funkcijami, mrež za povezovanje v hiši in globalizacije preko medmrežja, upravljalni del aparatov hitro zastara in s tem postane tudi cenovno neprimeren.

Projekt razvoja inteligentnih gospodinjskih aparatov in inteligentnega doma s hišnimi mrežami zajema poleg našega oddelka Point-Intelligentni dom, ki upravlja in vodi projekt, še vse razvojne službe po posameznih programih (služba oblikovanja, skupne razvojne službe, nabavna služba, marketinška služba ...).

2.4 Projekt: Povezljivi aparati Gorenje

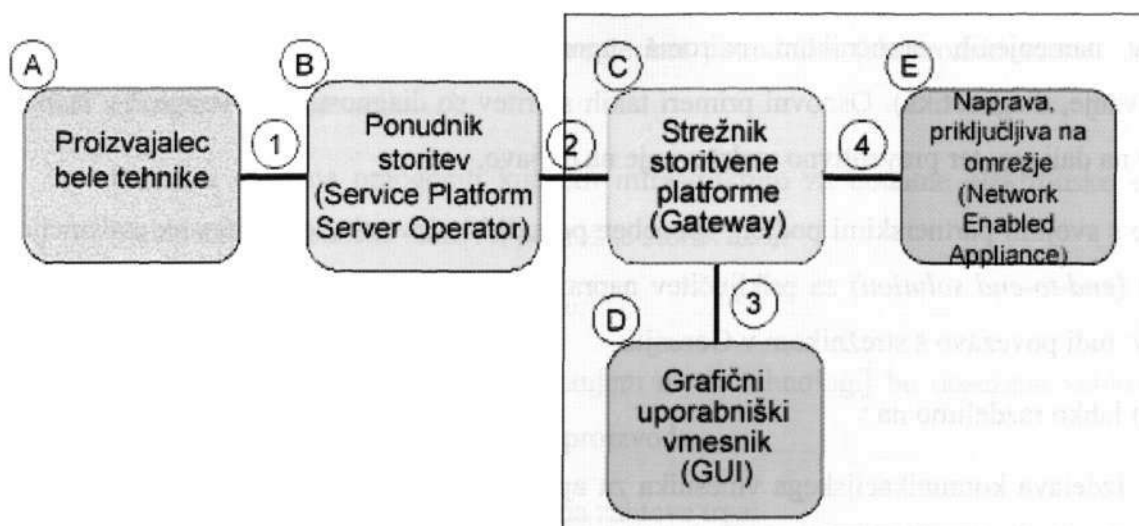
Gorenje se je odločilo, da bo pričelo z razvojem inteligentne bele tehnike, priključitve na internetno in domače omrežje, in s takšnimi produkti izboljšalo svojo ponudbo na trgu in dvignilo konkurenčnost svojih produktov. S tem bo Gorenju omogočen kakovostni dvig

storitev namenjenih naročnikom oziroma uporabnikom naprav bele tehnike (servis, vzdrževanje, diagnostika). Osnovni primeri takih storitev so diagnostika in vpogled v stanje naprav na daljavo, ter preventivno vzdrževanje na daljavo.

IBM je s svojimi partnerskimi podjetji sposoben ponuditi in izvesti vse storitve za realizacijo rešitve (*end-to-end solution*) za priključitev naprav bele tehnike na omrežje - tako domače, kot npr. tudi povezavo s strežnikom v Gorenju.

Projekt lahko razdelimo na :

1. Izdelava komunikacijskega vmesnika za aparate, ki bo podpiral dva komunikacijska standarda LonWorks in Konnex. Za komunikacijski medij pa bo uporabljeno obstoječe napajalno električno omrežje (*Power line*). S tem omogočimo povezavo aparata na omrežje (slika 2.2/E)
2. Izdelava CECED objektov za posamezen aparat (razloženo v 5. poglavju)
3. Izdelava potrebnih OSGi gonilnikov in aplikacij na strani hišnega strežnika za podporo aparatov Gorenje (slika 2.2/C)
4. Izdelava grafičnega uporabniškega vmesnika (GUI) (slika 2.2/D), za upravljanje aparatov in sistema (v hiši ali na daljavo npr. preko interneta, GSM,...)



Slika 2.2 : Gradniki rešitve sistema inteligentnega doma

Sistem inteligentne hiše mora biti:

- **transparenten** – naprave uporabljajo skupno komunikacijsko vodilo, tako, da se ne motijo med sabo.
- **razširljiv** – dodajamo lahko nove naprave, aplikacije in medije.
- **prilagodljiv** – naprave lahko namestimo kamorkoli v mrežo.
- mogoča je **avtokonfiguracija** – sistem se samodejno prilagaja spremembam v mreži, naprave se lahko vključijo brez intervencije strokovnjaka.
- **združljiv** – v sistem lahko vključimo naprave različnih proizvajalcev in standardov.

Naprave sistema so inteligentne v smislu:

- **lokalne intelligence**, ki omogoča, da naprava opravlja in upravlja proces, kateremu je namenjena, prilagodi se lahko stanju vhodnih veličin in stanju okolice.
- **mrežne intelligence**, ki omogoča, da se naprave povežejo v sistem hišnega omrežja.

Jedro sistema Inteligentni dom Gorenje je komunikacijski hišni vmesnik ali hišni strežnik. Ta predstavlja jedro hišnega sistema, vendar pa ni nujno, da opravlja le krmilno funkcijo sistema (to je lahko samo ena izmed njegovih nalog). Sistem je lahko distribuirano upravljan in tako strežnik opravlja predvsem nadzor stanja hišnih naprav in podsistemov, ki komunicirajo med seboj po omrežju. S pomočjo strežnika lahko naprave in podsistemi komunicirajo tudi navzven ali pa neposredno z uporabnikom s pomočjo za to namenjenih naprav (npr. GSM).

Možno je vgraditi eno ali več hišnih komunikacijskih večtočkovnih povezav, ki uporabljajo različne protokole (EHS/Konnex, TCP/IP, Lonworks). Tako je mogoče v sistem povezati podsisteme, ki temeljijo na različnih tehnologijah. Uporabljene so tudi različne fizične povezave (električno napajalno omrežje, RF, zvita parica, W-LAN,...). Da vse naštetu lahko deluje oz. sodeluje na eni sami napravi – torej hišnem strežniku – je bil razvit tako imenovan '*OSGi framework*', katerega lastnosti si bomo natančneje ogledali v 6. poglavju.

Osnovne funkcije komunikacijskega hišnega strežnika so:

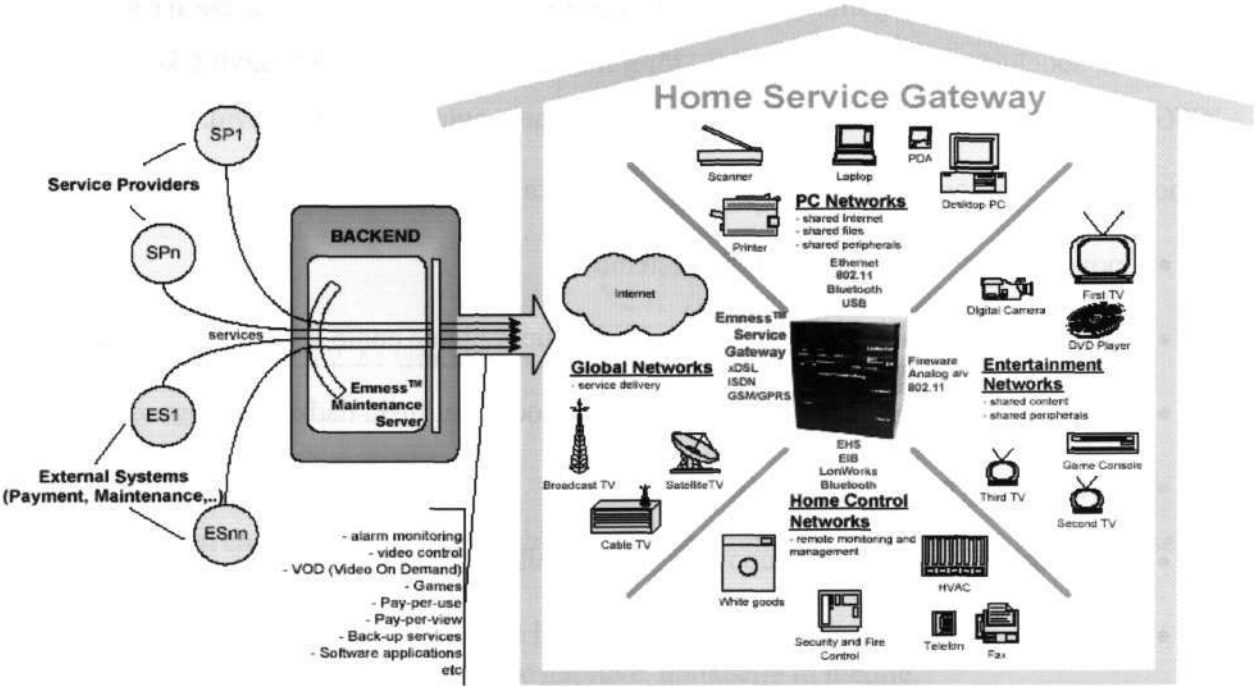
- povezovanje hišnih naprav in podsistemov,
- lokalni nadzor naprav in podsistemov preko upravljalne enote, ki je nanj povezana,
- povezava naprav na internet, s čimer je omogočen daljinski nadzor in diagnostika,
- optimizacija porabe energije,
- elektronske storitve (E-nakupovanje, E-bančništvo, E-kuhanje ...),
- beležke – pošiljanje sporočil med družinskimi člani,
- ...ipd.

Na strežnik je povezana upravljalna enota hišnega sistema, ki je lahko izvedena npr. kot vgrajen LCD prikazovalnik občutljiv na dotik, lahko pa to vlogo prevzame tudi katera od že obstoječih naprav v domu. S tem postane ta naprava vhodno/izhodna naprava sistema. Prikazovalnik oz. zaslon upravljalne enote je lahko nameščen v kuhinji, na posebnem mestu ali npr. na hladilniku. Za pomožne upravljalne enote lahko uporabimo tudi druge prikazovalne naprave (npr. barvni TV - še posebej v digitalni obliki, PC, GSM, dlančnik, ...).

Na sliki 2.3 vidimo primer sistema, ki ima več podsistemov:

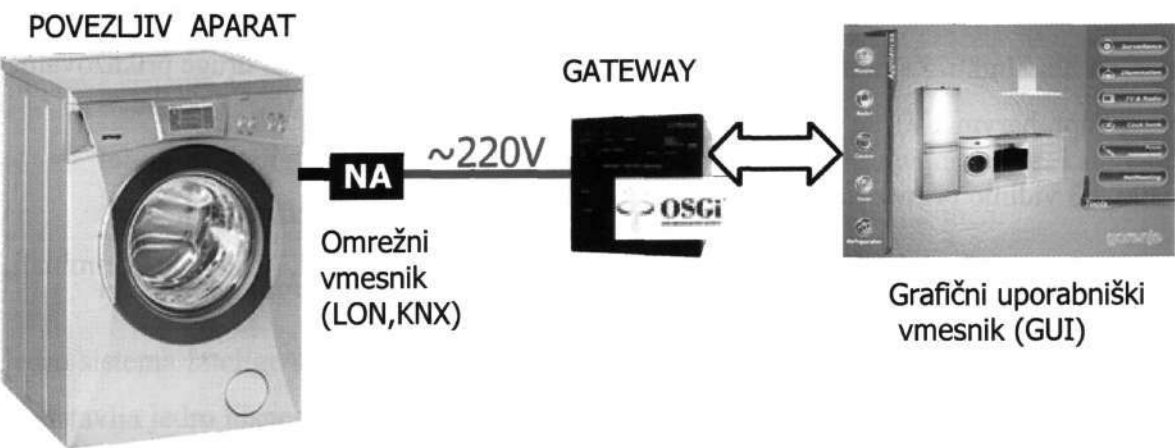
- računalniško omrežje, ki je že prisotno v večini domov (PC-ji, tiskalniki, terminali, dlančniki...),
- multimedijско omrežje (TV, DVD predvajalniki, kamere,...),
- omrežje gospodinjskih in hišnih naprav (ogrevanje, klimatiziranje, gospodinjski aparati...),
- ter del za povezavo v globalna omrežja (kabelska povezava, ISDN, ADSL...).

Preko tega nazadnje omenjenega dela se ustvari povezava s tako imenovanim *Service Agregator*-jem. Ta omogoča varno povezavo do hišnega omrežja ter ponuja vsemogoče storitve (*pay-per-use*, video na zahtevo, digitalna TV, servisne storitve, dobava dobrin,...)



Slika 2.3 : Sistem inteligentnega doma s storitvami

Podoben koncept (slika2.4) smo si zastavili tudi v Gorenju.



Slika 2.4 : Koncept razvoja sistema v Gorenju

3 KONNEX

To poglavje opisuje glavne elemente Konnex sistema (v nadaljevanju KNX), njegov koncept ter terminologijo. Informacije so uporabne predvsem kot kašipot za tiste, ki se prvič srečujejo s tem sistemom in KNX specifikacijo.

Tehnologija za nadzor in upravljanje stavb (*Building Control Technology*), ki jo zagotavlja KNX, je specializirana oblika avtomatiziranega procesnega nadzora, prilagojena potrebam hišnih oz. stavbnih aplikacij. Glavna naloga KNX-a je omogočiti osnovni decentralizirani, razporejen pristop do naprav, kar lahko imenujemo omrežje.

3.1 Nastanek Konnex združenja in osnovne lastnosti

KNX omrežje naprav [2] je rezultat združitve 3 vodilnih sistemov za **hišno in stavbno avtomatizacijo (EIB, EHS in BatiBus)** v novo specifikacijo **Konnex združenja (slika 3.1)**.

Slika 3.1 : Logotip Konnex združenja



V mesecu maju leta 1999 je devet podjetij - članic prej omenjenih združenj – ustanovilo Konnex združenje. Cilj tega je promocija enega samega komunikacijskega standarda imenovanega Konnex **KNX**, ki je nastal na osnovi znanja in izkušenj iz uporabe posameznih že obstoječih komunikacijskih protokolov. Osnovan je na komunikacijskem skladu EIB protokola in razširjen s fizičnim nivojem, nastavitvenim načinom in aplikacijskimi izkušnjami protokolov oz. standardov EHS in BatiBUS.

Trenutno to združenje obsega 100 vodilnih podjetij s celega sveta, ki se med drugim ukvarjajo tudi s hišno avtomatizacijo. To so predvsem podjetja, ki so aktivna na naslednjih področjih:

- Električna, elektronska in HVAC¹ oprema
- Bela tehnika in zabavna elektronika
- Dobavitelji električne energije
- Ponudniki telekomunikacijskih storitev
- Ponudniki storitev (montažerji HVAC in elektr. napeljave, sistemski integratorji, ...)

Glavne naloge združenja so:

- Promocija skupnega odprtega standarda KNX za hišno avtomatizacijo
- Aktivna podpora za trg hišne avtomatizacije
- Vzpodbujanje članov združenja za delo z globalnim standardom
- Visoka podpora članom z omogočanjem prijaznega delovnega okolja (pri administraciji, certificiranju, sistemski podpori,...) za razvoj in promocijo njihovih KNX produktov.

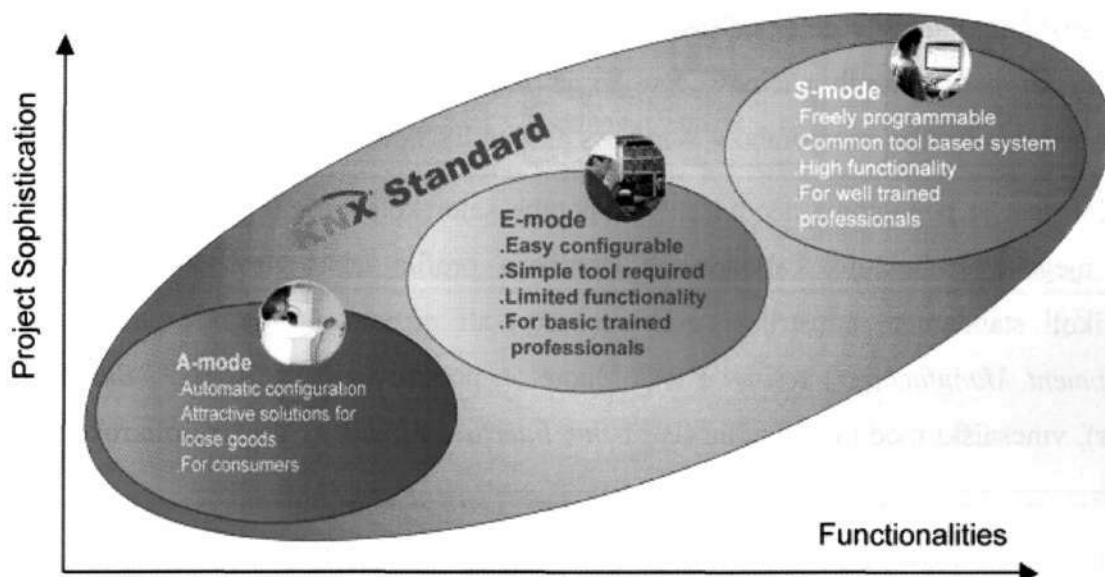
3.1.1 Osnovne lastnosti

Specifikacija KNX sistema ponuja poleg dobrih karakteristik v času izvajanja, še zmojljivo 'orodje' za podporo omrežja in mehanizme za njegovo upravljanje.

KNX standard pokriva tri različne nastavitvene načine (do danes) in štiri različne omrežne medije. Ravno zaradi raznolikosti uporabljenih medijev je sistem enostavno prilagodljiv na pogoje v določenih poslopih, ter za različne želje in zahteve končnih uporabnikov, kjer naj bi se sistem instaliral.

¹ HVAC – Heating, Ventilation and AirCondition

Vsi trije nastavitveni načini (Slika 3.2) - še posebej S-način in E-način – omogočajo tako certificiranim, kot tudi ostalim instalaterjem z osnovnim znanjem, da lahko ti uspešno testirajo in nastavijo celotno omrežje glede na potrebe in zahteve končnega uporabnika.



Slika 3.2: Nastavitveni načini KNX

Uporaba KNX standarda za komunikacijo naprav omogoča proizvajalcem enostavno vključitev le teh v vsako globalno hišno in stavbno omrežje, saj jim avtomatično zagotavlja popolno interoperabilnost z ostalimi napravami v sistemu.

Glavna opora S-načina je centralizirana svobodna povezava in parametrizacija (tipično z ETS orodji za PC okolje).

Združena je s profili naprav ki uporabljajo E-način in so lahko nastavljeni po pravilih strukturiranega povezovalnega principa z uporabo enostavnega upravljanja – brez uporabe PC orodja.

Ostane nam še A-mode – ki omogoča '**priključi in uporablja**' konfiguracijo (*plug and play*). Ta način je primeren predvsem za potrošniške produkte in naprave kot so npr. bela tehnika.

Vsi trije nastavitveni načini si delijo skupno interoperabilnost v času izvajanja, ki dopušča kreiranje obsežnega in več-domenskega hišnega komunikacijskega sistema.

3.2 KNX sistem z orodji

KNX eksplicitno predpisuje metodologijo in PC orodja za projektni inženiring; povezuje serije različnih naprav v delujočo instalacijo, ter integracijo različnih KNX komunikacijskih medijev in nastavitvenih načinov. Vse to je implementirano v ETS (*Engineering Tool Software*) in primerno za uporabo v Windows okolju. ETS orodje je neodvisno od dobavitelja.

KNX sistem je popolnoma neodvisen od uporabe katerekoli mikroprocesorske platforme in celo njegove arhitekture. Odvisno od izbranega profila lahko proizvajalec sam izbere katerikoli standarden industrijski mikroprocesor, ali razpoložljivo KNX OEM (*Original Equipment Manufacturer*) rešitev kot so enote za povezovanje vodil (BCU-*Bus Coupling Units*), vmesniški moduli za vodila (BIM-*Bus Interface Modules*), družine mikrokrmilnikov, ...itd.

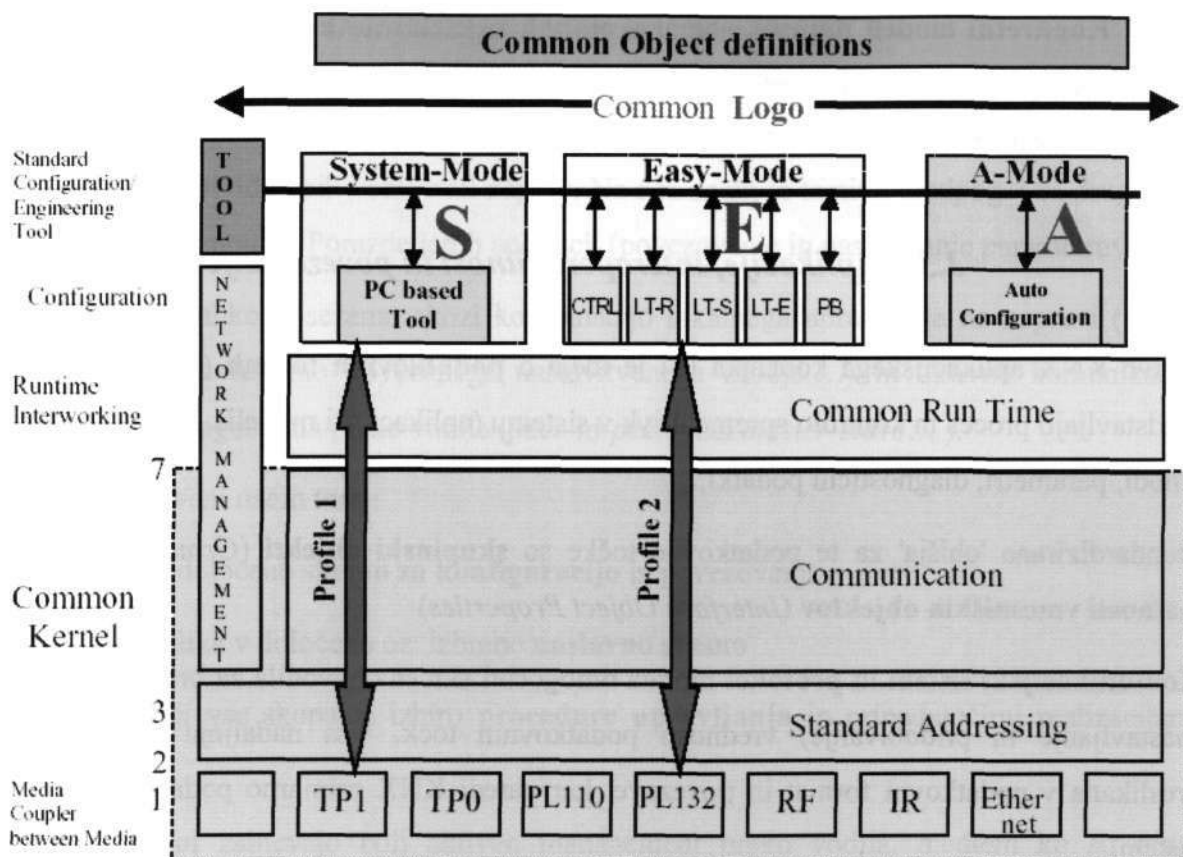
Nekateri KNX profili dopuščajo integracijo v majhen sistem (*firmware* < 5kb) in z lahkoto delujejo tudi na 8-bitnih procesorjih. Ostale implementacije uporabljajo 16- ali 32-bitne mikroprocesorje, ali celo PC-je v pravem pomenu besede.

Kot omenjeno, so KNX omrežja naprav fleksibilno prilagodljiva na trenutne pogoje in optimalne rešitve za vsako posamezno domeno in instalacijo. Imajo tudi možnost uporabe storitvenega omrežnega okolja za povečanje in razširitev koristi inteligentnega doma, pisarne ali poslovnega okolja. Za uporabo te funkcionalnosti priporoča KNX uporabo *ANubis* orodja.

3.3 Elementi KNX arhitekture

KNX specificira veliko mehanizmov in gradnikov za vzpostavitev delujočega omrežja, hkrati pa omogoča proizvajalcem naprav, da sami izbirajo za njih najbolj prilagodljivo konfiguracijo, primerno za posamezen ciljni trg.

Na sliki 3.3 vidimo pregled KNX modela, s poudarkom na različnih izbirah nastavitvenih načinov.



Ctrl = Controller Approach LT = Logical Tag (e.g. Code Wheel) PB = Push Button approach LTE = Logical Tag extended

Slika 3.3: KNX model

Glavne lastnosti KNX sistema so:

- **Interoperabilnost in (porazdeljeni) aplikacijski modeli** za različne funkcije hišne in stavbne avtomatizacije.
- **Shema za konfiguracijo in management** za pravilno uporabo vseh sistemskih virov, ter za omogočanje logičnega povezovanja delov porazdeljenih aplikacij, ki se izvajajo v različnih vozlih omrežja. KNX strukturira te v temeljni izbor nastavitvenih načinov.
- **Komunikacijski sistem**, z naborom različnih fizičnih komunikacijskih medijev, sporočilnim protokolom in ustreznim modelom za komunikacijski sklad v vsakem vozlu. Ta komunikacijski sistem mora podpirati vse zahteve omrežne komunikacije za konfiguracijo in management pri instalaciji, kot tudi za gostitev porazdeljenih aplikacij na omrežju. To je opredeljeno v KNX skupnem jedru (*KNX Common Kernel*)

- **Konkretni modeli naprav**, zbrani v profilih za učinkovito realizacijo in kombinacijo vseh prej naštetih elementov pri razvoju realnih produktov oz. naprav, ki so potem povezane v omrežje.

3.3.1 Aplikacije, interoperabilnost in povezovanje

Bistvo KNX aplikacijskega koncepta [3] je ideja o **podatkovnih točkah** (*Datapoints*). Te predstavljajo proces in kontrolo spremenljivk v sistemu (aplikacijski modeli). Lahko so vhodi, izhodi, parametri, diagnostični podatki,...

Standardizirana 'ohišja' za te podatkovne točke so **skupinski objekti** (*Group Objects*) in **lastnosti vmesniških objektov** (*Interface Object Properties*)

Komunikacijski sistem in protokol morata omogočati skržena navodila za branje in pisanje (nastavljanje in pridobivanje) vrednosti podatkovnih točk. Vsa nadaljnja semantika je preslikana v podatkovni format in povezave, kar naredi KNX primarno podatkovno voden sistem.

Za doseganje interoperabilnosti morajo podatkovne točke implementirati **standardizirane podatkovne tipe**, ki so grupirani v **funkcionalne bloke** (*Functional Bloks*).

Funkcionalni bloki skupaj s podatkovnimi tipi so povezani z aplikacijskimi polji. Nekateri od njih pa so v splošni rabi in jih imenujemo kar **funkcije s skupnim interesom** (npr. datum in čas).

Podatkovne točke so dosegljive skozi eno-oddajne ali več-oddajne mehanizme, ki razdvojijo komunikacijo in aplikacijske poglede, ter dovolijo gladko integracijo med implementacijskimi alternativami.

Za logično povezavo aplikacij skozi omrežje ima KNX tri osnovne povezovalne sheme: ena za prosto, druga za strukturirano in tretja za etiketirano (*tagged*) povezovanje. Natančnejši opis sledi v naslednjih poglavjih.

3.3.2 Osnovne nastavitvene sheme

Ob grobi razdelitvi obstajata dva nivoja konfiguracije sistema [3]. Prvič, nivo omrežne topologije in individualnih vozlov ali naprav. Na nek način je ta nivo predpogoj za drug nivo, ki obsega konfiguracijo Porazdeljenih aplikacij (povezovanje in nastavljanje parametrov).

Konfiguracijo lahko dosežemo skozi kombinacijo lokalnega upravljanja na napravi (pritisek gumba, uporaba lokalno povezanega nastavitvenega orodja,...) in aktivne komunikacije omrežnega managementa preko vodila (*peer-to-peer*, tudi *master-slave*...).

KNX nastavitveni način torej:

1. izbere določeno **shemo za konfiguracijo** in povezovanje
2. jo preslika v določeno oz. izbrano **naslovno shemo**
3. zaključi vse skupaj z izbiro **procedure upravljanja** in pripadajočimi realizacijami virov

Nekateri načini zahtevajo bolj aktiven management preko vodila, medtem ko so drugi predvsem orientirani v lokalno konfiguracijo.

3.3.3 Management omrežja in virov

Za prilagoditev vseh aktivnih nastavitvenih potreb sistema, ter za ohranitev 'harmonije' v raznolikosti, je KNX opremljen z močnim orodjem za omrežni management [3]. Ta instrument je uporaben skozi cel življenjski cikel instalacije:

- Začetna nastavitve,
- integracija več-točkovnih instalacij,
- posledično diagnosticiranje in vzdrževanje,
- kasnejše razširjanje omrežja in ponovna konfiguracija.

KNX omrežni management specificira niz mehanizmov za raziskovanje, nastavljanje in pridobivanje nastavitvenih podatkov iz omrežja. Predlaga proceduro (zaporedje sporočil) za dostop do različnih omrežnih virov znotraj naprav, kot tudi identifikacijskih podatkov in

formatov za te vire – vse to za omogočanje interoperabilnosti vseh KNX omrežnih naprav. Ti viri so lahko naslovi, komunikacijski parametri, aplikacijski parametri, ali kompleksnejši nizi podatkov kot so povezovalne tabele ali celo izvršljiv aplikacijski program.

Omrežni management v osnovi uporablja oz. omogoča uporabo storitev, ki jih ponuja aplikacijski nivo.

Kakorkoli že, sami mehanizmi in viri niso dovolj. Soliden omrežni management mora tudi dosledno vzdrževati in upoštevati konsistentna pravila (npr. izbira naslovov pri povezovanju ...itd).

3.3.4 Komunikacija :Fizični nivo

KNX sistem omogoča proizvajalcem samostojno izbiro kateri fizični medij ali njihovo kombinacijo bodo uporabili za komunikacijo med napravami. Ti mediji so lahko [3]:

- **TP 0** prevzet po BatiBus, in **TP 1** osnovni medij EIB; Priskrbita izboljšani rešitvi za zvito parico (*Twisted Pair* -TP).

Glavne značilnosti so: prenos podatkov in napajanja po enem paru, asinhroni znakovno orientirani prenos podatkov in poldupleksna obojestranska komunikacija. Prenos TP 0 je 2400 bit/s, TP1 pa 9600 bit/s. Oba medija imata implementirano CSMA/CA izogibanje trkom.

- **PL 110** (EIB) in **PL 132** (EHS) (*Powerline* -PL); omogočata komunikacijo preko običajnega napajalnega električnega omrežja.

Glavne značilnosti obeh so: razpršena frekvenčna modulacija signala, asinhroni prenos podatkov in poldupleksa dvosmerna komunikacija. Razlikujeta se po uporabi različne centralne frekvence (110 kHz in 132 kHz), uporabe dekodirnega procesa in prenosa podatkov (PL 110 = 1200 bit/s, PL 132 = 2400 bit/s). Oba medija uporabljata CSMA in sta kompatibilna s standardom EN 50065-1.

- **RF** (radio-frekvenčni medij); medij, je bil povsem specificiran znotraj Konnex združenja in omogoča brezžično komunikacijo v frekvenčnem območju 868 MHz. Glavne značilnosti so: frekvenčna modulacija signala, asinhrona komunikacija in poldupleksna obojestranska ali enostranska komunikacija. Centralna komunikacija je

868.30 MHz za naprave s kratkim dometom delovanja in delovnim razmerjem aktivnega delovanja pod 1% (*duty cycle* < 1%) ter podatkovnim prenosom 32kHz. KNX RF je kompatibilen z ERC/REC 70-03 in evropskim standardom ETS 300-220.

- **IR** (infra-rdeča povezava) je bil prevzet kar od EIB kot pridružen standard in bo služil kot osnova za bodoče implementacije.

Poleg teh zgoraj naštetih fizičnih medijev ima KNX skupno univerzalno rešitev implementacije za medije, ki omogočajo IP (*Internet Protocol*) komunikacijo, kot so *Ethernet* (IEEE 802.2), *Bluetooth*, *WiFi/Wireless LAN* (IEEE 802.11), *'FireWire'* (IEEE 1394)...itd.

3.3.5 Komunikacija: Skupno jedro in sporočilni protokol

Komunikacijski sistem se mora nagibati k potrebam aplikacijskih modelov, konfiguracije in omrežnega managementa [3]. Nad fizičnimi nivoji in njihovimi določenimi podatkovnimi nivoji je model skupnega jedra (*Common Kernel*), ki ga uporabljajo vse naprave KNX omrežja.

Komunikacijski sistem za izpolnjevanje vseh zahtev uporablja sedem nivojski OSI model:

- **POVEZOVALNI NIVO** – zagotavlja dostop in kontrolo medija in nadzor logične povezave
- **OMREŽNI NIVO** – zagotavlja segmentirano potrjevanje sporočil. Ta nivo je zanimiv predvsem za vozle z usmerjevalno funkcijo.
- **TRANSPORTNI NIVO** – omogoča štiri tipe komunikacijskih zvez med komunikacijskimi točkami: ena-na-več nepovezavno, ena-na-vse nepovezavno, ena-na-ena nepovezavno in ena-na-ena povezavno orientirano.
- **SEJNI in PREDSTAVITVENI NIVO** - sta prazna
- **APLIKACIJSKI NIVO** – omogoča raznolike možnosti aplikacijskih storitev za aplikacijski proces. Te storitve se razlikujejo glede na tip uporabljene komunikacije v transportnem nivoju.

3.3.6 KNX viri

Kot smo videli, je omrežni management sestavljen iz procedur za upravljanje z viri in skupnega jedra, ki omogoča delo s storitvami za ta namen [3].

Pod izrazom viri so v tem kontekstu lahko:

- **Sistemske (nastavitvene) viri**, z naslovi in informacijami o parametrih, ki omogočajo komunikacijskim nivojem v sistemu opravljati njihove naloge.
- **Parametri** za nadzor aplikacije

Trenutna KNX specifikacija podaja podroben opis vlog, identifikatorjev, formatov in kodiranja posameznega vira, ki je navsezadnje skupek podatkovnih elementov.

Definirani so lahko različni formati za dane abstraktne vire, ki omogočajo preprostejšo ali pa bolj dovršeno realizacijo, mogoče tudi realizacijo odvisno od nastavitvenega načina.

KNX vmesniški objekti predstavljajo okvir (neodvisen od implementacije) za realizacijo virov, ter posamezne elemente, ki so lahko modelirani kot podatkovne točke.

3.3.7 Modeli naprav

KNX instalacija vedno vsebuje niz naprav povezanih na neko vodilo ali omrežje. Vsi delčki sistema o katerih smo do sedaj govorili, so realizirani v ali skozi te naprave. Vsi pojmi se združujejo v številu različnih **logičnih vozliščnih arhitektur** za naprave, ki uporabljajo različne vire in implementacijo protokola [3].

Modeli se spreminjajo glede na vozliščne zmožnosti, managementske lastnosti in nastavitvene načine, ter seveda glede na vlogo, ki jo naprava v omrežju predstavlja (*Master, Router, Gateway...*)

KNX prav tako standardizira nekatere osnovno-namenske modele naprav (BCU, BIM...), ki so predvsem uporabljeni v kombinaciji z ETS orodjem in prenešenimi (*download*) programi.

3.3.8 Identifikacija naprav

Kot dopolnitev k osnovnemu operacijskemu sistemu, je zagotovljen niz identifikacijskih mehanizmov [3]:

- Naprave so lahko identificirane in kasneje dosegljive preko omrežja skozi njihove individualne naslove ali preko njihove unikatne serijske številke, odvisno od nastavitvenega načina.
- Instalacijska 'raztegljivost' in vzdrževanje je precej olajšano s produktno identifikacijo (določena pri proizvajalcu) in funkcionalno identifikacijsko informacijo (neodvisna od proizvajalca), ki jo pridobimo iz naprave.

Naloge mehanizmov, ki uporabljajo unikatno serijsko številko naprave, so:

- Pridobitev individualnega naslova iz dane serijske številke naprave
- Nastavitev individualnega naslova iz dane serijske številke naprave
- Obnovitev serijske številke naprave iz podanega individualnega naslova

Unikatnost serijskih številk je zagotovljena preko nadzorovanega dodeljevanja števil, katerega območje je domena oddelka za certificiranje KNX (*Konnex Association Certification Department*).

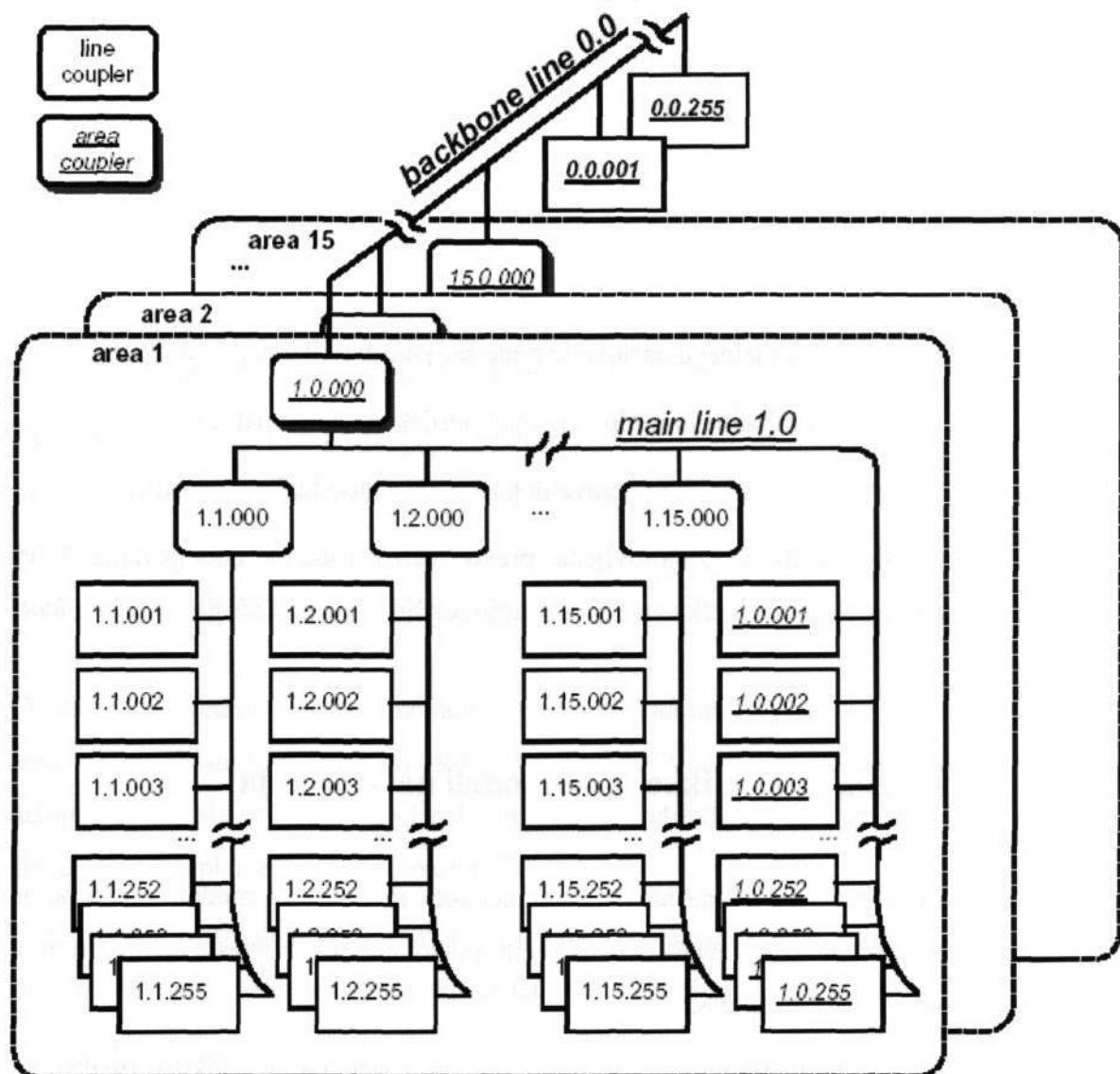
3.4 Sistem, komunikacija in modeli naslavljanja

Preden se spustimo v nastavitvene načine, aplikacijske modele, ter modele medsebojnega delovanja, si za boljše razumevanje podrobneje pogledjmo sheme naslavljanja in s tem povezane komunikacijske načine KNX [3-4].

Kodirni prostor za naslavljanje popravi osnovno zmožnost sistema po velikosti (maksimalno število naslovljivih naprav in podatkovnih točk). Naslavljanje se reflektira v kodirnem formatu sporočila kot 'senca' komunikacijskega protokola in jedra.

3.4.1 Logična topologija in individualni naslovni prostor

KNX je popolnoma razpršeno omrežje, ki lahko oskrbuje do 65.536 naprav v 16-bitnem individualnem naslovnem prostoru [3]. Logična topologija ali struktura pod-omrežij dovoljuje 256 naprav na eni **liniji**, kot je prikazano na sliki 3.4.



Slika 3.4: Logična topologija KNX omrežja

Linije so lahko povezane v skupino preko **glavne linije** (*main line*) v skupno **območje** (*area*). Celotna domena pa je formirana iz 15-ih območij skupaj z povezovalno linijo imenovano **hrbtenjačo** (*backbone line*).

KNX ANubis orodje omogoča tudi integracijo oz. povezovanje pod-omrežij preko IP.

Na sliki 3.4 vidimo tudi, da nam takšna topologija omrežja omogoča tudi zelo pregledno numerično strukturo individualnih naslovov, ki določajo unikatno identifikacijo vsake naprave oz. vozla na omrežju (z nekaj izjemami).

Pri uporabi električnega omrežja kot fizičnega medija za prenos komunikacije, so sosednje domene med seboj logično ločene z 16-bitnim hišnim naslovom (*Domain Address*).

Glede na visoko logično omogočeno število naprav v omrežju (preko 65 tisoč), se dejanske omejitve pokažejo pri sami instalaciji omrežja. Odvisne so od same implementacije omrežja (uporabljenega medija za prenos, tipa oddajno-sprejemnih enot, zmožnosti napajalnih napetosti) in tudi od različnih faktorjev iz okolja (elektromagnetni šum,...).

3.4.2 Omrežni management

KNX za upravljanje (konfiguracijo) omrežja in naprav uporablja kombinacijo razpršenega oddajanja (*broadcast*) in točka-točka (*point-to-point*) komunikacije [4].

Najpogosteje je vsaki novi napravi pri instalaciji dodeljen individualen naslov preko razpršenega oddajanja (opcijsko se lahko uporabi tudi unikatna serijska številka naprave), ki se od te točke naprej uporablja za točka-točka komunikacijo.

Podprto je '**polno**' več-prejemniško naslavljanje, kar je glavna opora KNX komunikaciji v času delovanja. 'Polno' pomeni:

1. KNX ni omejen na grupiranje naprav: vsaka naprava lahko individualno objavi nekaj podatkovnih točk (poznanih kot skupinski komunikacijski objekti), ki so lahko nato neodvisno urejene po skupinah v vse-omrežne souporabniške spremenljivke. Prav tako pa so lahko objavljene tudi lastnosti vmesniških objektov, kot skupne spremenljivke.
2. Souporabniška oz. skupna spremenljivka se lahko v popolnosti bere/piše v obeh smereh. Na ta način lahko vse naprave pošiljajo tudi nenaročene večoddajne okvirje.
3. KNX ima za te skupne spremenljivke na voljo 16-bitni naslovni prostor. To pomeni, da ima lahko ena instalacija vse do 64k skupnih spremenljivk (oz.

skupinskih naslovov), vsaka pa ima katerokoli število lokalnih primerov oz. instanc.

Na ta način KNX teži k zmanjšanju odvečne avtomatizacijske hierarhije nivojev skozi primerno naslavljanje in modeliranje shem za naprave.

3.4.3 Okvirji KNX sporočila

Poglejmo si še dejanski format KNX sporočila, ki je serijsko kodiran v okvirje oz. telegrame, ki so nato poslani na vodilo [4].

Odvisno od modulacijske tehnike ali od dostopa in nadzora trkov izbranega medija, so lahko definirane nekatere preambule ali zaporedne ovojnice (kar mi tukaj ignoriramo).

Na naslednjem primeru okvirja si pogledjmo dejansko odgovarjanje na vmesnik nad drugim komunikacijskim nivojem. Specialna potrditev okvirjev ipd. je natančno opisana v KNX specifikaciji.

octet 0	1	2	3	4	5	6	7	8	...	N - 1	N ≤ 22
Control Field	Source Address		Destination Address		Address Type: NPCI; length	TPCI	APCI	data/ APCI	data		Frame Check

Slika 3.5: KNX standardna struktura sporočila

Kontrolno polje (*Control Field*) določa prioriteto okvirja in razlikuje med standardnimi in razširjenimi okvirji (sporočili). V vsakem primeru obstaja individualen izvorni naslov (*Source Address*) in individualen ali skupinski ciljni naslov (*Destination Address*); tip tega je določen v posebnem polju.

TPCI (*Transport layer Protocol Control Information*) nadzira transportni nivo komunikacijskih zvez (za vzpostavitev in vzdrževanje točka-točka zvez).

APCI (*Application layer PCI*) lahko prisluškuje storitvam aplikativnega nivoja (beri, piši, odgovori,...), ki so dosegljive za pomembno shemo naslavljanja in komunikacijskih zvez.

Odvisno od naslovne sheme in APCI, lahko standarden okvir prenese vse do 14 oktetov podatkov. Segmentacija za masovne prenose, kot je prenos celotne aplikacije programa, je odgovornost upravljalškega odjemalca (*Management Client*) (npr. ETS orodje).

Standarden okvir zagotavlja direktno nadgrajeno združljivost z EIB standardom. Razširjen okvir pa lahko obvladuje do 248 oktetov podatkov.

Na koncu, kontrola okvirja (*Frame Check*) pomaga zagotavljati podatkovno konsekvantnost in zanesljiv prenos podatkov.

3.5 Aplikacijski modeli, podatkovne točke in povezave

Vsi elementi KNX arhitekture, ki smo si jih do sedaj predstavili, služijo predvsem za infrastrukturo in za doseganje aplikacij kot so npr. razsvetljava, HVAC, varnost,... ter njihovo delovanje na sistemu [3].

KNX v bistvu modelira aplikacijo na omrežju naprav kot zbirko podatkovnih točk za sprejemanje in oddajanje, ki so porazdeljene preko števila naprav.

Sistem oživi, ko podatkovne točke v različnih napravah povežemo preko skupnega identifikatorja, kot je primer z več-oddajnimi skupinskimi naslovi.

Podatki so torej prenosljivi med različnimi napravami, ki imajo svoje lokalne aplikacije oz. programe, kar je v bistvu tudi glavni namen obstoja omrežja.

Ko lokalni program v napravi (npr. senzor) zapiše vrednost k pošiljajoči oz. oddajni podatkovni točki (naprej PT), ta pošlje sporočilo z ustreznim naslovom in novo vrednostjo.

Sprejemna PT z enakim naslovom to vrednost sprejme in o tem obvesti svojo lokalno aplikacijo oz. program. Ta lahko potem skladno s to novo vrednostjo ukrepa, če je to potrebno.

Ta ukrep je lahko sprememba internega stanja ali posodobitev ene od svojih oddajnih PT, kot je recimo krmilnik, možna pa je tudi sprememba katerega od izhodnih stanj, ali kombinacija katere od teh možnosti.

Na ta način lokalne aplikacije v različnih napravah s povezanimi PT tvorijo porazdeljeno aplikacijo.

KNX obsega tri osnovne sheme za povezovanje PT glede na pomensko informacijo, ki jo vsebuje sam naslov (semantična informacija) in glede nato, ali je povezovanje natančno vnaprej določeno ali le sledi nekaterim bolj ohlapnim pravilom. Iz tega sledijo tri različne klasifikacije:

- **Prosto povezovanje** (*free binding*)
- **Strukturirano povezovanje** (*structured binding*)
- **Etiketirano povezovanje** (*tagged binding*)

Preden si te tri načine povezovanja podrobneje ogledamo, je potrebno razjasniti še nekaj pojmov.

3.5.1 Skupinski objekti

Osnovna oblika realizacije KNX podatkovnih točk (PT) je podana s skupinskimi objekti [3]. Kot nakazuje že samo ime, so ti dosegljivi preko standardnega skupinskega naslavljanja. V kombinaciji s standardnim formatom skupinskega sporočila, vse skupaj tvori osnovne temelje sistemskega navzkrižnega medsebojnega delovanja (*Interworking*) in večvrstne integracijske storitve.

Skupinski objekti so lahko v uporabi pri vseh treh prej omenjenih načinih povezovanja. Odvisno od same aplikacije in nastavitvenega načina, se spreminja tudi osveščenost lokalnega programa.

V tem pogledu KNX dovoljuje zelo zanimiv inkapsulacijski vzorec, ki lokalnemu programu zagotavlja sistematičen, od povezovanja neodvisen vmesnik na vodilo. Z uporabo te aplikacije, ki uporablja skupinsko naslavljanje, lahko učinkovito dosegamo najbolj radikalno razdvajanje v omrežni komunikaciji. To vzpodbuja uporabo iste aplikacijske kode za različne nastavitvene načine.

Lokalna aplikacija vidi vodilo kot omejen nabor skupinskih objektov. Ti ustrezajo tistim PT, ki so zanje neposredno pomembne. Če povemo drugače: vsak komunikacijski objekt

predstavlja svoji aplikaciji lokalno spremenljivko s pripadajočimi atributi. Ta spremenljivka pa vzdržuje vrednost, ki je bila sprejeta iz ali pa bo poslana na vodilo. Preko atributov lahko privzeti upravljavec (nad komunikacijskim skladom vozla) obvešča aplikacijo o spremembi ustrezne vrednosti, in obratno; aplikacija lahko zahteva od sklada, da le ta pošlje določeno vrednost. S tem lahko predvidevamo ciklično povpraševanje na obeh straneh vmesnika. Bolj dovršena implementacija lahko preslika ta vmesnik k prilagojenem upravljavcu povratnega klica.

Nekatere verzije skupinskih komunikacijskih skladov podpirajo ta način z dvema nivojema posrednosti. Transportni nivo pretvori sprejet skupinski naslov v čisto lokalno interno navodilo (z uporabo tabele virov skupinskih naslovov). Sedaj mora aplikacijski nivo poskrbeti za obdelavo in preslikavo tega internega navodila k enemu ali k več-številčnim identifikatorjem komunikacijskih objektov (tudi z uporabo multipleksiranja).

3.5.2 Lastnosti vmesniških objektov kot PT

Za ugoditev dodatnih zahtev, KNX omogoča tudi bolj popolno predstavo PT v obliki lastnosti, ki pripadajo posameznim vmesniškim objektom. Vmesniški objekt preprosto grupira niz lastnosti PT v skupno vmesniško strukturo ali objekt.

Medtem, ko skupinski objekti vozla predstavljajo osnovni niz PT, ki so vse direktno naslovljive, se vsaka lastnost vmesniškega objekta relativno sklicuje na ta objekt – glede na vzorec:

<Object Reference>, <Property> .

Glede na to, je vmesniški objekt primeren za modeliranje funkcionalnih blokov .

Vmesniški objekti niso omejeni na aplikacijske PT, poleg tega omogočajo tudi tako imenovano '*Datapoint-style*' modeliranje upravljavskih virov v neki napravi.

Vmesniški objekti se pri standardnem naslavljanju z indeksom in pri točka-točka komunikaciji vsebinsko navezujejo na vozle v omrežju. V tem načinu so za tipične komunikacije v času delovanja uporabljeni v A-načinu nastavitve.

3.5.3 Prosto ali strukturirano povezovanje

Spoznali smo že KNX komunikacijo in modele naslavljanja z upoštevanjem razlik pri razpršeni oddaji ter oddaji več ali enemu prejemniku [3]. Jasno je, da pride naslavljanje v poštev takrat, ko je potrebno povezati različne lokalne aplikacije, ki delujejo v različnih vozlih na omrežju.

V tem in naslednjem poglavju si pogledjmo princip povezovanja, kjer je kot osnova upoštevan način izbire naslovov, ne glede na to, na kakšen način je sporočilo distribuirano.

Z vidika omrežnega upravljanja je način povezovanja razločevalna karakteristika pri nastavitvenih načinih.

Da lahko vzpostavimo povezavo do nekega komunikacijskega partnerja, moramo najprej izvesti naslednja dva koraka:

1. potrebno je izbrati numerično vrednost za naslov
2. naslov je potrebno pripisati ustrezni PT za povezavo

To pomeni, da izbrana numerična vrednost ne nosi oz. ne vsebuje aplikacijske semantike (predstavlja le naslov). Iz tega lahko sklepamo le to, da so vsem PT, ki želijo direktno komunicirati med seboj, dodeljeni enaki naslovi. Ta koncept je ravno nasprotje etiketiranemu povezovanju, katerega opis sledi v naslednjem poglavju.

Naslednji vidik je dodelitev izbranega naslova, kjer razlikujemo prej omenjene kategorije:

- **Prosto povezovanje** – tukaj ni a priori recepta po katerem bi povezovali PT med seboj. Je le nekaj zelo splošnih pravil, kot je npr. ustaljenost enakega podatkovnega tipa v kombinaciji s prostim naslavljanjem, kar podpira 'multicast' grupiranje po meri na nivoju individualnih PT. Prosto povezovanje je značilno za S-način nastavitve.
- **Strukturirano povezovanje** – aplikacijski model v KNX specifikaciji zagotavlja natančen vzorec za povezovanje celega niza PT, ki ponavadi ustreza funkcionalnim blokom ali kanalom. Kljub temu pa je izbira vrednosti naslova povsem prosta. CTRL-način, Pritisni gumb način, LT-S in A-način način sledijo temu konceptu (opisi omenjenih načinov so v poglavju 3.7).

3.5.4 Etiketirano povezovanje

Kar nekaj načinov uporablja pristop etiketiranega povezovanja [3]. To povezovanje je vnaprej določeno oz. strukturirano v aplikacijskih modelih – v tem primeru vrednost naslova ni več poljubna. Del naslova, imenovan semantični identifikator PT, neposredno že vsebuje ciljno PT komunikacijskega partnerja.

Za E-način je etiketirano povezovanje tesno povezano s postavljanjem con. Logična etiketa (*tag*) ali conirni del naslova izbere predvidenega komunikacijskega partnerja na nivoju naprave.

Za dano PT je semantika identifikatorja fiksirana z aplikacijskim modelom; cona je nastavljena (pogosto kar lokalno na napravi) za točno določen primer pri dani instalaciji sistema. Ta koncept oskrbi enostavno povezovanje med napravami ali funkcijskimi bloki, seveda z upoštevanjem vnaprej določenega aplikacijskega modela.

Če so PT pripisane isti coni, te formirajo skupino – tako coniranje podpira tudi več-oddajni princip. A-način uporablja etiketiran model pri pristopu strežnik-odjemalec.

Obstajajo tri možnosti etiketiranega povezovanja:

1. Etiketa je preslikana v standardno skupinsko naslavljanje

Nekateri E-načini nastavitve preslikajo semantični identifikator k povezovalni kodi (*Connection Code*) v obliki fiksnega območja prostora skupinskih naslovov.

2. Individualno naslavljanje

Za oskrbo komplementarnih aplikacij in nastavitvenih zahtev, KNX podpira tudi PT, kjer etiketa ni del skupinskega naslova. Te implementacije izkoriščajo od implementacije neodvisno strukturo KNX vmesniških objektov, ki dopuščajo, da individualna PT pomeni oz. predstavlja lastnost takšnega objekta.

3. Razširjeno naslavljanje

Tudi v tem primeru je lahko uporabljen enak princip kot pri točki (2.). Prostor za skupinske naslove pri LTE načinu je ustvarjen posebej za ta namen in je

uporaben le za informacijo o coni. Povezovalna koda (ali semantični ID) je zgrajena iz *Objektnega tipa + Lastnosti ID*.

Za zaključek: aplikacijski modeli za etiketiranje imajo močno in natančno povezovalno semantiko, ki je že fiksirana v samem modelu, ki je vgrajen v napravo preko semantičnega identifikatorja. Kot nasprotje pa je s prostim povezovanjem omogočeno načrtovanje specialnih aplikacij za potrebe individualnih instalacij.

3.6 Model medsebojnega delovanja

Spoznali smo že vse osnovne gradnike sistema KNX in zato si pogledajmo še njegovo delovanje oz. koncept [4]. Glavni cilj KNX sistema je omogočiti odprt sistem za inteligentne stavbe, ki podpira komunikacijo naprav različnih proizvajalcev – to je v bistvu najpomembnejša funkcija KNX sistema, tako za instalaterje in integratorje, kot tudi seveda za končne uporabnike. Medsebojno delovanje omogoča izdelavo zelo raznolikega sistema z zelo bogato možnostjo integracije velikega spektra raznovrstnih naprav.

3.6.1 Aplikacija: Oblika PT in funkcionalni bloki

KNX principi medsebojnega delovanja se predvsem opirajo na aplikacijski model, ki smo ga opisali v prejšnjih poglavjih.

V glavnem opisuje posamezno aplikacijo oz. program z vidika omrežja; z drugimi besedami, odgovarja na vprašanje: kaj je vmesnik med programom in njegovo PT? Vsebovati mora opis delovanja in obnašanja naprave, kar se ponavadi opiše z internimi stanji stroja (*state machines*), sporočili in fizičnimi vhodi/izhodi – vse to določa tako imenovane opise **funkcionalnih blokov** vsakega dela porazdeljene aplikacije.

Znotraj specifikacije funkcionalnih blokov je vsaki PT dodeljeno pojasnjevalno ime, skupaj z zahtevanim tipom. Tip PT fiksno določa format podatkov, ki jih PT pošilja na ali sprejema iz podatkovnega vodila.

Jedro družine KNX podatkovnih tipov obsega tipe za :

- Binarne vrednosti (*boolean*)
- Kontrole ustreznosti (*Relative control*)(%)
- Analognih vrednosti (*long in short float*)
- Vrednost števca (*signed in unsigned integer*)
- Datum in čas
- Status (*bit field*)
- ...

Za implementacijo pri etiketiranem povezovalnem modelu mora opis funkcionalnega bloka specificirati tudi (standarden) semantični identifikator svoje PT.

3.6.2 Parametri PT

Aplikacijske spremenljivke PT, ki kažejo te osnovno-namenske tipe, pokrivajo večino elementarnih komunikacijskih potreb za operacije v času delovanja. Spremenljivke posameznih aplikacij le tem omogočajo delovanje njihovih bistvenih funkcij.

Kot dopolnilo k spremenljivkam najdemo **parametre**. Specializirane PT namreč včasih zahtevajo bolj specifične tipe, ki ponujajo bolj fin in dovršen nadzor nad osnovnim vodenjem aplikacije.

Pri uporabi S-načina in A-načina nastavitve so lahko parameterske PT implementirane na 'bolj privaten' način brez poseganja v osnovna opravila aplikacije. Pod izrazom 'bolj privaten' razumemo, da so lahko te parametrske PT dosežane le s strani nadzornika z določenim znanjem o implementaciji. Po drugi strani, to predvideva na nevtralen način dosegljivo potrebno znanje, kot je primer pri ETS orodju za projektno načrtovanje (ob pomoči detajlnih informacij iz proizvajalčeve baze podatkov za pomembne produkte in aplikacije, ki po možnosti vsebujejo tudi podroben opis pomnilnika).

3.7 Nastavitveni načini

Posvetimo se spet problemu konfiguracije sistema [4]. Za naslavljanje raznolikih potreb vsebuje KNX specifikacija paket orodij za upravljanje lastnosti, ki omogočajo izbiro med nekaj nastavitvenimi načini, prilagojenimi za različne potrebe posameznih trgov, lokalnih navad, stopnje potrebnega izobraževanja ali aplikacijskega okolja.

Specifikacija do neke mere proizvajalcu zagotavlja svobodo, medtem ko zagotavlja stanovitnost in medsebojno delovanje v izbranem nastavitvenem načinu, tudi pri uporabi naprav različnih proizvajalcev. Vsi nastavitveni načini vsebujejo pogoje oz. pravila, ki omogočajo razširitev ali modifikacijo instalacije znotraj izbranega načina.

ETS orodje omogoča večvrstne instalacije z možnostjo razširljivosti in medsebojnega delovanja. Glavni razlog za to je uporaba trdnih upravljavskih procedur v času delovanja, izmenjave podatkov preko skupinskih objektov, ki so dosegljivi skozi skupinsko naslavljanje in več-prejemniško komunikacijo. ETS uporablja funkcijo opisa naprave (*device descriptor feature*), ki omogoča razpoznavanje tipa naprave in uporabljenega nastavitvenega načina. Nato uporabi upravljavsko proceduro, ki odgovarja prejeti informaciji.

Naprave skladne z določenim nastavitvenim načinom naj bi implementirale omrežno upravljanje in profile za čas delovanja, kot je navedeno v pomembnem delu KNX specifikacije (*Volume 6: del 3/5 'Management'*)

3.7.1 Sistemski nastavitveni način(S-Mode)

Naprave, ki implementirajo S-način, ponujajo najbolj prilagodljiv večnamensko uporaben nastavitveni proces, medtem ko dovoljujejo kompaktno implementacijo: kompleksnost povezovanja in aplikacijske konfiguracije je prenesena na močnega nastavitvenega upravljavca. To vlogo ponavadi prevzame množica orodij za PC okolje iz ETS družine, ki ga dobavlja Konnex združenje. S pomočjo ETS projektnega orodja je možno nastaviti te naprave in jih spraviti v delovanje.

Za pridobivanje specialne informacije o napravah, ki so potrebne za to, ETS uporablja podatkovno bazo, ki vsebuje opise vseh možnih funkcionalnosti naprav, ki jih podpira. Ta

predloga za naprave (*product template*) je ustvarjena in oskrbovana s strani proizvajalca naprave, ki uporablja ETS orodje.

Ponavadi proizvajalec dopolnjuje to bazo podatkov za končnega uporabnika. Izučeni instalater lahko tako uvozi v bazo podatkov produktne predloge, ki izvirajo od različnih proizvajalcev in mu kljub temu dovoljujejo izgradnjo kompleksnega KNX omrežja. Na ta način lahko izbira med širokim spektrom funkcionalnosti različnih proizvajalcev.

ETS orodje za KNX projektno načrtovanje podpira konfiguracijo za naslednje lastnosti:

- **Povezovanje:** nastavitve skupinskega naslova za omogočanje skupinske objektne komunikacije med funkcionalnimi bloki. Skupinski objekti so lahko postavljeni v skupino, če si delijo enak tip PT. Tabele naslovov in tabele posrednosti so zgrajene s strani nastavitvenega upravljalca (kot je ETS), in so nato naložene v napravo.
- **Parametrizacija:** nastavljanje parametrov naprav glede na dokumentacijo proizvajalca. Nekateri parametri so standardni za upoštevanje funkcionalne bloke, drugi pa so lahko specifični glede na proizvajalca.
- **Nalaganje (*Download*)** aplikacijskega programa je mogoče tudi za večnamenske naprave s to posebno funkcijo. To so posebne naprave, ki vsebujejo dva fizična dela (BCU z vgrajenim *flash* spominom + izmenjevalni aplikacijski modul) in lahko ponujajo različne funkcije odvisno od izbranega naloženega aplikacijskega programa. Kakorkoli že, glede na omejeno razmerje signal/perioda, profili za aplikacijski prenos niso predvideni za RF.

Vse to omogoča KNX instalaterju da načrtuje in 'ukroji' želeno funkcionalnost porazdeljene aplikacije (ki natančno odgovarja zahtevam in potrebam vsake individualne instalacije) s pomočjo ETS orodja in uporabe gradbenih blokovnih knjižnic proizvajalca v obliki lastne produktne baze podatkov. To se nanaša predvsem na koncept prostega povezovanja.

Skrajšan opis: S-način uporablja standarden format okvirja. Potrebuje aktivno omrežno upravljanje (kot ga zagotavlja ETS) z razpršenim oddajanjem in individualnim naslavljanjem. Povezave v času delovanja privzeto uporabljajo skupinske naslove v območju za prosto povezovanje; dizajn dopušča dodajanje etiketirane ali potrditvene povezave za doseganje medsebojnega delovanja pri E-načinu osnovanem na povezovalnih kodah.

3.7.2 CTRL nastavitven način

CTRL-način je definiran za podporo instalacij z omejenim številom naprav na enem logičnem segmentu fizičnega medija. Instalacija, ki uporablja ta način vsebuje specialno napravo imenovano krmilnik (*Controller*), ki je zadolžena za podporo nastavitvenega procesa. Krmilnik podpira eno ali več aplikacij (npr. razsvetljavo). Ni nujno, je pa priporočljivo, da krmilnik ostane prisoten pri instalaciji tudi v času delovanja.

KNX omrežje naprav, ki implementira ta način, kaže različne funkcionalnosti z omejeno parametrizacijo, kot je opisano v ustrezni aplikacijski specifikaciji. Opremljeno je z zmožnostjo dinamične konfiguracije za nastavitev zahtevanega individualnega ali skupinskega naslova in parametrov.

V času konfiguracije je vloga krmilnika vzpostavitev povezave med tako imenovanimi kanali, ki predstavljajo niz skupinskih objektov za dano funkcionalnost. Povezave in parametri so identificirani s posegom instalaterja. Poseg je lahko različen od enega do drugega proizvajalca krmilnika. Nabor kanalov, ki jih podpira KNX specifikacija, je edinstveno specificiran. Posledično lahko napravo kateregakoli proizvajalca vzame v nadzor krmilnik kateregakoli proizvajalca. Krmilnik ne potrebuje znanja o funkcionalnostih, ki jih podpirajo naprave. Te informacije dobi oz. jih prebere iz samih naprav in jih nato zakodira v opis naprave, ki je implementiran z vsako napravo.

Upoštevajoč navodila instalaterja krmilnik pripiše napravam individualne naslove in preračuna povezave na nivoju PT z uporabo znanih pravil, ki so del specifikacije. Nato nastavi še parametre, ki so dosegljivi za uporabljene naprave. Operacija je v času delovanja neodvisna od nastavitvenega krmilnika.

Skrajšan opis: CTRL-način uporablja skupinsko komunikacijo s standardnim formatom okvirja. Potrebuje aktivno omrežno upravljanje (kot ga zagotavlja ETS) z razpršenim oddajanjem in individualnim naslavljanjem. Povezave v času delovanja privzeto uporabljajo skupinske naslove za strukturirano povezovanje.

3.7.3 Način 'Pritisni gumb' (Push-button Mode)

Ta način je tako kot CTRL-način namenjen za podporo instalacij z omejenim številom naprav na enem logičnem segmentu fizičnega medija. Tu sicer ne potrebujemo posebne naprave (krmilnika) za konfiguracijo, ampak imajo vse naprave, ki uporabljajo ta način, vključena sredstva (*means*) za konfiguracijo v povezavi z svojo aplikacijo. Te naprave vsebujejo fiksno parametrizirano funkcionalnost, kot je opisano v ustrezni aplikacijski specifikaciji.

Vsaka naprava je opremljena s sposobnostjo za dinamično konfiguracijo z nastavljanjem individualnih in skupinskih naslovov in parametrov. Izmenjava parametrov je ravno tako mogoča, vendar je v večini lokalne narave na napravah.

V času konfiguracije instalater zaporedno določa napravam funkcije, ki bodo nato povezane. Na kakšen način se to dejansko izvaja, je odvisno od proizvajalca naprave. Izmenjava nastavitvenih podatkov, ki se izvršuje med napravami, je izvedena skozi en sam servis na aplikacijskem nivoju. Vsaka oddajna PT naprava zahteva svoj lasten unikatni skupinski naslov. Ta naslov bo posredovan k sprejemni PT z uporabo 'pritisni gumb' procedure. Konfiguracija se pokorava pravilom kanalov in opisom funkcionalnih blokov, ki so podani v KNX specifikaciji.

Skrajšan opis: Za konfiguracijo se 'pritisni-gumb' način zanaša le na aktivno upravljanje ene naprave direktno na drugo. Uporablja skupinsko naslavljanje s standardnimi okvirji in strukturiranim povezovanjem.

3.7.4 LT (Logical Tag) nastavitveni način

LTR (Logical Tag Reflex)

Pri LTR načinu nastavitveno orodje ni potrebno. Naprave in njihove funkcionalnosti so opremljene z mehanizmi za pripis vrednosti, ki morajo biti upoštevane pri konfiguraciji.

Ustrezne funkcije v različnih napravah, ki so jim bile pripisane enake vrednosti za etiketo, so med seboj povezane in skladno delujejo. Možne so več-kanalne naprave. Te dobijo zaporedne etikete za vsak posamezen kanal. Operacija v času delovanja je ravno tako osnovana na

skupinskih objektih, ki so v pogledu kanalov in funkcijskih opisov usklajeni s KNX standardom.

Za vzpostavitev pravih povezav se mora razlaga etikete podrežati pravilom, ki so podana v KNX specifikaciji (poglavje '*Resources*'). Razlaga je odvisna od standardne povezovalne kode za pomembne kanale.

Skrajšan opis: Z vidika omrežnega upravljanja *LTR* naprave izgledajo kot zbirka, ker nikoli ne posedujejo posameznega individualnega naslova. Uporabljajo standardni okvir, z etiketiranim povezovanjem, ki je preslikan na standardne skupinske naslove.

LTS (Logical Tag Supervised)

V *LTS* načinu prav tako ne potrebujemo nastavitvenega orodja, potreben pa je nadzornik za aplikacijo. En nadzornik lahko nadzira več kot eno aplikacijo. Naprave in njihove funkcionalnosti so opremljene z mehanizmi za pripis vrednosti, ki morajo biti upoštevane kot individualni naslovi pri konfiguraciji.

Nadzornik uporablja vstavljen opis, ki ga nudi opis naprave in standardne mehanizme za vstavljanje individualnih naslovov, enega za drugim, ter različnih skupinskih naslovov, ki jih naprave potrebujejo v času delovanja. Skladne funkcije v različnih napravah, ki jim je bila dodeljena enaka vrednost za skupinski naslov, so povezane skupaj in lahko medsebojno delujejo. Omogočene so tudi več-kanalske naprave. Operacija v času delovanja je ravno tako osnovana na skupinskih objektih, ki so v pogledu kanalov in funkcijskih opisov usklajeni s KNX standardom.

Skrajšan opis: *LTS* način uporablja standardni format okvirja. Zahteva aktivno omrežno upravljanje z uporabo razpršenega oddajanja in individualnega naslavljanja. V času delovanja povezave uporabljajo območje skupinskega naslavljanja za strukturirano povezovanje.

3.7.5 LTE (Logical Tag Extended)- razširjen LT nastavitveni način

V *LT* načinu so naprave nastavljene s pomočjo etiket, ki se lokalno fizično nastavljajo.

Trenutno je LTE način omejen na uporabo pri HVAC aplikacijah, ki potrebujejo daljši niz strukturiranih podatkov. Ti podatki se izmenjavajo preko vmesniških objektov z uporabo razširjenega okvirja KNX protokola. Z izkoriščanjem razširjenega naslovnega prostora predstavljajo etikete pomembno informacijo za ustvarjanje con, kar je nujno potrebno pri modularno strukturiranih aplikacijah (npr. ogrevanje velikih poslovnih stavb).

Osnovne PT se lahko delijo med ostalimi aplikacijami in so definirane v sami KNX specifikaciji. Te PT so kot ponavadi dostopne preko skupinskih objektov in se navezujejo na temelj etiketiranih nastavitev.

Skrajšan opis: *LTE* način definira razširjene okvirje za svojo osnovno komunikacijo v času delovanja z etiketirano povezavo na vmesniške objektne lastnosti. Aplikiran profil zahteva od *LTE* naprav podporo za dopolnilen vmesnik s prosto povezano standardno komunikacijo s skupinskim naslavljanjem. Implementira standardno upravljanje za individualno naslavljanje pri prosti povezavi.

3.7.6 A-način nastavitve

Poglejmo si še nastavitveni način, ki je najbolj zanimiv za naprave, ki jih izdelujemo v Gorenju – torej za gospodinjske aparate.

Nasprotno od vseh prej opisanih načinov, ki so predani fiksni instalaciji, je A-način namenjen za naprave, ki jih lahko v omrežje poveže tudi ne-izučeni uporabnik. Torej mora ta način vključevati mehanizme, ki napravi omogočajo samostojno iskanje ustreznih povezav. Upoštevana pa je tudi možnost, da so nekatere od teh naprav 'prestavljive'.

Naprave so opremljene z lastnim sistemom za pridobivanje individualnih naslovov. Te naprave so v glavnem del aplikacije, ki ima svoj predan aplikacijski krmilnik, ponavadi je to nastavitveni oskrbnik (*Master*) za to aplikacijo.

Skrajšan opis: V A-načinu je komunikacija osnovana na standardnih okvirjih z možnostjo točka-točka komunikacije vmesniških objektov, ki jo prične aplikacijski krmilnik (pri etiketiranih povezavah), ter z možnostjo proste povezave s porazdeljeno skupinsko objektno komunikacijo.

Konfiguracija skupinskega naslova je narejena dinamično z nastavitvenim oskrbnikom z uporabo sredstev (servisov), ki pripadajo aplikacijskemu nivoju. Informacijo, ki je potrebna za samo konfiguracijo pridobi nastavitveni oskrbnik preko opisne informacije v vmesniškem objektu pri točka-točka komunikaciji.

3.8 Profili

KNX specifikacija v bistvu predstavlja neko vrsto 'orodja', kjer lahko izbiramo med različnimi lastnostmi, ki napravam omogočajo medsebojno delovanje znotraj danega nastavitvenega načina in znotraj celotnega omrežja [4].

Če želimo obdržati koherentnost sistema, lažje načrtovanje in omogočeno certificiranje naprav, je potrebno upoštevati lastnosti, ki so združene pod skupnim imenom **Profili**.

Profili obstajajo v omejenem številu, ustvarjeni pa so bili za pokrivanje potreb in navad znotraj skupine Konnex združenja. Iz tega sledi, da bo podan profil (ali kombinacija več profilov) omogočil izgradnjo naprav in sistemov, ki se enostavno integrirajo v KNX sistem.

Profili so dosegljivi za :

- Sistemski način konfiguracije (S-Mode)
- Enostaven način konfiguracije (E-Mode)
- Avtomatski način konfiguracije (A-Mode)
- In za upravljaljske odjemalce za katerikoli način

Ob času certificiranja se mora proizvajalec odločiti kateremu profilu odgovarjajo njegove naprave in glede na to, so nato izpeljani testi pri certificiranju.

3.8.1 Opis profilov

Katerikoli profil je lahko osnovan na omogočenem fizičnem prenosnem mediju. Zbirka zahtev za te medije je podana najprej v Konnex specifikaciji (*Volume 6 / 'Profiles'*).

Skupno jedro in lastnosti omrežnega managementa so odvisne od izbranega nastavitvenega načina. Profili za S-način konfiguracije naprave so lahko splošne narave in omogočajo popolno združljivost z ETS orodjem, ali pa vsebujejo tudi standardizirane lastnosti komponent, ki so dostopne v OEM.

Ti profili so predvsem obstoječe implementacije, ki so 'podedovane' od EIB, vendar pa obstajajo tudi novejši, ki zagotavljajo dodatne strojne in vgrajene programske lastnosti (poleg osnovnih), kar popestri aplikacijski dizajn. Specialni profili so potrebni za usmernike in mostiče.

Profili za E-način so samo osnovni, vendar vključujejo tudi nekaj dediščine implementacije prejšnjih sistemov: CTRL-način DMA1 in DMA2, Push button, LT, LTE in A-način.

4 LONWORKS

LonWorks tehnologija [5] je nastala pod okriljem ameriškega podjetja Echelon, ki je vodilni dobavitelj infrastrukturne strojne in programske opreme, za hitro rastoč trg omrežnih naprav.

Od leta 1988 je Echelon-ovo rešitev - LonWorks sistem - prevzelo/sprejelo že tisoče proizvajalcev naprav in sistemov. Milijoni povezljivih LonWorks naprav so bili instalirani v različne stavbe, tovarne, transportne sisteme, domove, servisna podjetja in na stotine drugih aplikacij po celem svetu.

LonWorks tehnologija omrežnih naprav omogoča končnemu uporabniku daljinsko povezovanje, nadziranje in diagnostiko inteligentnih naprav.

Danes je LonWorks omrežje de facto standard za omreženje vsakodnevnih naprav. Ta tehnologija je poznana skoraj vsem organizacijam, ki predpisujejo standarde na tem področju po svetu (vključno z AAR, ANSI, ASHRAE, IEEE, IFSF in SEMI).

Echelon dobavlja preko 90 produktov za neodvisne proizvajalce opreme (OEM) in systemske integratorje.

4.1 Kaj je LonWorks?

LonWorks tehnologija [5] ponuja rešitve za veliko problemov pri načrtovanju, gradnjah, instalacijah in vzdrževanju omrežij naprav: ta omrežja lahko številčno obsegajo vse do 32.000 naprav in so lahko uporabljana praktično povsod – od supermarketov do bencinskih črpalk, v letalih in vlakih, od laserjev do avtomatov na žetone, za enodružinske hiše ali za nebotičnike.

Danes se v skoraj vsaki industriji že izogibajo lastniško (*proprietary*) razvitih centraliziranih sistemov. Proizvajalci raje uporabljajo odprte sisteme, mikrokrmilnike 's police' ter posamezne dele, s katerimi lahko zagotovijo zanesljivost, fleksibilnost, nizko ceno in dobro delovanje.

LonWorks tehnologija podpira koncept za realizacijo sistemov z odprto komunikacijo pri gradnji oz. razvoju avtomatizacije v industriji in drugod. Je komplet orodij in komponent za izvedbo mrežno nadziranih sistemov.

Mrežno nadzorovan sistem je vsaka skupina naprav, ki delujejo v medsebojni komunikaciji z nadziranjem senzorjev, stikal ipd. in omogočajo dostop do sistemskih podatkov.

LonWorks sistem uporablja za opravljanje nalog LonTalk protokol, poznan tudi kot *'ANSI/EIA 709.1 Control Networking Standard'*.

LonWorks sistem temelji na naslednjih konceptih:

- Nadzorovani sistemi imajo veliko skupnih zahtev ne glede na aplikacijo
- Mrežno nadzorovan sistem je veliko 'močnejši', fleksibilnejši in preglednejši kot tisti, ki ne uporablja mrežne povezave
- Dolgoročno gledano, s poslovnega vidika, lahko z mrežno nadzorovanim sistemom ustvarimo in prihranimo več denarja kot s sistemom brez mrežne povezave

V nekaterih pogledih je LonWorks omrežni sistem podoben računalniškem podatkovnem omrežju, ki ga poznamo kot LAN. Podatkovno omrežje predstavlja računalnik povezan z različnimi komponentami, ki komunicirajo med sabo z uporabo standardnega TCP/IP protokola. Podatkovna omrežja so optimizirana za prenašanje velike količine podatkov in zato načrt podatkovnih omrežnih protokolov dopušča občasne zamude pri sprejemanju in oddajanju podatkov.

Mrežno nadzirani sistemi vsebujejo podobna orodja, ki so optimizirana glede na zahteve v zvezi z nadzorom (cena, delovanje, velikost in odzivni čas). Omogočajo tudi razširitev sistemov znotraj omrežja v razrede, kar pa pri podatkovnih omrežjih ni bilo mogoče.

Z vključevanjem LonWorks komponent v mrežno nadzorovane sisteme se proizvajalcem le teh občutno zmanjša čas načrtovanja in razvijanja teh produktov oz. sistemov. Rezultat tega je

tudi cenejši razvoj in doslednost, ki omogoča napravam različnih proizvajalcev medsebojno komunikacijo.

Inženirji se lahko pri načrtovanju povsem posvetijo izgradnji čim boljše aplikacije in se jim ni potrebno obremenjevati s problemom komunikacije in delovanjem sistema kot celote. S tem je omogočeno tudi sodelovanje različnih proizvajalcev pri razvoju neke aplikacije. Uporabnik lahko zato izbira med velikim številom komponent različnih proizvajalcev.

Poenostavljeno je dodajanje, prestavljanje in spreminjanje komponent v nekem sistemu ob vsakem času. Število vozlišč v nekem omrežju je praktično neomejeno.

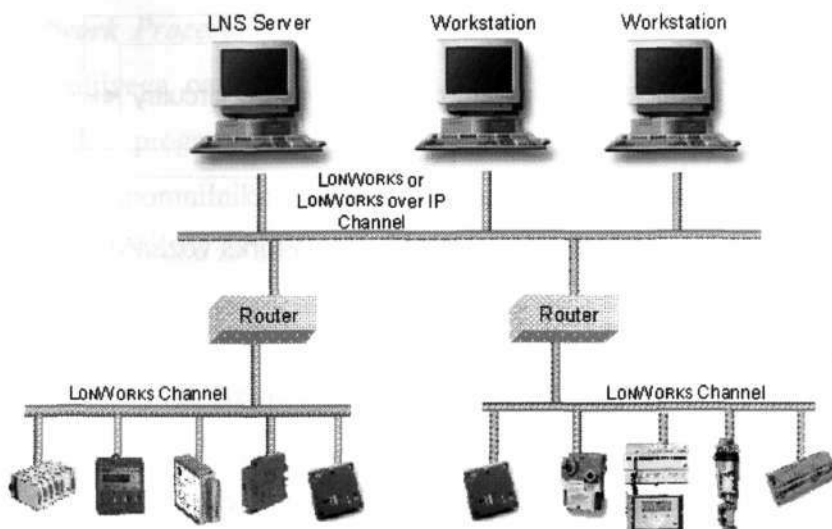
4.2 Razlika med LAN in LON

LAN – (*Local Area Networks*) so omrežja informacijske tehnologije (IT). Podatki, ki se prenašajo, so besede (tekst), grafika in drugo. Naprave v omrežju so računalniki, tiskalniki, skenerji, ipd. Učinek je definiran kot prenosna hitrost (transfer rate) in je merjen v Mbit/sek.

LON – (*Local Operating Networks*) so omrežja z nadzorom. Podatki, ki se prenašajo, so vrednost zunanje temperature, *on* ali *off* stanje stikala, lega ventila, ipd. Naprave v omrežju so senzorji, regulatorji, stikala,... Učinek je definiran kot prenos na sekundo in odzivni čas (v LonWorks omrežju je lahko prenesenih do 700 paketov na sekundo pri prenosni hitrosti podatkov 1.25Mbit/sek).

4.3 Kaj je LonWorks Control Network?

Za razumevanje dobrega načrtovanja kontroliranega omrežja [6] je potrebno najprej razumeti funkcijo odprtega omrežja (slika 4.1). Povedano enostavneje: odprto omrežje dovoljuje nekemu številu inteligentnih naprav, da neposredno komunicirajo med seboj. Torej ni potreben poseben nadzorni krmilnik, ki bi pridobival informacije iz naprav in jih nato posredoval naprej k drugim. To pomeni da je vsaka naprava sposobna objave informacije neposredno na omrežje k drugim napravam.



Slika 4.1 : Odprto kontrolno omrežje

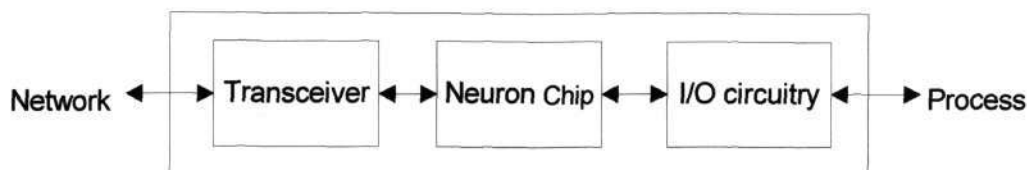
LonWorks omrežje je sestavljeno iz inteligentnih naprav, imenovanih '*nodes*' (v nadaljevanju vozli). Ti vozli so povezani med seboj za medsebojno komunikacijo. Povezani so z enim ali več komunikacijskimi mediji (TP, PWL, RF...). Komunikacija je izvedena s tako imenovanim LonTalk protokolom, ki je specifičen za *LonWorks Control Networks*.

Vozel je lahko en sam senzor ali npr. programiran termostat. Vozel realizira procesno funkcijo svoje naprave z izvrševanjem zanj napisanega programa. Funkcije vozlov so lahko zelo enostavne. Medsebojno vplivanje in vzajemno delovanje vozlov omogoča izvajanje kompleksnih nalog. Prav to je ena izmed prednosti *LonWorks* tehnologije.

4.4 *LonWorks* komponente

LonWorks tehnologija [5] vsebuje vse elemente za načrtovanje, razvoj, delovanje in vzdrževanje nadzora v omrežju: strojno opremo, programsko opremo in znanje oz. izkušnost.

Vozel je inteligentna naprava v omrežju (senzor, regulator, stikalo, aparati...). Vsi vozli skupaj tvorijo omrežje. Medsebojno so lahko povezani z različnimi, za posamezno situacijo primernimi komunikacijskimi mediji, kot so različni kabli, RF ali IR povezava, optični kabel ali električni vod. S pomočjo teh medijev vozli med seboj komunicirajo z uporabo LonTalk protokola.



Slika 4.1: Osnovna zgradba LonWorks vozlišča

Sestavni deli vozla so (slika 4.1):

Neuron Chip – nevronski čip.

Transceiver – oddajnik/sprejemnik je naprava, ki realizira elektronsko in mehansko povezavo med nevronskim čipom in komunikacijskim medijem. (obstajajo pa že variante, ki vsebujejo neuron čip in transceiver v enem ohišju – PL3120 in PL3150)

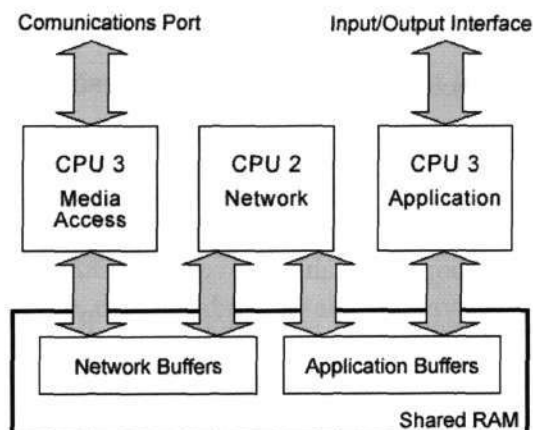
I/O circuitry – aplikacija specifična za vhodno/izhodno elektroniko.

Network Interfaces – mrežni vmesniki– povezujejo računalnik z LonWorks omrežjem.

4.4.1 Neuron čip

Vsak *Neuron* čip ima tri procesorje (slika 4.2) [5]. Dva od teh sodelujeta s komunikacijskim podsistemom in tako omogočata prenos informacij od vozla do vozla. Tretji procesor pa je zadolžen za izvajanje aplikacijske kode.

Procesor 1 je **Media Access Control (MAC)** procesor. Upravlja s prvim in drugim nivojem v OSI referenčnem modelu.



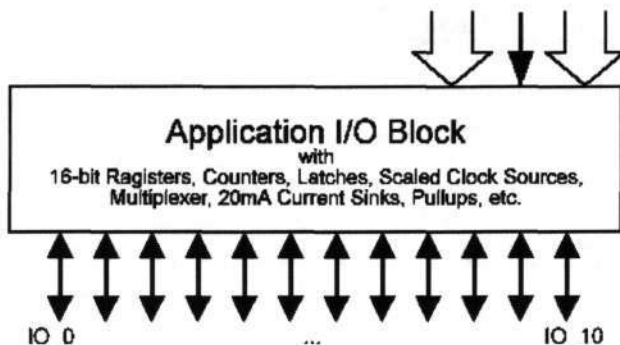
Slika 4.2: CPU nevronskega čipa

Procesor 2 je **Network Processor**. Odgovoren je za nivoje od tri do šest. Njegovo delo je upravljanje spremenljivega omrežnega procesa, naslavljanje, proces prenosa, ugotavljanje pristnosti, diagnostika, programski časovniki za upravljanje mreže in rutinske funkcije. Uporablja omrežne medpomnilnike v skupnem pomnilniku za komunikacijo s procesorjem 1, in aplikacijske medpomnilnike za komunikacijo s procesorjem 3.

Procesor 3 je tako imenovan **Application Processor**. Izvršuje aplikacijsko kodo s pomočjo operacijskega sistema. Aplikacijsko kodo napiše uporabnik.

Procesni vmesnik

Vhodno/izhodni (I/O) vmesnik ima integrirano strojno in programsko opremo (firmware) za povezavo s strojno opremo specifično za aplikacijo: ventili, stikala, releji, razni mikrokrmilniki, modemi, itd. (slika 4.3).



Slika 4.3: Procesni vmesnik Neuron čipa

Ti priključki so lahko oblikovani na veliko načinov, da zagotovijo fleksibilne (nastavljive) vhodno/izhodne funkcije.

Programski jezik Neuron C omogoča programerju, da določi enega ali več priključkov kot I/O razrede (slika 4.4). Ti potem preprosto predstavljajo vhod, izhod ali napisano firmware rutino v ROM pomnilniku, ki so dosegljivi uporabnikovemu aplikacijskemu programu. I/O razred s pomočjo vmesnika upravlja napravo preko enega ali več priključkov za Neuron C aplikacijo.

I/O Mode		I/O Pin	0	1	2	3	4	5	6	7	8	9	10
Direct	Bit Input, Bit Output												
	Byte Input, Byte Output		All Pins										
	Leveldetect Input												
	Nibble Input, Nibble Output		Any Four Adjacent Pins										
	Touch I/O												
Parallel	Muxbus I/O		Data Pins 0-7								ALS	WS	RS
	Parallel: Master/Slave A		Data Pins 0-7								CS	R/W	HS
	" Slave B		Data Pins 0-7								CS	R/W	AO
Serial	Magcard Input		Optional Timeout								C	D	
	Magtrack I Input		Optional Timeout								C	D	
	Bitshift Input, Bitshift Output		C	D	C	D	C	D	C	D	C	D	C
	Neurowire: Master		Optional Chip Select								C	D	D
	" Slave		Optional Timeout								C	D	D
	I ² C I/O										C	D	
	Serial Input												
	Serial Output												
	Wiegand Input		01		01		01		01				
Timer/Counter	Dualslope Input	Control											
	Edgelog Input												
	Infrared Input												
	Overtime Input												
	Period Input												
	Pulsecount Input												
	Quadrature Input						4+5		6+7				
	Totalcount Input												
Timer/Counter	Edgedivide Output	Output											
	Frequency Output												
	Oneshot Output												
	Pulsecount Output												
	Pulsewith Output												
	Triac Output	Control											
	Triggeredcount Output	Control											
			High Sink				Pull Ups				Standard		

Slika 4.4: I/O enote Neuron čipa

Priključki od IO_4 do IO_7 imajo programabilne nivoje, ki so omogočeni ali onemogočeni z enostavnim ukazom C-prevajalnika:

```
#pragma enable_io_pullups;
```

Priključki od IO_0 do IO_3 imajo visoko tokovno zmogljivost (20mA; 0,8V), ostali pa standardno tokovno zmogljivost (1.4mA; 0.4V).

Vsi priključki (IO_0 do IO_10) so TTL kompatibilni. Priključki od 0 do 7 pa imajo še visoko vhodno upornost (torej imajo visoko občutljivost).

Komunikacijska vrata imajo 5 priključkov, ki so lahko oblikovani za povezave z različnimi transceiverji.

Deluje lahko na naslednje tri načine:

1. **Single_ended Direct Mode** je najbolj običajen način. Deluje z zunanjim aktivnim sprejemnikom/oddajnikom (v nadaljevanju transceiver). Uporaben je za komunikacijo z mediji kot so RF, IR, optičnimi vlakni in koaksialnimi kabli z uporabo kodiranja tipa Manchester.
2. **Differential Direct Mode** deluje z zunanjimi pasivnimi komponentami, z uporabo na nevronskega čipa vgrajenega transceiverja. Ta namreč prilagodi električne nivoje na prepleteno parico z uporabo diferencialnega kodiranja tipa Manchester.
3. **Special Purpose Mode** uporablja zunanje inteligentne transceiverje. Ti sprejemajo podatke iz nevronskega čipa (v kodiranem formatu), zatem izvedejo formatiranje in sinhronizacijo teh podatkov, da jih prilagodijo za prenos v omrežje. Na drugi strani sprejemajo podatke preko omrežja in jih v obratni smeri prilagodijo za prenos do nevronskega čipa. Ko je nevronskega čip nastavljen za delovanje v *Special Purpose Mode*, izkorišča za prenos podatkov navadni električni vod. Lahko pa vzajemno deluje z napajalno sprejemno-oddajno enoto.

Servisni priključek (Service pin) je specifičen vhod (izhod) nevronskega čipa. Uporabljen je med instalacijo, konfiguracijo in vzdrževanjem vozlišča v omrežju.

Ozemljitev tega priključka (s pritiskom na gumb) povzroči, da *firmware* na nevronskega čipu pošlje 48-bitno (ID) identifikacijsko informacijo v omrežje. To informacijo potem omrežno orodje uporabi za instalacijo vozlišča.

Za kontrolo delovanja servisnega priključka obstaja tudi LED dioda (ki ni obvezna, je pa priporočljiva), ker lahko z njeno pomočjo hitro ugotovimo stanje LON komponente.

4.4.2 PL3120 in PL3150 Power Line Smart Transceiver

Najbolj zanimiv med *Neuron* mikroprocesorji [7] je trenutno PL31xx, ki ga je razvilo podjetje Echelon. S tem mikroprocesorjem je postala uporaba električnega omrežja kot komunikacijskega medija še bolj smiselna izbira.

Ta mikroprocesor vsebuje jedro *Neuron* procesorja vključno s 'power line' transceiverjem, vse skupaj v zelo majhnem ohišju. To naredi ta mikroprocesor idealen za uporabo pri aparatih, avdio/video napravah, razsvetljavi, za ogrevanje/hlajenje, varnostne sisteme, skratka vse možne naprave, ki za komunikacijski medij uporabljajo električno napajalno omrežje.

Ta mikroprocesor združuje precej tehničnih inovacij, ki zagotavljajo zanesljivo delovanje:

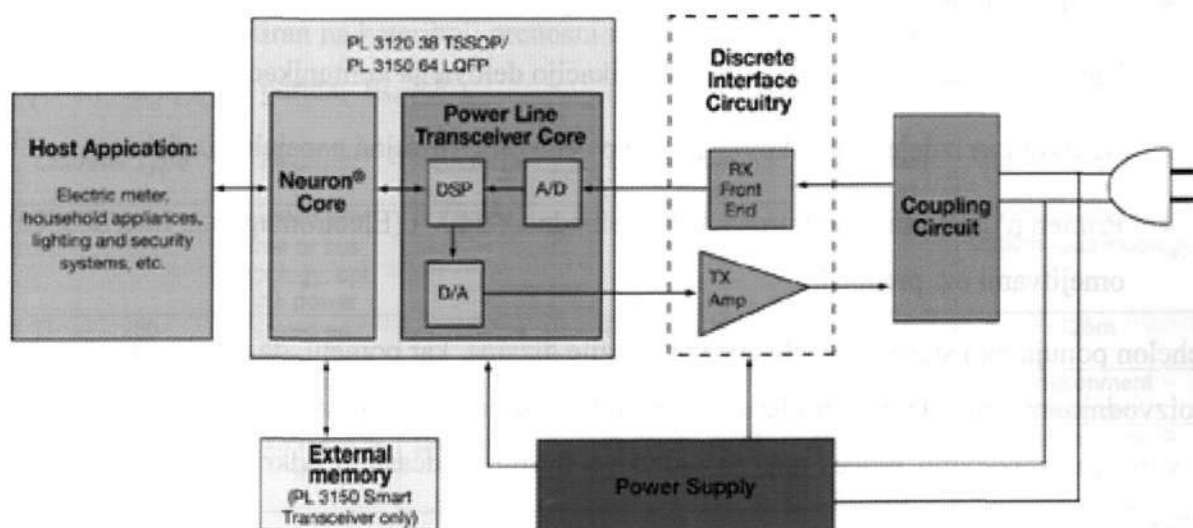
- **Ozkopasovna tehnologija z digitalnim signalnim procesiranjem:** koristno je uporabljeno dovršeno digitalno signalno procesno jedro, ki uporablja patentirano zmanjševanje šuma in algoritem za popravljanje popačenosti. Te lastnosti omogočajo transceiverju odpravljanje velike širine motenj na električnem omrežju, vključno z impulznimi šumi, ponavljajočimi šumi in faznimi popačenji. To omogoča napravam s tem procesorjem, da uspešno sprejemajo signale tudi tam, kjer jih druge rešitve niti ne zaznajo.
- **Dvojna nosilna frekvenca:** edinstven sistem s funkcijo dvojne nosilne frekvence avtomatično izbere alternativno sekundarno komunikacijsko frekvenco, v primeru onemogočene uporabe primarne frekvence zaradi šuma.
- **Vnaprejšnja korekcija napak:** Na električnem omrežju je veliko izvorov različnih шумov, ki močno vplivajo na poslabšanje prenosa podatkovnih paketov. PL31xx uporablja zelo učinkovit algoritem za odkrivanje napak (*Forward error Correction - FEC*), kot dodatek k cikličnem preverjanju odvečnosti (*cyclic redundancy check (CRC)*) – vse skupaj preprečuje napake pri prenosu paketov.
- **Močen izhodni ojačevalec:** zunanja izhodna ojačevalna tehnologija uporabljena v kombinaciji s pametnim transceiverjem omogoča amplitudo signala 7 Vpp, istočasno pa ustreza vsem svetovnim zahtevam glede oddajanja.
- **Široko dinamično območje:** Dinamično območje je odvisno od občutljivosti sprejemnika. To območje je pri PL31xx je > 80dB.

Ta mikroprocesor je skladen z vsemi potrebnimi svetovnimi standardi na tem področju:

- FCC,
- Industry Canada,
- Japan MPT,
- in European CENELEC EN50065-1 regulations

Torej ga lahko uporabljamo v aplikacijah povsod po svetu. CENELEC komunikacijski protokol je avtomatično upravljan s temi pametnimi transceiverji. Zaradi tega se uporabnikom ni potrebno ukvarjati s kompleksnim razvojem časovnih algoritmov in algoritmov za dostop do omrežij, ki so predpisani po standardu CENELEC EN50065-1. Ti pametni transceiverji tako delujejo v CENELEC okvirjih (A-Band) ali v splošnih signalnih območjih (C-Band), brez potrebe po skladiščenju več različnih delov za različne aplikacije.

Pri izdelavi kompletne naprave osnovane na PL31xx (slika 4.5) je potrebno zelo majhno število zunanjih komponent. Podjetje Echelon ponuja komplet razvojnega orodja (*Development Support Kit (DSK)*), s katerim lahko stranke dokaj enostavno implementirajo tak vmesnik.



Slika 4.5: Blok diagram tipične PL 31xx 'Smart transceiver' naprave

4.4.3 Dokazana tehnologija in razvojni viri

Osnovna tehnologija [6], ki je uporabljena pri pametnih transceiver-jih je bila razvita in optimizirana skozi leta praktičnih testiranj v aplikacijah po celem svetu. Uporabljenih je bilo več kot 20 milijonov ozkopasovnih transceiver-jev podjetja Echelon v širokem uporabniškem območju (uporabniške, servisne, stavbne, industrijske in transportne aplikacije).

Echelon si je z leti dela in podpore svojim strankam, pridobilo veliko izkušenj na tem področju. Predvsem na področju komunikacije preko električnega omrežja za različne aplikacije.

Obstaja široka paleta tehnične dokumentacije, diagnostičnih orodij, podpornih programov in izobraževalnih tečajev, ki omogočajo podporo strankam in uporabnikom pri njihovih projektih.

Široka knjižnica podporne dokumentacije vsebuje:

- Priporočeno arhitekturo za uporabniške, servisne, transportne aplikacije in aplikacije industrijske/hišne avtomatizacije.
- Shema sklopnega vezja za različne aplikacije
- Navodila za takojšnje testiranje in verifikacijo delovanja komunikacije
- Navodila za izdelavo učinkovitega nizko cenovnega dizajna napajalnega dela
- Primeren dizajn in testne procedure za skladnost z EMI (Electromagnetic Interference) omejitvami oz. predpisi.

Echelon ponuja tudi storitev pregleda oz. recenzije dizajna, kar pomeni, da je lahko pred samo proizvodnjo naprave le ta ocenjena s strani kompetentnega inženirja, kar lahko pospeši razvojni in proizvodnji proces nekega izdelka s hitrim predčasnim odkrivanjem morebitnih napak.

4.5 LonTalk protokol

Naprave v LonWorks omrežju komunicirajo med seboj z uporabo LonTalk protokola [8] – standardiziranega jezika omrežja. LonTalk je sestavljen iz serije osnovnih protokolov, ki omogočajo inteligentno komuniciranje med različnimi napravami v omrežju.

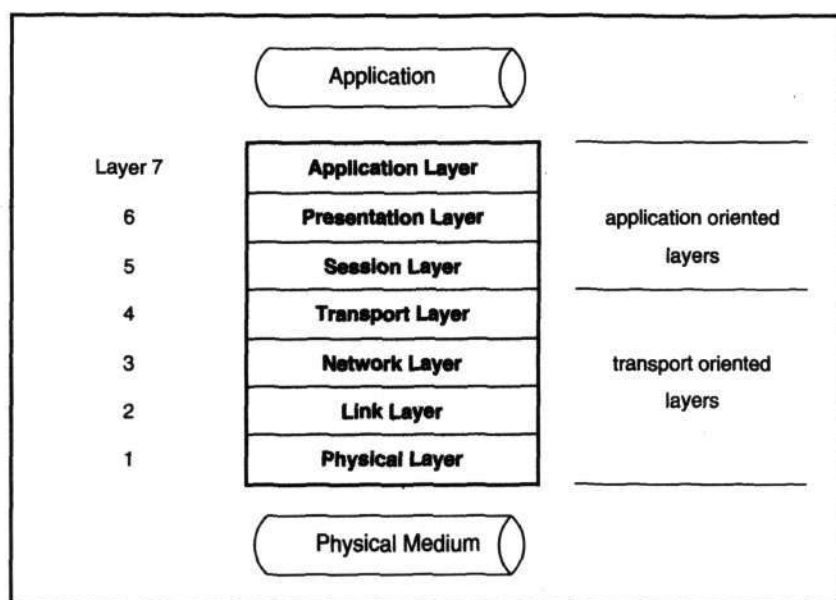
Protokol predpisuje niz servisov, ki aplikacijskem programu v napravi omogočajo pošiljanje in sprejemanje sporočil drugih naprav v omrežju brez informacij o topologiji omrežja, o imenih, naslovih ali funkcijah drugih naprav. LonWorks protokol lahko opsijsko omogoča tudi potrditve sporočil od 'konca do konca', avtentikacijo sporočil in prioriteto dostavljanje sporočil za zagotavljanje omejenega prenosnega časa. Podpora za servise omrežnega upravljanja dovoljuje oddaljenemu orodju za omrežno upravljanje sodelovanje z napravami po celotnem omrežju, vključno z možnostjo ponovne konfiguracije omrežnih naslovov in parametrov, prenašanje aplikacijskih programov, poročanje o morebitnih težavah na omrežju ter start/stop/reset aplikacije v napravi.

LonTalk protokol je del *firmware*-a na nevronskega čipu. S pomočjo ustreznih transceiverjev je lahko implementiran na katerikoli prenosni medij (Tabela 4.1)(električno omrežje (~220V), optični kabli, zvita parica, IR, RF).

Channel Type	Medium	Bit Rate	Compatible Transceivers	Maximum Devices	Maximum Distance
TP/FT-10	Twisted pair, free or bus topology, opt. link power	78kbps	FTT-10, FTT-10A, LPT-10	64-128	500m (free topology) 2200m (bus topology)
TP/XF-1250	Twisted pair, bus topology	1.25Mbps	TPT/XF-1250	64	125m
PL-20	Power line	5.4kbps	PLT-20, PLT-21, PLT-22	Environment Dependent	Environment Dependent
IP-10	LonWorks over IP	Determined by IP network	Determined by IP network	Determined by IP network	Determined by IP network

Tabela 4.1: Tipi prenosnih kanalov (medijev)

Protokol sledi referenčnemu modelu za medsebojno povezanost odprtega sistema (OSI) in oskrbuje opravila na vseh 7-ih nivojih tega referenčnega modela (Slika 4.6):



Slika 4.6: OSI referenčni model

Nivo 1: FIZIČNI NIVO - definira prenos čistih binarnih podatkov po komunikacijskem kanalu. Zagotavlja, da bo en poslan bit na izvorni strani natanko en sprejet bit na ciljni sprejemni strani. LonWorks protokol je neodvisen od uporabljenega komunikacijskega medija, tako je podprtih več protokolov fizičnega nivoja odvisno od komunikacijskega medija.

Nivo 2: POVEZOVALNI NIVO - definira metode za dostop do medija in podatkovno kodiranje za zagotavljanje učinkovite rabe samega komunikacijskega kanala. Biti fizičnega nivoja so razdeljeni na podatkovne okvirje. Ta nivo definira kdaj lahko izvorna naprava pošilja podatkovni okvir in definira kako je ta okvir sprejet na strani sprejemne naprave in zaznava morebitne napake pri prenosu. Definiran je tudi mehanizem prioritete za zagotavljanje prenosa pomembnejših sporočil.

Nivo 3: OMREŽNI NIVO – definira kako so paketi podatkov usmerjeni od izvorne do ene ali več sprejemnih naprav. Definira tudi imenovanje in naslavljanje naprav za zagotavljanje pravilnega prenosa podatkov.

Nivo 4: TRANSPORTNI NIVO – zagotavlja zanesljivo dostavo paketov sporočil. Ti se lahko izmenjujejo z uporabo potrjevalnega servisa, kjer izvorna naprava čaka na potrditev od sprejemne naprave in ponovno pošlje sporočilo, če potrditve ni sprejela. Transportni nivo

definira tudi na kakšen način se odkrivajo in izločajo podvojena sporočila (v primeru če se izgubi potrditveno sporočilo)

Nivo 5: SEJNI NIVO – Dopolnjuje nadzor nad izmenjavo podatkov na nižjih nivojih. Podpira oddaljene aktivnosti na način, da lahko odjemalec (*client*) poda zahtevo oddaljenemu strežniku in nato sprejme odgovor na to zahtevo. Definira tudi protokol avtentikacije, ki omogoča sprejemnikom sporočil, da ugotovijo ali je bil pošiljatelj avtoriziran za pošiljanje sporočila.

Nivo 6: PREDSTAVITVENI NIVO – doda strukturo k izmenjavi podatkov na nižjih nivojih z definiranjem kodiranja podatkov v sporočilu. Podatki so lahko kodirani kot omrežne spremenljivke, aplikacijska sporočila ali zunanji okvirji. Interoperabilno kodiranje omrežnih spremenljivk je zagotovljeno s standardnimi omrežnimi tipi spremenljivk (*Standard Network Variable Type* - SNVT).

Nivo 7: APLIKACIJSKI NIVO – doda aplikacijsko združljivost k izmenjavi podatkov na nižjih nivojih. Standardni objekti podpirajo interoperabilnost z jamčenjem, da aplikacija uporablja skupno semantično interpretacijo pri izmenjavi podatkov. Ta nivo definira tudi protokol za prenos datotek, ki je uporabljen za prenosni tok podatkov med aplikacijami.

4.6 Interoperabilnost – medsebojno delovanje

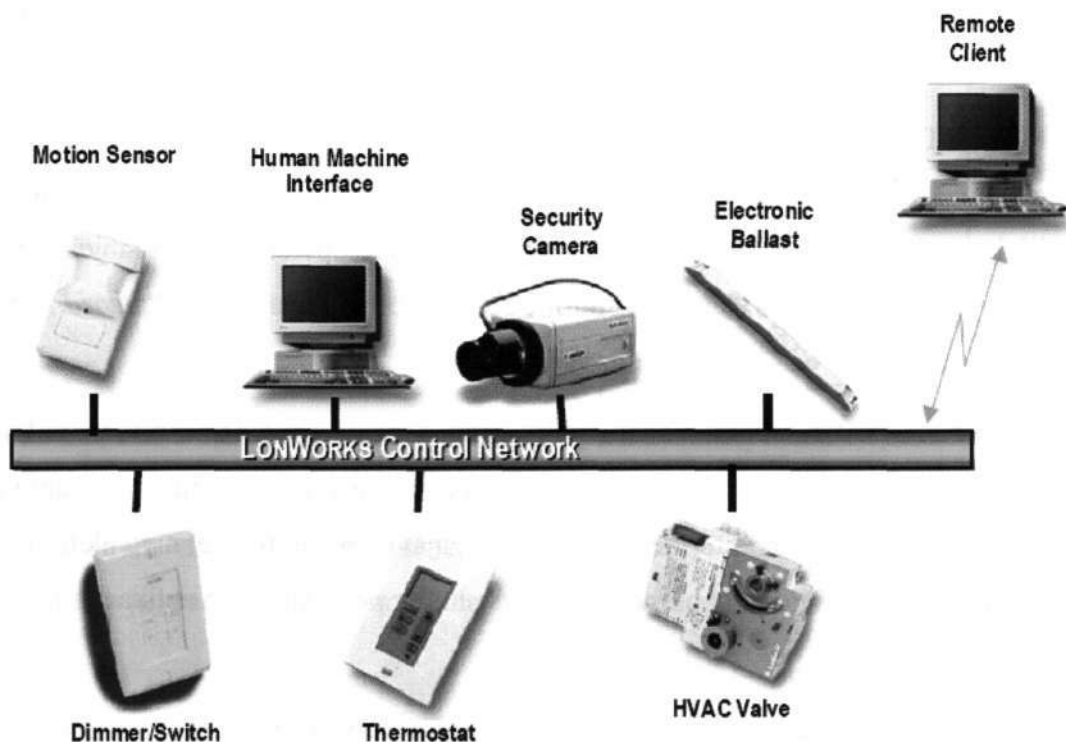
LonWorks sistem omogoča razvoj interoperabilnih naprav [5] in sistemov v pravem pomenu besede. LonWorks komponente, ki jih vključuje LonWorks sistem, so neodvisne od komunikacijskega medija in ne predpisujejo, kako naj bodo strukturirani aplikacijski programi. Vendar pa sama uporaba LonWorks komponent še ne zagotavlja interoperabilnosti LonWorks naprav različnih proizvajalcev v enem sistemu.

Res je, da so LonWorks komponente široko uporabljane v lastniških sistemih kot so nadzorni sistemi vozil, transportni sistemi in centralni telefonski nadzorni sistemi. Takšni sistemi ne uporabljajo celotne funkcionalnosti in koristi LonWorks sistema.

Med zbirko interoperabilnih (medsebojno delujočih) naprav in odprtim sistemom (slika 4.7) je pomembna razlika: nemogoče je imeti odprt sistem brez interoperabilnih naprav, povsem mogoče pa je postaviti zaprt sistem z zbirko takšnih naprav.

Z drugimi besedami povedano – interoperabilne naprave so nujen, vendar ne zadosten pogoj za postavitev odprtega sistema. Pravilno omrežno načrtovanje je dodatna zahteva za implementacijo resničnega odprtega sistema.

Za pospeševanje interoperabilnosti in razvoj naprav, ki to podpirajo, je bila ustanovljena LonMark organizacija.



Slika 4.7: Odprt sistem

4.7 LonMark organizacija

Ker za interoperabilne produkte obstaja ogromno priložnosti v različnih industrijah, je bilo leta 1994 ustanovljeno LonMark združenje za interoperabilnost. Ustanovitelji so bili podjetje Echelon in skupina LonWorks uporabnikov (skupaj 36 podjetij), ki so bili predani izdelavi interoperabilnih produktov [9].

Interoperabilnost pomeni, da so različne naprave istega ali različnih proizvajalcev integrirane v eno samo omrežje brez zahtev po posebej prilagojenih vozliščih ali posebnem programiranju.

LonMark združenje je predano razvoju standardov za omogočanje interoperabilnosti, certificiranje produktov glede na te standarde in promocijo koristi takšnih sistemov.

Združenje zagotavlja interoperabilnost na nivoju naprav. LonWorks naprave, ki so bile certificirane pri LonMark organizaciji – imenovane so LonMark naprave – lahko nosijo LonMark oznako (logotip).

Vodstvo organizacije predstavlja industrijski koncil, sestavljen iz članov, ki zastopajo vse zainteresirane skupnosti. Članstvo v tej organizaciji je odprto za vsa zainteresirana podjetja, organizacije ali individualne uporabnike, ki razvijajo, proizvajajo ali uporabljajo LonMark certificirane produkte.

Združenje razvija tehnične produktne specifikacije in navodila, ki zagotavljajo medsebojno delovanje produktov, ki bodo razviti z upoštevanjem teh pravil. Prav tako razvija in objavlja funkcionalne profile, ki natančno opisujejo vmesnik na aplikacijskem nivoju, vključno z omrežnimi spremenljivkami, nastavitvenimi lastnostmi, ter privzetimi in ostalimi (pri zagonu) odzivi, zahtevanimi za specifične, splošno uporabljane nadzorne funkcije.

LonMark združenje se osredotoča na dve področji:

1. Specifikacija standardnih transceiverjev in pripadajočih fizičnih kanalov.
2. Definicija standardov za strukturo in dokumentiranje aplikacijskih programov naprav.

4.7.1 Standardizacija fizičnih kanalov in transceiverjev

Ti standardi so vsebovani v '*LONMARK Layers 1-6 Interoperability Guidelines*'. Ta dokument prikazuje vse standardne fizične kanale, za katere so certificirani odgovarjajoči transceiver-ji. Opisuje tudi navodila za uporabo LonWorks protokola – med-pomnilniški prostor, števc, tipi, vnosi v naslovne tabele...itd.

Tipi kanalov, ki so najpogostejše uporabljani v komercialnih in industrijskih aplikacijah, so TP/FT-10 (*twisted pair*) zvita parica proste topologije z 78 kbps) in TP/XF-1250 (zvita parica, s topologijo omrežja za 1.25 Mbps).

V stanovanjskih/hišnih aplikacijah je najpogostejše uporabljan PL-20 kanal (električno omrežje s 5.4 kbps). Občasno je ta medij uporabljan tudi za ojačitev obstoječega napajalnega ožičenja kot prenosnega medija v komercialnih in industrijskih aplikacijah.

4.7.2 Standardizacija aplikacijskih programov

Ti standardi so vsebovani v '*LONMARK Application Layer Interoperability Guideline*'. Ta navodila temeljijo na funkcionalnih profilih, ki so implementirani kot LonMark objekti na posameznih napravah. Vmesniki do aplikacij so definirani kot eden ali več LonMark objektov. Vsak objekt izvaja svojo (natančno dokumentirano) funkcijo in komunicira z drugimi LonMark objekti na isti ali drugi napravi glede na natančno definirano specifikacijo vhodno-izhodnega vmesnika.

Ko je ustvarjena celotna zbirka LonMark objektov, se je potrebno lotiti naslednjega koraka pri načrtovanju in ustvarjanju omrežja. Iz zbirke je potrebno izbrati ustrezne LonMark objekte in jih povezati med seboj. Ti so definirani kot niz enega ali več vhodnih/izhodnih omrežnih spremenljivk. Te spremenljivke imajo svoje semantične definicije, ki so povezane z obnašanjem teh objektov glede na vrednosti spremenljivk, ter na nastavitvenih lastnosti za LonMark objekt.

Zaradi možnosti razširitve v prihodnosti in zaradi omogočanja razlikovanja med različnimi proizvajalci, so definicije LonMark objektov sestavljene iz obveznih, opsijskih, ter za proizvajalca specifičnih omrežnih spremenljivk in nastavitvenih lastnosti.

Zaradi zagotavljanja interoperabilnosti, obnašanje LonMark naprave ne more biti odvisno od vmesnikov specifičnih za posameznega proizvajalca.

4.7.3 Standardni tipi omrežnih spremenljivk (SNVT)

Da lahko aplikacije različnih proizvajalcev enostavno sodelujejo med seboj z uporabo omrežnih spremenljivk, morajo imeti podatki znotraj teh enak pomen za vse strani. Na primer: vse vrednosti temperature se morajo prenašati preko medija v skupnem formatu, ki je lahko stopinje Kelvina, Celzija ali Fahrenhaita – za pravo interoperabilnost se lahko uporablja le ena od teh.

Ta proces določanja je pospešen s strani LonMark združenja, ki je že definiralo in objavilo preko sto skupnih sistemskih spremenljivk, ki jih imenujemo SNVTs = *Standard Network Variable Types* (včasih se uporablja izgovorjava: "snivets"). Najnovejša lista teh spremenljivk je objavljena na straneh www.LonMark.org.

Uporaba teh SNVTs ne določa, na kakšen način bo sprejeti podatek prikazan končnemu uporabniku omrežja. Kljub temu, da je npr. temperatura poslana kot stopinje Celzija, je lahko nato enostavno prikazana v drugi enoti, recimo stopinjah Kelvina pod nadzorom orodja, ki ga uporablja uporabnik omrežja.

4.7.4 Nastavitvene lastnosti

Večina LonMark objektov zahteva prilagoditev glede na specifično sistemsko aplikacijo. LonMark navodila specificirajo podatkovno strukturo imenovano *nastavitvene lastnosti*, ki zagotavljajo standarde za dokumentacijo in za formate omrežnih sporočil, ki so uporabljani za prenos prilagoditvenih podatkov za napravo s pomočjo omrežnih orodij.

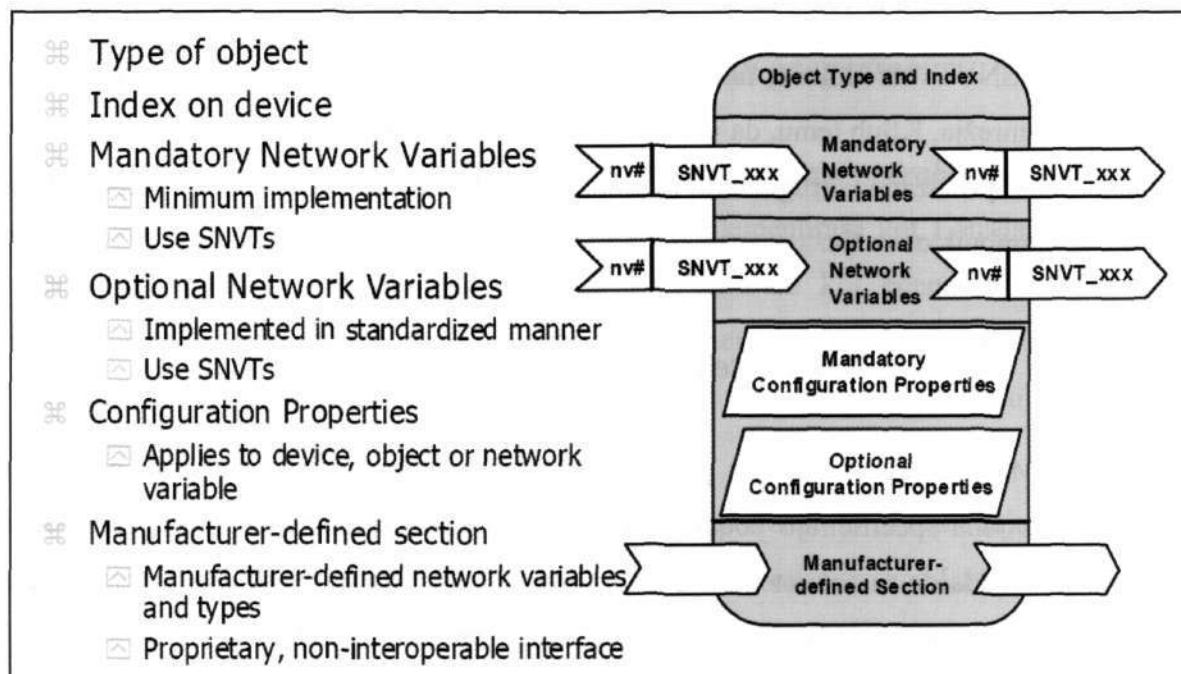
LonMark združenje definira standardno zbirko nastavitvenih lastnosti, ki so imenovane SCPTs = *Standard Configuration Property Types* (včasih se uporablja izgovorjava: "skip-its").

Prav tako lahko posamezni proizvajalci definirajo svojo lastno zbirko SCPTs, te so potem imenovane UCPTs = *User-defined Configuration Property Types* ("you-keep-its").

4.7.5 LONMARK objekti in funkcionalni profili

Funkcionalni profili so lahko splošni, kot je na primer enostaven '*Open Loop Sensor*' objekt, lahko pa so oblikovani za specifično aplikacijsko uporabo, kot je recimo HVAC ali sistemi razsvetljave.

LonMark združenje oblikuje skupine po interesnih članih za opravljanje posameznih nalog kot so: načrtovanje, odobritve in objava funkcionalnih profilov v različnih funkcionalnih področjih (HVAC, varnost, razsvetljava, ...).



Slika 4.8: Funkcionalni profil

Funkcionalni profil (slika 4.8) podrobno opisuje vmesnik aplikacijskega nivoja, vključno z omrežnimi spremenljivkami, nastavitvenimi lastnostmi in privzetimi opisi delovanja pri zagonu, ki so zahtevani na LonMark napravi za specifične krmilne funkcije in tiste, ki so v splošni rabi. Profili standardizirajo funkcije in ne produkte! Torej dajejo industrijskim skupinam univerzalno predlogo za dokument, v katerem je opisano funkcionalno obnašanje naprav s skupnimi (razumljivimi) enotami.

Ta predloga olajša oz. poenostavi postopek specifikacije in poveča interoperabilnost brez ogrožanja unikatnih možnosti ali lastnosti naprave, ki jih želi posamezen proizvajalec uporabiti, da se lahko njegovi produkti razlikujejo od konkurenčnih. Produkt je lahko osnovan na enem ali več funkcionalnih profilih.

Aplikacijski program je v LonMark napravi potemtakem sestavljen iz enega ali več LonMark objektov, ki so osnovani na definiciji funkcionalnega profila, ter nastavljeni in uporabljeni neodvisno drug od drugega. Objekti so lahko povezani s katerikoli drugim objektom na

omrežju za zagotavljanje želene systemske funkcionalnosti. Večina LonMark naprav vsebuje tudi tako imenovan '*node object*', ki dovoljuje nadzor lastnega statusa, ter statusa drugih objektov na omrežju s primernim omrežnim orodjem.

Vse LonMark naprave morajo biti '*samo-dokumentirane*'. S tem je omogočeno, da lahko vsako LNS orodje z omrežja pridobi vse potrebne informacije za povezavo katerekoli LonMark naprave v sistem (ter za njeno konfiguracijo in upravljanje).

Vsaka LonMark naprava mora imeti tudi *zunanjo vmesniško datoteko*. To je specialno formirana tekstovna datoteka s končnico *.XIF (External Interface File)*, ki omogoča omrežnim orodjem načrtovanje in konfiguracijo omrežne podatkovne baze že pred samo fizično povezavo naprave, ki jo lahko nato enostavno pripišejo (*commission*) v omrežje po sami instalaciji.

Na spletni strani LonMark združenja obstaja podatkovna baza teh zunanjih referenčnih datotek za vse LonMark certificirane naprave.

4.8 LNS programsko orodje

LNS (*LonWorks Network Service*) za Windows okolje [5].

Omrežni operacijski sistem mora zagotavljati skupne omrežne funkcije za podporo nadzorovanja, za instalacijo in za oblikovanje omrežja.

Z LNS operacijskim sistemom lahko več uporabnikov (s svojim orodjem, ki je lahko npr. NSI) istočasno dela na istem omrežju oz. sistemu popolnoma brez konfliktov. To je mogoče, ker vsako orodje z vidika omrežja predstavlja oddaljenega odjemalca. Tako se lahko npr. serviser omrežja s svojim orodjem kjerkoli na mreži priključi vanjo in opravi svoje delo.

NSI (*Network Services Interface*) je cenovno dostopna strojna oprema, ki omogoča enostaven dostop do podatkov, ki so rezultat aplikacijskih in LNS strežnikov.

Ker odjemalsko orodje predhodno ne potrebuje omrežne podatkovne baze, lahko zanj uporabimo vse, od precej dragega prenosnega osebne računalnika pa do cenених 'žepnih' izvedb (s cenеним mikrokrmilnikom in preprostim LCD prikazovalnikom), pač v odvisnosti od spretnosti in uporabnika.

Za Windows okolje obstaja posebna vrsta LNS: Plug-ins (standardna 32-bitna Windows izvedba LNS operacijskega sistema). LNS v Windows okolju izdelava globalno podatkovno bazo, do katere je omogočen dostop s pomočjo LCA OCX mehanizma do LON komponent oz. spremenljivk.

LNS omogoča tudi nadzor, vzdrževanje in kontrolo omrežja.

5 CECED



CECED (*European Committee of Domestic Equipment Manufacturers*) [11] organizacija je bila ustanovljena leta 1958 na pobudo zahodnoevropskih nacionalnih združenj (*Western-European National Associations*).

Zaradi lažjega in boljšega komuniciranja z evropskimi institucijami je bila leta 1997 postavljena stalna pisarna združenja v Bruslju. Prav tako je bilo od tega leta naprej omogočeno tudi včlanjevanje individualnih podjetij.

CECED združenje predstavlja Evropsko industrijo gospodinjskih naprav. Ta industrija zaposluje 200.000 ljudi in ima letni promet okoli 35 milijard evrov.

Člani tega združenja proizvajajo naslednje produkte:

- Velike gospodinjske aparate (hladilniki, zamrzovalniki, pomivalni stroji, pralni stroji, sušilniki perila, pečice in kuhalniki,...)
- Majhne gospodinjske naprave (od brivskih aparatov do sesalnikov)
- Oprema za ogrevanje, prezračevanje, klimatizacijo,...

V zadnjih letih je industrija izboljšala tehnologijo do te mere, da je to povzročilo večjo proizvodnjo in nižje cene za končne uporabnike. Na leto je v Evropi v povprečju prodanih 50 milijonov velikih in 200 milijonov majhnih gospodinjskih aparatov.

Vse te naprave so torej dnevno prisotne v naših življenjih, kar posledično tudi povišuje kvaliteto življenja.

Glavni smisel CECED organizacije je zastopanje in zaščita interesov evropske industrije. Združenje deluje kot partner pri dialogu s političnimi in ostalimi institucijami EU, ter hkrati tudi sodeluje z mediji, novinarji ter ne-vladnimi organizacijami, ki predstavljajo različne interese (potrošniške, okolje-varstvene organizacije...), ter ostalimi interesnimi skupinami.

CECED pomaga industriji pri doseganju konsenzov na različnih skupnih interesnih področjih ali vprašanjih, kot so recimo produktna/procesna standardizacija in iniciativa za postavljanje pravil.

Osnovni princip dela CECED-a je združevanje vseh primernih strank (oz. strani) z namenom skrajšanja procesa odločanja. Za sprejemanje posamezne odločitve znotraj CECED organizacije so povabljene tiste organizacije oz. podjetja, na katera ima ta odločitev lahko vpliv. Tako ostanejo delovne skupine majhne in učinkovite.

Delovne skupine analizirajo specifične teme povezane s tehniko ali trgov, ter pripravijo mnenje za upravni odbor, tehnični odbor ali za enega od produktnih oddelkov.

Obstoječe delovne skupine so za naslednja področja:

- Dostopnost / uporabnost
- *Convergence Group* & Standardizacija (sem spada povezljivost naprav)
- *E-Commerce*; PI – produktne informacije
- Energija
- Mednarodna standardizacija
- IRIS (*International Repair System*)
- Razsežnost (kemikalije)
- Standardizacija varnosti
- Statistične in tržne raziskave
- Trgovski sejmi

In tehnične delovne skupine po produktih:

- Klimatizacija
- Kuhanje (pečice, štedilniki, kuhalniki)
- Hlajenje (*Cold*) (hladilniki, zamrzovalniki)
- Majhni gospodinjski aparati (sesalci, kavni avtomati, brivniki, likalniki,...)
- Grelniki vode
- Pranje (*Wet*) (pralni stroji, sušilniki, pomivalni stroji)

CECED '*Convergence*' delovna skupina, sestavljena iz glavnih Evropskih proizvajalcev gospodinjskih naprav, definira skupen standard, ki omogoča hišnim aparatom različnih proizvajalcev polno medsebojno delovanje v avtomatiziranem hišnem sistemu.

Ta nova platforma imenovana CHAIN (*Ceced Home Appliances Interoperating Network*) definira protokol za povezovanje velikih aparatov v en sam sistem, ustvarjen za nadzor in avtomatizacijo ključnih storitev v nekem domu: npr. oddaljen nadzor delovanja aparatov, varčno upravljanje z energijo, oddaljeno diagnostiko in avtomatsko vzdrževanje naprav, posodobitev podatkov ali prenos novih podatkov ali programov, ter različne spletne storitve...

CHAIN omogoča polno medsebojno delovanje tudi pri uporabi različnih komunikacijskih tehnologij znotraj enega omrežja. To je omogočeno s CHAIN kompatibilnim hišnim strežnikom ali katerim drugim strežnikom, ki ima možnost razširitve za podporo različnih komunikacijskih tehnologij. Rezultat tega je **CHAIN virtualno omrežje**, ki integrira različne naprave kljub specifični komunikacijski tehnologiji.

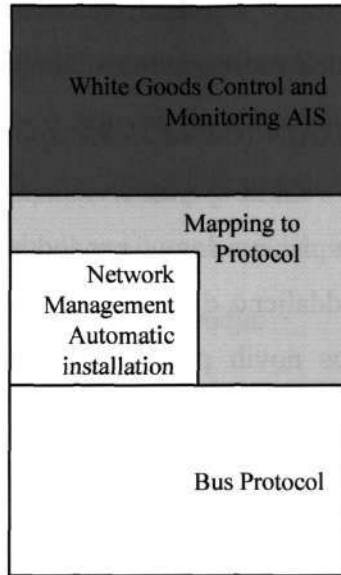
5.1 CECED Specifikacija

CECED specifikacija [12], imenovana AIS (*Application Interworking Specification*) pokriva **upravljanje in nadzor naprav** (*Appliances control and monitoring*). Dokument opisuje nabor funkcij povezanih s povezljivimi gospodinjskimi napravami v hišnem omrežju.

Cilj te specifikacije je natančna definicija na kakšen način lahko produkti različnih proizvajalcev medsebojno delujejo in komunicirajo, ter na kakšen način jih je mogoče instalirati v omrežje z ali brez za to potrebnih specialnih orodij na čim bolj avtomatiziran način.

Za doseganje takšnih ciljev je potrebno upoštevati (Slika 5.1):

- AIS dokumentacijo, ki se predvsem osredotoča na aplikacijska sporočila.
- Shema za preslikavo iz AIS na izbran komunikacijski protokol
- Skupne procedure za upravljanje omrežja pri avtomatskem instalacijskem postopku
- Skupen komunikacijski protokol



Slika 5.1: Odvisnosti od specifikacije

AIS specifikacija je sestavljena iz štirih delov oz. knjig (*Volume 1-4*), katerih pomembnejše definicije sledijo v naslednjih poglavjih:

1. AIS vol1: '**Functional specification**' - **Funkcionalna specifikacija** - na splošno vsebuje osnovne opise instalcijskih postopkov glede na različne kategorije gospodinjskih naprav, specifikira ključne dogodke in z njimi povezane uporabniške funkcije, podaja osnovni opis aplikacij hišnih naprav in njihovih diagramov stanj, ter specifikira nadzor in upravljanje medsebojno delujočih naprav.
2. AIS vol2: '**Data structure**' – **Struktura podatkov** – definira obliko, vrednosti ter strukturo sporočil ter podatkov
3. AIS vol3: '**Abstract Test Suite**' – **Testna garnitura** – vsebuje opis vseh potrebnih testov, ki napravi zagotavljajo CECED oz. CHAIN kompatibilnost
4. AIS vol4: '**Mapping**' – **Preslikava** – Definira način preslikave CECED sporočil v izbran komunikacijski protokol. Trenutno obstajajo tri podpoglavja za do sedaj specificirane protokole (EHS 1.3, KNX, LON)

5.1.1 Funkcionalna specifikacija (AIS Vol.1)

5.1.1.1 Pristop k instalaciji gospodinjskih naprav

Bistveni del tega dela AIS specifikacije [12] je opis instalacijskih postopkov. Poglejmo si torej nekaj opisov.

Instalacija povezljivih aparatov bele tehnike je sestavljena iz dveh pomembnejših faz:

1. Inicializacija specifičnega omrežnega podatkovnega protokola:
 - Hišni naslov oz. naslov domene (*home address, domain address*). Ta vrednost je uporabljena pri odprtih medijih (kot je električno napajalno omrežje ali RF) za razpoznavanje omrežnih sporočil znotraj ene hišne enote. Pomembno je, da aparati ne sprejemajo sporočil iz drugih domov v bližini.
 - Omrežni naslov (*network address*) instalirane naprave služi kot unikaten identifikator za napravo v omrežju. Z uporabo teh naslovov lahko naprave neposredno komunicirajo med seboj.
2. Inicializacija podatkov komunikacijskih povezav, ki so v bistvu sestavljeni iz omrežnih naslovov oddaljenih naprav.

Postopek instalacije mora zadostiti potrebam in tipičnim pričakovanjem današnje industrije in trga. Različne kategorije naprav zahtevajo različne pristope in postopke instalacije. Dejstvo je, da pri vseh napravah ni mogoče pričakovati popolne avtomatizacije instalacijskega postopka in ta tudi ni smiselna (npr. pri varnostnih sistemih).

Na splošno so pričakovanja glede instalacije gospodinjskih naprav (specifikacija jih omenja kar kot WG -*White Goods*), da mora ta biti enostavna, zavarovana in varna. Uporabnik lahko recimo samostojno izvede instalacijo brez pomoči specializiranega instalaterja. Lahko imamo popolnoma avtomatsko (prikluči in uporablja) instalacijo, ali pa takšno, pri kateri mora delno sodelovati tudi uporabnik (prikluči, stisni in uporablja) – recimo s pritiskom na določen gumb ali njihovo kombinacijo za začetek procedure.

Omejeno zaporedje aktivnosti ima tipično dva namena:

- Pridobivanje hišnega naslova

- Obvladovanje primera v katerem sodeluje več naprav istega tipa v istem domu (npr. več pralnih strojev). Funkcija, ki obvladuje to problematiko se imenuje '*Instance Identification*'

V praksi lahko pričakujemo, da bo kljub vsemu obstajalo veliko različnih pristopov k instalaciji:

- Način, kako doseči instalacijske aktivnosti na nivoju naprave, se lahko razlikuje od naprave do naprave, v kolikor ostanejo cilj teh aktivnosti in z njimi povezani protokoli enaki (le različna uporaba kombinacij gumbov).
- Aktivnosti na nivoju ostalih naprav so prav tako lahko zelo različne. Vključujejo lahko različne naprave od ene do druge instalacije, vse dokler ostajajo cilj in protokoli enaki. Na primer, znotraj sistema je možno uporabiti instalacijsko napravo za nadzor instalacije in operacij ostalih naprav. Ta specifična naprava je potem odgovorna za dodeljevanje hišnih naslovov ter za posamezno obravnavanje naprav iste vrste.

Kljub temu, da se lahko intervencija uporabnika pri instalaciji razlikuje od proizvajalca do proizvajalca, bi morali vsi uporabljati enak upravljavski omrežni protokol. To bi zagotavljalo interoperabilnost sistema oz. naprav.

5.1.1.2 Specifikacija ključnih dogodkov in uporabniških funkcij

Instalacijske procedure gospodinjskih naprav naj bi sledile skupnim scenarijem in pravilom. Bolj natančno, procedure bi morale biti opisane z enako listo ključnih dogodkov, ki zadevajo celotno instalacijo, vsak ključni dogodek pa naj bi bil povezan z uporabniško funkcijo.

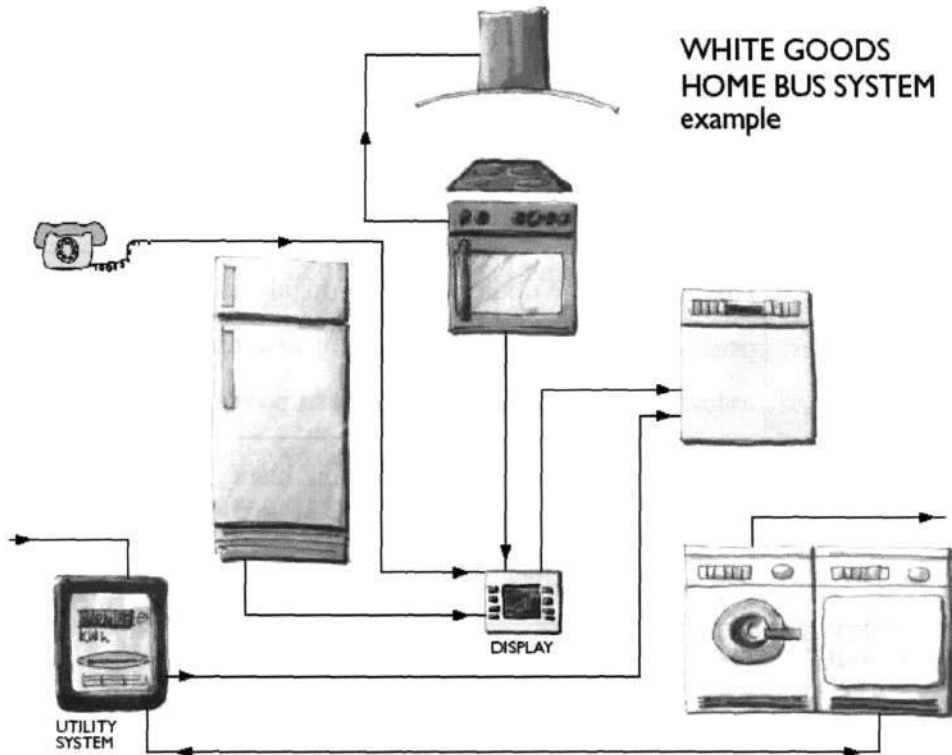
V naslednji tabeli so definirani ključni dogodki pri instalaciji:

Dogodek	Uporabniška funkcija
Prva instalacija	Povezljive naprave so prvič instalirane v nekem domu. Npr. instalacija hišnega strežnika in pralnega stroja
Vklop	Vklop ene ali več naprav (npr. vklop celotne instalacije po izpadu električne energije)
Dodajanje naprave	Nova naprava je dodana v hišno instalacijo. V nekaterih primerih je lahko naprava za nekaj časa uporabljena v avtonomnem načinu (brez možnosti povezave).
Odstranitev naprave	Obstoječa naprava je odstranjena iz hišnega sistema. To je lahko posledica fizične odstranitve, ali pa preprosto zato, ker smo napravo prestavili v avtonomen način (brez povezave)
Ponovna instalacija naprave	Del sistema je ponovno instaliran.
Ponovna instalacija	Cel sistem je ponovno instaliran.
Odstranitev sistema	Sistem je odstranjen. To se lahko zgodi recimo v primeru, ko se uporabnik seli na drugo lokacijo.
Nadzor komunikacijskih povezav	Izvedena je verifikacija komunikacijskih povezav.

Tabela 5.1 : Ključni dogodki pri instalaciji

5.1.1.3 Osnovni opis aplikacijskega področja hišnih naprav

Ta del specifikacije opisuje način povezovanja naprav na nivoju omrežja (slika 5.2).



Slika 5.2 : Primer aplikacije omrežja

Natančneje so opisane pravice za dostop do določenih delov omrežja in določenih podatkov (poglavje 3, Vol.1 AIS).

Opis funkcionalnosti naprav je sestavljen iz :

- **Prenosnih (*Transition*) ukazov:**

- START - Takojšnji zagon naprave ali nadaljevanje izvajanja programa
- STOP - Takojšnja zaustavitev izvajanja programa. Naprava sama 'odloča' o tem ali se ostale nastavitve naprave (program, časovni zamik...) izbrišejo
- PAUSE - Takojšen prehod v stanje pavze, ki omogoča kasnejše nadaljevanje
- ENABLE GAS - Omogočanje delovanja plinske naprave
- Ipd...

- **Ostalih ukazov:**

- ON – Zagon naprave
- OFF – Izklop naprave
- WASHING PARAMETERS- Informacije o izvajanju pralnega/sušilnega cikla.
- COOKING PARAMETERS - Informacije o izvajanju kuhalnega cikla.
- REFRIGERATION PARAMETERS – Informacije o izvajanju hladilnega cikla.
- CONDITIONAL START - Pogoji za uspešen zagon programa (npr. časovni zamik)
- FAILURE DETECTED – Zaznana je bila napaka
- Ipd...

- **Možna trenutna stanja naprave:**

- OFF – Naprava je ugasnjena
- STAND BY - Naprava čaka na nastavitve in trenutno se ne izvaja noben program.
- PROGRAMMED - Nastavitve za napravo so zaključene in poslane k napravi, ki čaka le še na sporočilo za start. Sprejme lahko dve ukazni sporočili: START ali pogojni start.
- PROGRAMMED WAITING TO START - Naprava je sprejela nastavitve in pogojni start, sedaj čaka na izpolnitev pogojev za začetek (časovni zamik, ustrezna tarifa električne energije,...).
- PAUSE - Naprava je trenutno v stanju pavze, ker je sprejela ukaz PAUSE. Po tem ko bo sprejet ukaz START, bo naprava nadaljevala z delom

Na nivoju naprav je opis podan z diagrami stanj. Primer takšnega diagrama za pralni stroj vidimo na sliki 5.3.

- Obvladovanje primera v katerem sodeluje več naprav istega tipa v istem domu (npr. več pralnih strojev). Funkcija, ki obvladuje to problematiko se imenuje '*Instance Identification*'

V praksi lahko pričakujemo, da bo kljub vsemu obstajalo veliko različnih pristopov k instalaciji:

- Način, kako doseči instalacijske aktivnosti na nivoju naprave, se lahko razlikuje od naprave do naprave, v kolikor ostanejo cilj teh aktivnosti in z njimi povezani protokoli enaki (le različna uporaba kombinacij gumbov).
- Aktivnosti na nivoju ostalih naprav so prav tako lahko zelo različne. Vključujejo lahko različne naprave od ene do druge instalacije, vse dokler ostajajo cilj in protokoli enaki. Na primer, znotraj sistema je možno uporabiti instalacijsko napravo za nadzor instalacije in operacij ostalih naprav. Ta specifična naprava je potem odgovorna za dodeljevanje hišnih naslovov ter za posamezno obravnavanje naprav iste vrste.

Kljub temu, da se lahko intervencija uporabnika pri instalaciji razlikuje od proizvajalca do proizvajalca, bi morali vsi uporabljati enak upravljavski omrežni protokol. To bi zagotavljalo interoperabilnost sistema oz. naprav.

5.1.1.2 Specifikacija ključnih dogodkov in uporabniških funkcij

Instalacijske procedure gospodinjskih naprav naj bi sledile skupnim scenarijem in pravilom. Bolj natančno, procedure bi morale biti opisane z enako listo ključnih dogodkov, ki zadevajo celotno instalacijo, vsak ključni dogodek pa naj bi bil povezan z uporabniško funkcijo.

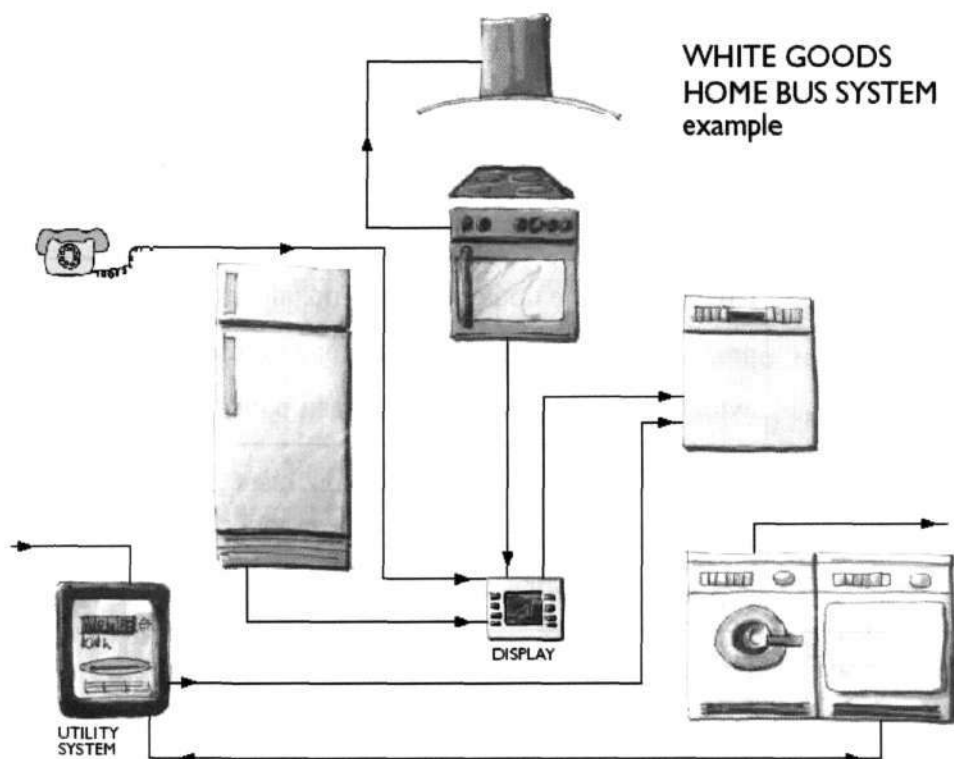
V naslednji tabeli so definirani ključni dogodki pri instalaciji:

Dogodek	Uporabniška funkcija
Prva instalacija	Povezljive naprave so prvič instalirane v nekem domu. Npr. instalacija hišnega strežnika in pralnega stroja
Vklop	Vklop ene ali več naprav (npr. vklop celotne instalacije po izpadu električne energije)
Dodajanje naprave	Nova naprava je dodana v hišno instalacijo. V nekaterih primerih je lahko naprava za nekaj časa uporabljena v avtonomnem načinu (brez možnosti povezave).
Odstranitev naprave	Obstoječa naprava je odstranjena iz hišnega sistema. To je lahko posledica fizične odstranitve, ali pa preprosto zato, ker smo napravo prestavili v avtonomen način (brez povezave)
Ponovna instalacija naprave	Del sistema je ponovno instaliran.
Ponovna instalacija	Cel sistem je ponovno instaliran.
Odstranitev sistema	Sistem je odstranjen. To se lahko zgodi recimo v primeru, ko se uporabnik seli na drugo lokacijo.
Nadzor komunikacijskih povezav	Izvedena je verifikacija komunikacijskih povezav.

Tabela 5.1 : Ključni dogodki pri instalaciji

5.1.1.3 Osnovni opis aplikacijskega področja hišnih naprav

Ta del specifikacije opisuje način povezovanja naprav na nivoju omrežja (slika 5.2).



Slika 5.2 : Primer aplikacije omrežja

Natančneje so opisane pravice za dostop do določenih delov omrežja in določenih podatkov (poglavje 3, Vol.1 AIS).

Opis funkcionalnosti naprav je sestavljen iz :

- **Prenosnih (*Transition*) ukazov:**

- START - Takojšni zagon naprave ali nadaljevanje izvajanja programa
- STOP - Takojšnja zaustavitev izvajanja programa. Naprava sama 'odloča' o tem ali se ostale nastavitve naprave (program, časovni zamik...) izbrišejo
- PAUSE - Takojšen prehod v stanje pavze, ki omogoča kasnejše nadaljevanje
- ENABLE GAS - Omogočanje delovanja plinske naprave
- Ipd...

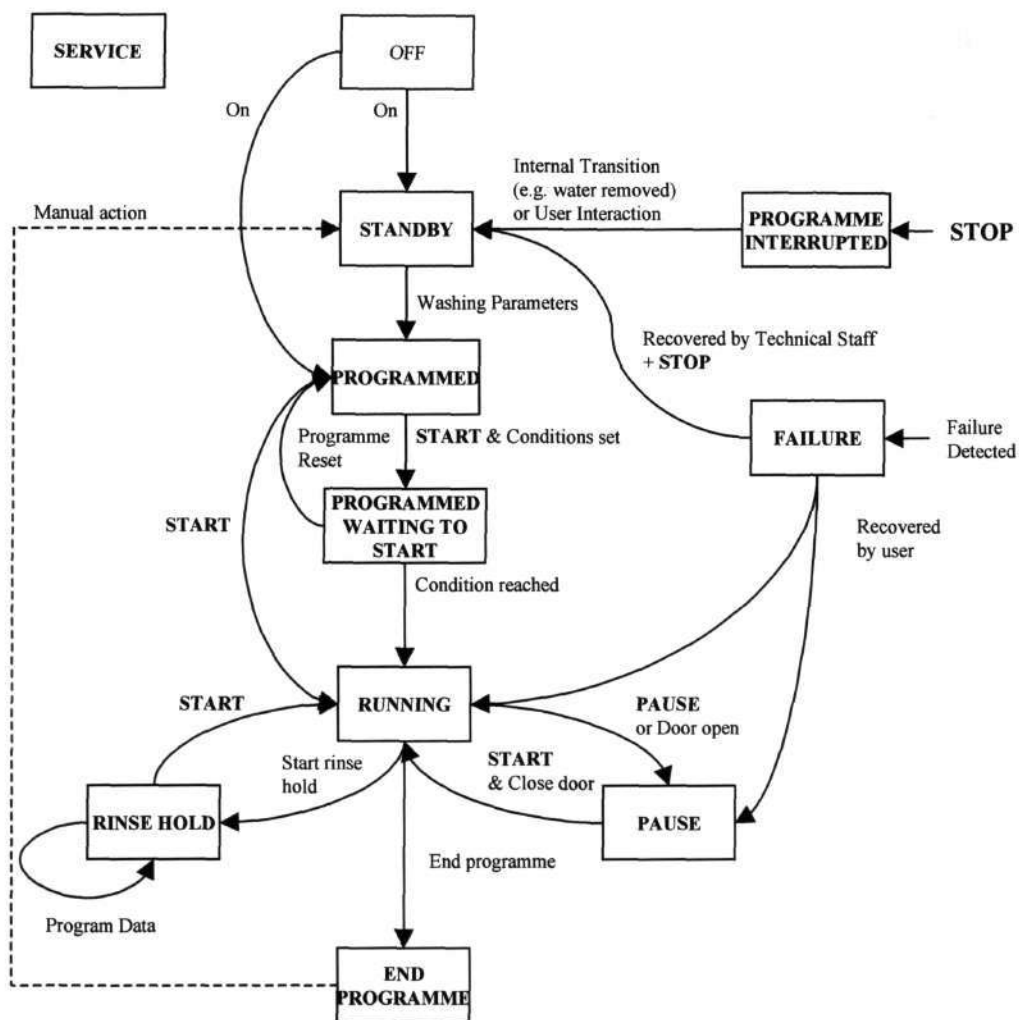
- **Ostalih ukazov:**

- ON – Zagon naprave
- OFF – Izklop naprave
- WASHING PARAMETERS- Informacije o izvajanju pralnega/sušilnega cikla.
- COOKING PARAMETERS - Informacije o izvajanju kuhalnega cikla.
- REFRIGERATION PARAMETERS – Informacije o izvajanju hladilnega cikla.
- CONDITIONAL START - Pogoji za uspešen zagon programa (npr. časovni zamik)
- FAILURE DETECTED – Zaznana je bila napaka
- Ipd...

- **Možna trenutna stanja naprave:**

- OFF – Naprava je ugasnjena
- STAND BY - Naprava čaka na nastavitve in trenutno se ne izvaja noben program.
- PROGRAMMED - Nastavitve za napravo so zaključene in poslane k napravi, ki čaka le še na sporočilo za start. Sprejme lahko dve ukazni sporočili: START ali pogojni start.
- PROGRAMMED WAITING TO START - Naprava je sprejela nastavitve in pogojni start, sedaj čaka na izpolnitev pogojev za začetek (časovni zamik, ustrezna tarifa električne energije,...).
- PAUSE - Naprava je trenutno v stanju pavze, ker je sprejela ukaz PAUSE. Po tem ko bo sprejet ukaz START, bo naprava nadaljevala z delom

Na nivoju naprav je opis podan z diagrami stanj. Primer takšnega diagrama za pralni stroj vidimo na sliki 5.3.



Slika 5.3: Diagram stanj za pralni stroj

5.1.1.4 Razumevanje interoperabilnosti

V četrtem poglavju prvega dela AIS specifikacije je natančno opisan koncept interoperabilnosti.

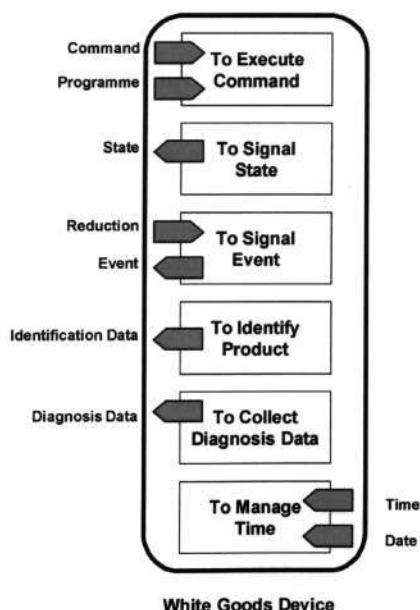
Za razumevanje tega dela si pogledjmo nekaj bistvenih definicij tega odseka, natančne razlage posameznih definicij so v sami specifikaciji:

- Gospodinske naprave med seboj sodelujejo preko **sporočil**.
- Vsako sporočilo odgovarja točno določeni **operaciji**, ki je aplicirana na **komunikacijski objekt**, ponavadi z določenimi podatki.
- Komunikacijski objekt je drugače imenovan **OID**

- OID je upravljan s strani gospodinjskega aparata, ali pa s strani naprave, ki trenutno sodeluje z tem aparatom.
- Operacija je sestavljena iz **prenosnega tipa** (individualnega ali skupinskega) in enega **primitiva** (pridobi vrednost, vrni vrednost, pošlji vrednost, spremeni vred.)
- Podatkovni del sporočila vsebuje **aplikacijske podatke**, ki jim po možnosti sledijo aplikacijski podatki
- **Opis vpliva sporočila** na napravo imenovan tudi **MID** (*message interaction descriptor*) je sestavljen iz enega OID in niza možnih operacij s podatki
- MID-i so funkcionalno razporejeni v skupine imenovane **funkcionalni bloki** naprav.

5.1.1.5 Funkcionalni bloki

Vsaka povezljiva gospodinjska naprava je strukturirana v komponente imenovane funkcionalni bloki. Vsak funkcionalni blok je odgovoren za obvladovanje specifičnih MID-ov. Na sliki 5.4 vidimo šest funkcionalnih blokov, ki so pomembni z vidika upravljanja oz. obvladovanja naprave in so specificirani v trenutni zadnji verziji AIS specifikacije:



- Izvršitev ukaza (*To Execute Command*)
- Prikaz stanja (*To Signal State*)
- Objava dogodka (*To Signal Event*)
- Identifikacija produkta (*To Identify Product*)
- Pridobivanje diagnostičnih podatkov (*To Collect Diagnosis Data*)
- Upravljanje s časom (*To Manage Time*)

Slika 5.4 Struktura funkcionalnega bloka za gospodinjske naprave

Vsak funkcionalni blok je povezan z enim ali več MID-i, kar je prikazano na sliki 5.4 s puščicami. Smer puščice prikazuje glavni tok podatkov. Na primer za izvršitev ukaza ima funkcionalni blok dva MID-a: prvi je povezan k ukaznemu OID in drugi k Programskemu OID.

5.1.1.6 Pravila medsebojnega delovanja aplikacij

Medsebojno delovanje mora upoštevati dva tipa različnosti:

1. Različnost profilov

Za vsak tip naprave bo obstajal profil ali pred definiran nabor specifikacije.

Ta nabor odgovarja izbiri obveznih (*mandatory*), opcijskih (*optional*), ne-standardiziranih (*non-standardised*) in lastniških (*proprietary*) lastnosti. (natančno so opisani v specifikaciji).

- **Obvezne** so tiste lastnosti, ki so standardizirane in morajo biti nujno implementirane z napravo.
- **Opcijske lastnosti** so prav tako standardizirane, vendar njihova implementacija ni nujna.
- **ne-standardizirane** lastnosti so implementirane v še ne standardizirane naprave in zato še niso vključene v to AIS specifikacijo. So prav tako natančno opisane in dokumentacija je javno dostopna s strani posameznega proizvajalca.
- **Lastniške** lastnosti pa so implementirane v napravi za namene lastniškega medsebojnega delovanja. Te posameznim proizvajalcem omogočajo implementacijo specifičnih funkcij, ki jih le ta želi zaščititi (konkurenca med proizvajalci).

2. Različnost verzij

Verzije odgovarjajo bodočemu razvoju specifikacije. V bodočih verzijah bodo dodane nove obvezne, opcijske, ne-standardizirane in lastniške lastnosti.

Medsebojno delovanje je lahko zagotovljeno le, če so dovoljene razlike definirane na kompatibilen način. To pomeni, da je potrebno zagotoviti oz. upoštevati nekaj pravil. Ta

pravila so natančno opisana v četrtem poglavju *Vol.1 AIS specifikacije 'Application Interworking Rules'*.

5.1.2 Struktura podatkov (AIS Vol.2)

V drugem delu specifikacije [12] so posamezni objekti, ki so omenjeni in opisani v prvem delu specifikacije, natančno določeni s pripisanimi heksadecimalnimi vrednostmi. Primer vidimo na sliki 5.5, kjer je prikazan izsek iz specifikacije, ki opisuje preslikavo v funkcionalni blok ukazov za izvrševanje (*To Execute Command*).

OID/MID	Data
Execution of a Command	Byte 0: Command Identification
	Values ranging from 1 to 127 are standardised
	START 1
	STOP 2
	PAUSE 3
	START SUPERFREEZING 4
	STOP SUPERFREEZING 5
	START SUPERCOOLING 6
	STOP SUPERCOOLING 7
	DISABLE GAS 8
	ENABLE GAS 9
Washing Parameters (Programme Data for WET)	Values greater or equal to 128 are proprietary
	Byte 0: Type of Programme Data
	Values from 1 to 128: reserved for standardised messages
Cooking Parameters (Programme Data for HOT)	Values from 128_ to 255: reserved for group 1 proprietary messages
	Other bytes: programme data
Refrigeration Parameters	See Washing Parameters

Slika 5.5 : Preslikava v funkcionalni blok

V tem delu so posredovane tudi različne tabele z vrednostmi, kot prikazuje izsek naslednje tabele (*Product Names and Product Types*):

Appliance	Product Name	Cluster	Category (Hex)	Id (Hex)	Product Type Id (Hex)
Combi	CB	6	General:1	00	0E00
Air Conditioner	AC	6	Ventilation:2	03	1603
Dishwasher	DW	6	Wet:0xA	01	5601
Tumble Dryer	TD	6	Wet:0xA	02	5602
Washer Dryer	WD	6	Wet:0xA	03	5603
Washing Machine	WM	6	Wet:0xA	04	5604
Gas Oven	GO	6	Hot:0xB	01	5E01
Gas Cook top	GT	6	Hot:0xB	02	5E02
...	...	...	...	...	...

5.1.3 Testna garnitura (AIS Vol.3)

V tretjem delu [12] so specificirani vsi potrebni testni postopki, ki omogočajo certificiranje naprave - kot CECED kompatibilne naprave.

Testni del specifikacije je sestavljen iz:

- **PICS** (*Protocol Implementation Conformance Statement*)

Se uporablja za specifikacijo stopnje prilagoditve testov za dano napravo. Sestavljen je iz spiska obveznih in opsijskih sposobnosti, ki jih naprava podpira. Terminologija je povzeta po ISO9646 standardu za prilagoditev testiranja pri implementaciji protokola.

- **PIXIT** (*Protocol Implementation eXtra Information*)

Tukaj so podane dodatne zahtevane informacije za testiranje, ki niso definirane pri PICS. Te informacije so specifične za napravo, ki je v postopku testiranja (ime proizvajalca, znamka...). Terminologija je prav tako povzeta po ISO9646 standardu.

- **Niza testnih primerov**

PICS

Namen PICS je ustvariti spisek sposobnosti, ki jih podpira CECED kompatibilna naprava.

Tipi sposobnosti:

- **obvezne za vse naprave** (npr. status naprave) – te sposobnosti niso našteve v PICS
- **obvezne za podan tip naprave** (npr. vsi pralni stroji morajo implementirati START ukaz)
- **obvezno NE-podpiranje za dano napravo** (npr. obvezno je NE-upoštevati START ukaz pri plinskih štedilnikih)
- **opsijske sposobnosti** (npr. START ukaz je opsijski za mikrovalovne pečice)

Vsaka sposobnost je opisana skozi:

- opis/vprašanje (npr. 'Je START ukaz razumljiv?')
- odgovor ('da' ali 'ne')
- abstraktno logično spremenljivko testne specifikacije, ki je uporabljena kot shramba za odgovor (npr. logična spremenljivka '*pCommand Start*' bo hranila vrednost '*True*'/'*False*' glede na odgovor Da/Ne

PIXIT

PIXIT so nadaljevalne tabele uporabljane za potrditev testne seje. Vsebujejo vrednosti, ki so specifične za dano napravo.

Vsak vnos v tabelo je opisan skozi:

- opis (npr. 'Ime podjetja testirane naprave (IUT -*Implementation Under Test*))
- format podatkov
- abstraktno spremenljivko testne specifikacije, ki je uporabljena za shrambo vrednosti (npr. spremenljivka '*xBrandName*' bo vsebovala trgovsko znamko naprave)

Primer vnosa opisa v tabelo :

Brand Id of IUT	String : 2 char	xBrandId
-----------------	-----------------	----------

Sintaksa testnih primerov

Testni primer je opisan skozi pregledno notacijo. Osnova za format te notacije je povzeta iz TTCN (*Tree Tabular Combined Notation*) iz ISO 9646 standarda.

V naslednjem primeru (slika 5.6) je prikazano kaj vsebuje testni primer:

- Nekaj globalnih informacij: ime, naslov, povzetek in pred-pogoje
 - o Pred-pogoji so logični izrazi, ki omogočijo uporabo PICS logičnih spremenljivk. Dovoljujejo določanje pogojev za testiranje sposobnosti naprave. V primeru na sliki 5.6 je pred-pogoj podpora prenosa od neuspeha (*Failure*) do mirovanja (*Stand-by*), ko je bil izdan ukaz STOP.
- Dva stolpca naslovljena '*stimulus*' in '*response*'

- *Stimulus* (spodbuda) stolpec vsebuje operacije, ki so lahko izvedene s potrditvenim testerjem za preizkušanje naprave.
 - *Response* (odziv) stolpec vsebuje operacije oz. odzive, ki jih izvede naprava ali operater.
 - Vsaka operacija lahko vsebuje večje število parametrov - npr. '*Data[0]* = *cCmdData_Stop*' pomeni, da je prvi zlog podatkov vrednost *cCmdData_Stop*'. Takšne vrednosti so inicializirane s strani testerja v primeru zahteve do naprave, ali s strani naprave v primeru odziva na zahtevo.
- Operacije so klici metod na objekte. Spisek objektov in metod z njihovo semantiko je definiran v tej testni garnituri. V primeru na sliki 5.6 sta uporabljena dva objekta: '*Operator*' in '*MID*'. *Operator* je povezan z metodama naredi ('*do*') in opravljeno ('*done*'). *MID* pa je povezan z metodama pošlji ('*send*') in sprejmi ('*received*').
 - Zaporedje operacij v testnem primeru je razdeljeno na dva dela: **uvod** (*preamble*) in **telo**. Uvod so operacije, ki so potrebne za nastavitve IUT v specifično stanje. Telo pa vsebuje operacije, ki dejansko izvajajo test.

Name:	Foo	
Title:		
Summary:	Example of test case	
Precondition:	(pCommand_Start_ProgrammedToRunning = yes)	
Stimulus		Response
Preamble		
Operator.do .action = GoToProgrammedMode		Operator.done
MID.Send .OID = cOID_DeviceStatus .Primitive = cPrimitive_GetValue .Data = None		MID.Received .OID = cOID_DeviceStatus .Primitive = cPrimitive_ReturnValue .Data[0] = cStateData_Programmed .Data[1..x] = <i>Not Verified</i>
Body		
MID.Send .OID = cOID_ExecutionOfACommand .Primitive = cPrimitive_ChangeValue .Data[0] = cCmdData_Start		MID.Received .OID = cOID_DeviceStatus .Primitive = cPrimitive_SendValue .Data[0] = cStateData_Running .Data[1..x] = <i>Not Verified</i>

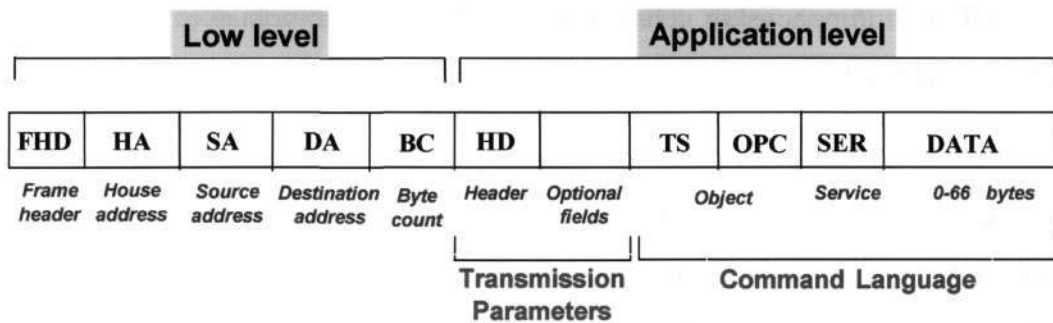
Slika 5.6: Primer opisa testnega primera

5.1.4 Preslikave na komunikacijske protokole (AIS Vol.4)

Do danes so bili ustvarjeni trije dokumenti v četrtem delu specifikacije [12]:

1. AIS vol4a **EHS Mapping** (v 0.96) (edini, ki je trenutno edini potrjen)
2. AIS vol4b **Lonmark-EN14908 Mapping** (v 0.96)
3. AIS vol4c **KNX A-mode Mapping Draft** (v 5)

Ti dokumenti imajo podobno zgradbo. Najprej je splošen opis izbranega komunikacijskega standarda oz. protokola. Temu sledi opis izgleda sporočila (primer na sliki 5.7).



Slika 5.7: Format okvirja za EHS

Nato pa sledijo vse potrebne tabele s podatki in preslikavami za posamezen protokol.

6 OSGi

Zveza OSGi (*Open Services Gateway initiative*) [13] je bila ustanovljena meseca marca leta 1999. Njeni ustanovitveni člani so si delili enako vizijo:

1. Omrežen dom je naslednje novo področje standardizirane servisne platforme.
2. Internet in nove tehnologije omogočajo nove storitve, vrednostne verige (angl.: value chains) in poslovne modele.
3. Pospeševanje teh možnosti zahteva skupen standard odobren z različnih strani v industriji vključno s ponudniki storitev, distributerji komunikacijskih storitev, ponudniki informacijskih tehnologij, distributerji avtomobilske elektronike, ter proizvajalci naprav.

Tako je nastala OSGi zveza, ki specificira, kreira, pospešuje in promovira uporabo odprtih sistemov za ponujanje storitev in temu primerno upravljavsko platformo na širokem industrijskem področju. Zveza služi kot osrednja povezovalna točka za sodelovanje ponudnikov storitev, razvijalcev, proizvajalcev in uporabnikov.

OSGi tehnologija je načrtovana na način, ki omogoča lažji razvoj novih, zanimivih storitev in aplikacij za novo generacijo povezljivih naprav. Z izkoriščanjem teh edinstvenih prodajnih možnosti lahko proizvajalci naprav, ponudniki storitev in razvijalci programske opreme izboljšajo/skrajšajo čas od razvoja do trgovine.

Kot neodvisna, ne-profitna organizacija, OSGi zveza omogoča jasno in enotno kreiranje in distribucijo pomembne intelektualne lastnine, vključno s specifikacijami, referenčnimi implementacijami in testnimi garniturami za vse svoje člane.

OSGi specifikacija je široko uporabna, ker predstavlja majhno plast, ki omogoča učinkovito sodelovanje več, na *Javi* osnovanih, komponent na eni sami JVM (*Java Virtual Machine*). Zagotavlja obširen varnostni model, ki omogoča delovanje komponent v zaščitenem okolju.

Kakorkoli, z dodeljenimi pravicami so lahko komponente ponovno uporabljene in lahko sodelujejo med seboj, za razliko od ostalih Java programskih okolij, kjer to ni mogoče. OSGi okvir priskrbi obširno polje mehanizmov za omogočanje varnega sodelovanja.

Uporaba OSGi specifikacije torej omogoča zmanjšanje stroškov pri razvoju programske opreme in obenem omogoča nove poslovne priložnosti.

6.1 Storitvena platforma OSGi

Storitvena platforma zagotavlja odprto [13], skupno arhitekturo (ponudnikom storitev, razvijalcem, ponudnikom programske opreme, upravljavcem strežnikov, ponudnikom opreme) za razvoj in upravljanje storitev na koordiniran način. Zaradi svoje fleksibilne in upravljane razporeditve, ter uporabe storitev, omogoča popolnoma novo kategorijo 'pametnih' naprav. Primarne 'tarče' za uporabo OSGi storitvene platforme so naprave, kot npr. *set-top-box*, storitveni strežniki, kabelski modemi, potrošniška elektronika, PC-ji, industrijski računalniki, avtomobili in druge. Te naprave z implementirano OSGi specifikacijo bodo storitvenim ponudnikom, kot so ponudniki telekomunikacijskih storitev, kabelski operaterji, servisna podjetja in drugi, omogočale dostavo raznolikih vrednih storitev preko njihovih omrežij.

OSGi okvir (*framework*) in specifikacija olajšuje instalacijo in delovanje več storitev na enotni platformi. V specifikaciji je zasnovan API (*Application Programming Interface*) standard okolja za izvajanje platforme. Ustreznost storitvenih platform z OSGi specifikacijo je pogojena s podporo API aplikacij.

Storitve, ki so izvedene z API so lahko: upravljanje celotnega življenjskega cikla, med-storitvene odvisnosti, podatkovno upravljanje, upravljanje naprav, dostop uporabnikov, upravljanje virov in varnost. Uporaba teh omogoča končnim uporabnikom uporabo omrežnih storitev na zahtevo ponudnika storitev, medtem ko se platforma ukvarja z upravljanjem instalacije in konfiguracije teh storitev.

Dolžnost OSGi zveze je priprava specifikacije, ki ustreza novim in prihajajočim potrebam trga.

Glede na vse identificirane zahteve, OSGi ekspertne skupine (ki se ukvarjajo z arhitekturo naslavljanja, osnovno platformo in storitvami) razvijajo dodatke/podaljške (*extensions*),

nove servisne/storitvene API-je in nove funkcionalnosti za enostavnejše programiranje, ter obenem za izboljšanje uporabniške administracije in nastavitvenega upravljanja.

OSGi specifikacije so javno dostopne. Dodatno zveza podpira razvojno orodje in vire preko svojih spletnih strani in ostalih sredstev in s tem omogoča ponudnikom storitev in proizvajalcem naprav izboljšavo vsebine in storitev glede na množico ostalih razvijalcev.

6.1.1 Vrste storitev in njihove koristi

OSGi specifikacija omogoča ponudnikom storitev in ostalim, da obvladujejo in upravljajo že obstoječo telefonijo, podatkovna omrežja in zabavne storitve, istočasno pa omogoča povsem nove storitve z dodano vrednostjo (npr. upravljanje z energijo, telematika, avtomatizacija in nadzorovanje doma, programsko posodabljanje naprav). Te storitve je mogoče nuditi preko Interneta, različnih širokopasovnih povezav in hitrih brezžičnih podatkovnih omrežij v domu, avtomobilu, preko mobilnega telefona in drugih možnih okolij.

Specifikacija je načrtovana na način, ki dopolnjuje in razširja praktično vse stanovanjske in mobilne omrežne standarde in iniciative. Na enak način specifikacija povečuje vrednost obstoječih žičnih in brezžičnih omrežij, s tem, da omogoča fleksibilnost do vseh širokopasovnih tehnologij. Aplikacije, ki so omogočene z OSGi storitveno platformo vključujejo:

- Storitve za dom:

- Komunikacije
- Informacije/zabava
- Varnostne in nadzorne sisteme
- Upravljanje z energijo in meritve porabe
- Diagnosticiranje in servisiranje naprav
- Medicinsko in zdravstveno nadzorovanje

- Storitve za avtomobile:

- Navigacija

- Asistenca pri nesrečah
- Mobilna trgovina
- Informacije/zabava
- Diagnostika vozil
- Lokacijsko povezane storitve

Storitvena platforma ponuja veliko koristi širokemu krogu uporabnikov, ker omogoča dinamično dostavo veliko vrst storitev na zahtevo:

- Korist za ponudnike storitev in ponudnike različnih vsebin je pri ustvarjanju novih virov oz. tokov zaslужka na takšen način, ter pri zmanjševanju stroškov pri servisnih storitvah (npr. manj dejanskih prevozov po terenu).
- Omrežni operaterji najdejo koristi pri povečani uporabnosti omrežja, večji zvestobi uporabnikov in zmanjšanem nezadovoljstvu kupcev oz. uporabnikov.
- Koristi proizvajalcev se kažejo pri razširitvi aplikacij naprav in razvoju skupnega oz. enotnega API okvirja.
- Končni uporabniki imajo tako znižane stroške in kompleksnost povezljivosti, medtem ko lahko koristijo prednosti novih storitev na zelo varen način.

6.1.2 Lastnosti storitvene platforme

OSGi storitvena platforma je zbirka API-jev, ki definirajo standarde za ogrodje servisne platforme za naprave. Nabor bistvenih (jedro) in opcijskih API-jev skupaj definira OSGi kompatibilno storitveno platformo. Kjer je možno, uporablja platforma obstoječo specifikacijo Java tehnologije, kjer uporaba te tehnologije v osnovi ni mogoča, pa je delo skupin osredotočeno ravno na integracijo teh standardov.

Osnovne API aplikacije pokrivajo naslavljanje dostave storitev, odvisnosti in upravljanje skozi življenjski cikel, upravljanje virov in oddaljeno storitveno administracijo. Vse bistvene osnovne API aplikacije so prispevane od članov zveze ali pa so razvite v OSGi tehnični ekspertni skupini.

Opcijske API aplikacije pa definirajo mehanizme za izvažanje virov na HTTP- osnovan spletni strežnik, mehanizme za interakcijo uporabnikov s servisno platformo in upravljanje s podatki.

Java je uporabljena zaradi svoje prenosne tehnologije, ki lahko deluje na različnih platformah vključno z različnimi tipi naprav (avtomobilska in potrošniška elektronska oprema, gospodinjske naprave, komunikacijske naprave, računalniki,...)

6.1.3 OSGi - Interoperabilnost standardov

OSGi storitvena platforma je 'prijazna' do večine standardov, ki so trenutno atraktivni. Fokusira se na aplikacijski nivo in je odprta za skoraj vse protokole. Kjer že obstaja Java specifikacija (npr. *Jini*), jo OSGi platforma lahko uporabi. Kjer pa je v uporabi standard, ki ne bazira na Javi (*HAVi*, *Universal Plug and Play*), pa se OSGi platforma osredotoča na povezovanje teh standardov na konsekventen način.

API aplikacije za upravljanje naprav na OSGi storitveni platformi podpirajo standarde kot so: **HomePlug**, **X-10**, **CEBus**, **LonWorks**, **Konnex**, ter zagotavljajo povezovanje in doseganje teh naprav na osnovi Java tehnologije. Podobno je zagotovljeno delovanje tudi za brezžične standarde, kot sta **Bluetooth** in **Wi-Fi**.

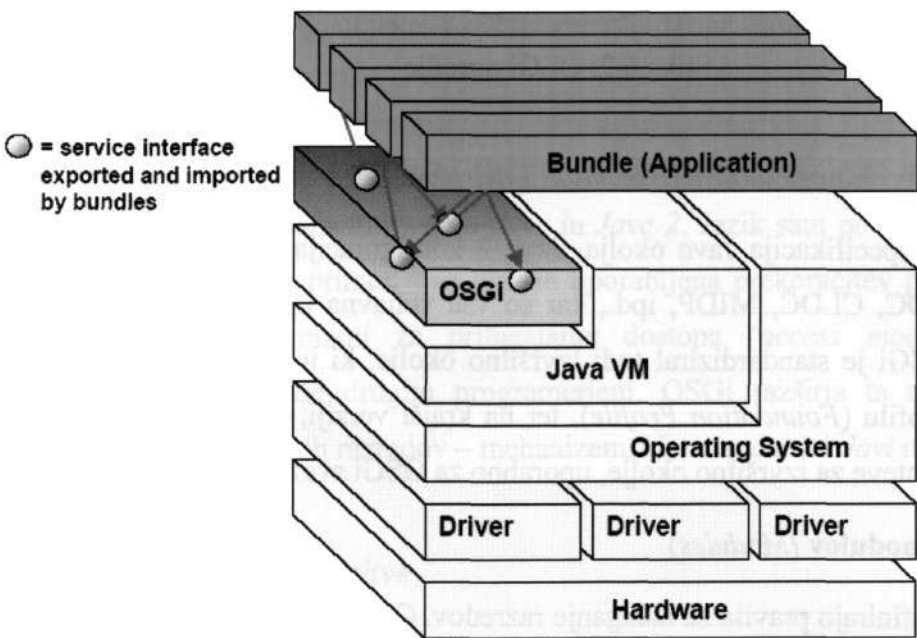
Prav tako je OSGi standard uporaben skupaj s katerokoli verzijo Windows operacijskega sistema ali katerimkoli drugim operacijskim sistemom, ki podpira *Java Virtual Machine - JVM*.

6.2 OSGi tehnologija

OSGi specifikacija [13] definira standardizirano, komponentno orientirano računalniško okolje za omrežne storitve. Dodajanje OSGi storitvene platforme k napravam omogoča upravljanje življenjskega cikla programskih komponent v napravi iz katerekoli točke v omrežju. Programske komponente so lahko instalirane, posodobljene ali odstranjene na hiter in enostaven način, brez motenega delovanja naprave. Te programske komponente predstavljajo knjižnice ali aplikacije, ki lahko dinamično odkrivajo in uporabljajo ostale

komponente. Programske komponente je mogoče kupiti ali samostojno razviti. OSGi zveza je razvila veliko vmesnikov za standardne komponente, ki so dostopne preko skupnih funkcij, kot so http strežniki, konfiguracija, prijavljanje, varnost, uporabniška administracija, XML in druge.

Arhitektura programskih komponent se dotika naraščajočega problema v razvoju programske opreme - veliko število konfiguracij, ki jih je potrebno razviti in nato vzdrževati. Standardizirana arhitektura OSGi komponent (Slika 6.1), precej poenostavi ta nastavitven proces.

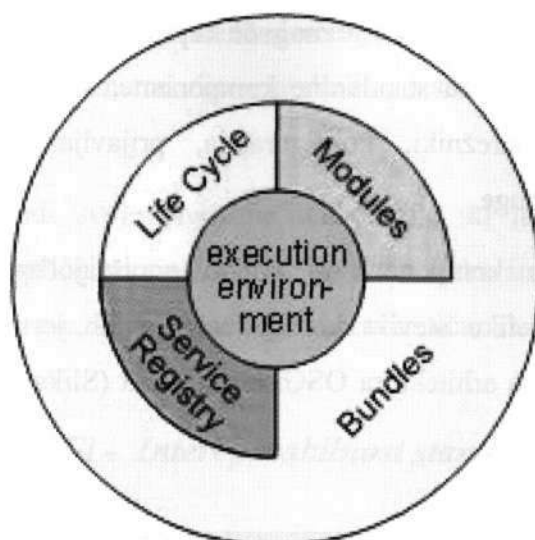


Slika 6.1: Arhitektura hišnega strežnika z uporabo OSGi okvirja

6.2.1 OSGi okvir (Framework)

Osnovna komponenta OSGi specifikacije je tako imenovan **OSGi okvir**, ki aplikacijam, imenovanim **svežnji (bundle)** zagotavlja standardno okolje.

OSGi ogrodje je razdeljeno na nekaj nivojev (slika 6.2):



Slika 6.2: OSGi ogrodje

- **Nivo izvršilnega okolja** (*Execution Environment*)

Je specifikacija Java okolja. Java 2 konfiguracija in profili, kot so J2SE, CDC, CLDC, MIDP, ipd., kar so vsa veljavna uporabna izvršilna okolja. OSGi je standardiziral tudi izvršilno okolje, ki je osnovano na temeljnem profilu (*Foundation Profile*), ter na krajši verziji, ki specificira minimalne zahteve za izvršilno okolje, uporabno za OSGi sveženj.

- **Nivo modulov** (*Modules*)

Definirajo pravila za nalaganje razredov. OSGi ogrodje ima strogo definiran model za nalaganje razredov. V Javi običajno obstaja ena definirana pot (*classpath*), ki vsebuje vse razrede in potrebne vire. OSGi nivo modulov doda privatne razrede za module, kot tudi kontrolirano povezovanje med moduli.

- **Nivo upravljanja življenjskega cikla** (*Life Cycle Management*)

Ta nivo doda svežnje, ki so lahko dinamično instalirani, zagnani, ustavljeni, posodobljeni in de-instalirani. Svežnji so glede nalaganja razredov odvisni od nivoja modulov, dodajo pa API aplikacijo za upravljanje modulov v času delovanja. Nivo upravljanja življenjskega cikla predstavlja dinamiko, ki običajno ni del aplikacije. Uporabljena je široka odvisnost mehanizmov za zagotavljanje pravilnega delovanja okolja.

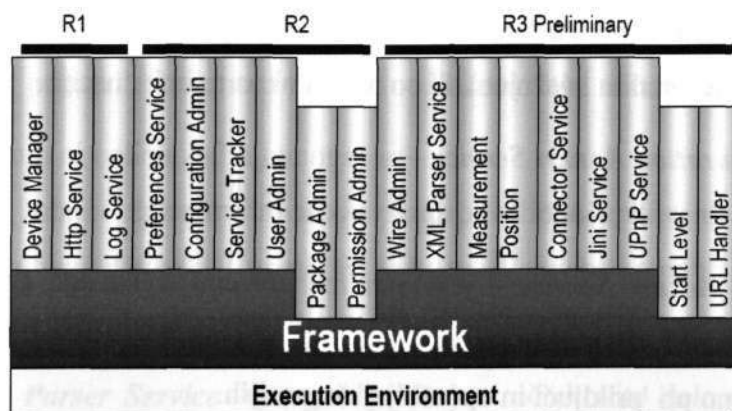
- Nivo registracije storitev (*Service Registry*)

Registracija storitev priskrbi svežnjem model sodelovanja, ki upošteva dinamiko. Svežnji lahko sodelujejo preko tradicionalne skupne uporabe razredov, vendar ta način ni preveč kompatibilen z dinamičnim instaliranjem in de-instaliranjem kode. Zato nivo registracije storitev priskrbi razumljiv model za izmenjavo objektov med svežnji. Številni dogodki so definirani za obvladovanje prihajajočih in odhajajočih storitev. Storitve so le *Java* objekti, ki lahko predstavljajo karkoli. Veliko storitev je v obliki strežniških objektov (npr. HTTP strežnik), druge storitve pa predstavljajo objekte v resničnem svetu (npr. Bluetooth telefon).

Dodatno je vsebovan tudi **varnostni sistem**, ki je globoko prepleten med vsemi nivoji. Varnost temelji na varnostnem modelu *Jave* in *Jave 2*. Jezik sam po sebi omejuje število različnih konstrukcij. Na primer, pri virusih uporabljena prekoračitev predpomnilnika je tukaj nemogoča. Mehanizmi za prilagajanje dostopa (*access modifiers*) v jeziku preprečujejo vidnost kode drugim programerjem. OSGi razširja ta model s tem, da dovoljuje uporabo privatnih razredov – mehanizem, ki v standardni *Javi* ni omogočen.

6.2.2 Storitve

Nad OSGi ogrodjem je OSGi zveza definirala veliko storitev (Slika 6.3), ki so specificirane z Java vmesnikom.



Slika 6.3: OSGi servisi/storitve

Svežnji lahko implementirajo te vmesnike in registrirajo storitev z nivojem registracije storitev. Odjemalci lahko to storitev poiščejo v registru, ali reagirajo nanjo, ko se ta pojavi/izgine.

Storitve ogrodja (Framework Services):

OSGi okvir omogoča *Permission Admin Service*, *Package Admin Service* in *Start Level Service*. Ti servisi/storitve so opsijski del in narekujejo delovanje ogrodja.

- ***Permission Admin*** – Ta servis upravlja s pravicami trenutnih ali bodočih svežnjev. Pravice so aktivirane takoj po njihovih nastavitvah.
- ***Package Admin*** – svežnji uporabljajo skupne pakete razredov in virov. Njihova posodobitev lahko povzroči potrebo po ponovnem pregledu medsebojnih odvisnosti. *Package Admin service* priskrbi informacijo o dejanskem stanju skupne uporabe paketov v sistemu, lahko pa tudi osveži skupne pakete (ponovno določi odvisnosti)
- ***Start Level*** – je nabor svežnjev, ki naj bi skupaj delovali ali bili inicializirani pred začetkom delovanja ostalih. *Start Level Service* nastavi trenutne začetne nivoje, pripiše sveženj začetnemu nivoju in razišče trenutne nastavitve.

Storitve sistema:

Storitve sistema omogočajo horizontalne funkcije, ki so praktično potrebne v vsakem sistemu. Primeri teh storitev so *Log Service*, *Configuration Admin Service*, *Device Access Service*, *User Admin Service*, *IO Connector Service* in *Preferences Service*.

- ***Log Service*** – prijava informacij, opozoril, sporočil o razhroščevanju ali napakah je upravljana skozi ta servis, ki sprejema 'log' vpise, ter jih nato posreduje ostalim svežnjem, ki so naročeni na te informacije.
- ***Configuration Admin Service*** – ta zagotavlja fleksibilen in dinamičen model za nastavljanje ali pridobivanje nastavitvene informacije.
- ***Device Access Service*** – je OSGi mehanizem za iskanje gonilnika za novo napravo in avtomatski prenos svežnja, ki implementira ta gonilnik. To je uporabno pri 'prikluči in uporablaj' scenarijih.
- ***User Admin Service*** – za namene avtentikacije in autorizacije uporablja bazo podatkov z uporabniškimi informacijami (javnimi in privatnimi).

- ***IO Connector Service*** – implementira *CDC/CLDC javax.microedition.io* paket kot servis/storitev. Ta omogoča svežnjem zagotavljanje novih in alternativnih protokolnih shem.
- ***Preferences Service*** – omogoča dostop do hierarhične podatkovne baze lastnosti (podobno kot *Windows Registry* ali *Java Preferences class*).

Storitve protokola:

OSGi zveza je definirala številne servise, ki preslikajo zunanji protokol na OSGi servis.

- ***Http Service*** – poleg ostalih stvari omogoča tudi delovanje aplikacij namenjenih za strežnike (*servlet runner*). Svežnji lahko priskrbijo te aplikacije, ki postanejo dosegljive preko HTTP protokola. Zaradi zmožnosti dinamičnega posodabljanja OSGi storitvene platforme je http servis zelo zanimiv spletni strežnik, ki je lahko posodobljen z novimi aplikacijami, če je potrebno tudi daljinsko, brez zahteve po ponovnem zagonu.
- ***UPnP Service*** – UPnP (*Universal Plug and Play*) je prihajajoč standard za uporabniško elektroniko. OSGi-jev *UPnP Service* preslika naprave na *UPnP* omrežje k registraciji storitve (*Service Registry*). Alternativno lahko preslika tudi OSGi storitve na *UPnP* omrežje.
- ***Jini Service*** – je omrežni protokol, ki je uporaben za odkrivanje Jini storitev na omrežju in njihov prenos v gostitelja v obliki Java kode, ter končno izvajanje teh.

Raznovrstne (*Miscellaneous*) storitve:

Ponavadi svežnji vzpostavljajo pravila za iskanje storitev/servisov, ki jih potrebujejo za delo. V veliko primerih pa mora biti to odločitev ob načrtovanju, instalaciji, oz. konfiguraciji.

- ***Wire Admin Service*** –povezuje skupaj različne servise, kot je definirano v nastavitveni datoteki. Uporabljen je koncept servisa uporabnika in proizvajalca, ki izmenjuje objekte preko žice.
- ***XML Parser Service*** –omogoča svežnjem učinkovito delo s prevajalnikom z zelenimi lastnostmi in združljivostjo z JAXP.

7 UPORABA STANDARDOV V PRAKSI

Zdaj, ko smo spoznali izrazoslovje in pomembnejše standarde s področja hišne avtomatizacije, si lahko pogledamo še različne možnosti uporabe teh v praksi.

Kot je bilo v prejšnjih poglavjih že večkrat omenjeno, je bistvo odprtega sistema možnost priključitve različnih povezljivih naprav različnih proizvajalcev, ki uporabljajo različne medije za komunikacijo, v en sam sistem in to po možnosti brez strokovne pomoči usposobljenih instalaterjev – način **priključi in uporablja**.

Pri nastajanju Konnex standarda, so imeli v mislih predvsem izpolnjevanje zgoraj omenjenih zahtev. Zaradi tega je nastajanje tega standarda dolgotrajen in zapleten postopek, ki še vedno ni zaključen.

LonWorks tehnologija je v uporabi že 17 let. Nastala je iz postavitve povsem lastniškega sistema in za postavitev takšnega sistema je potrebna pomoč strokovno usposobljenega instalaterja s potrebno opremo. Ker pa se zadnja leta kaže pomembnost uporabe odprtih sistemov, so tudi pri tem standardu pričeli z ustvarjanjem pogojev za '*priključi in uporablja*' način uporabe naprav.

Vsak sistem oz. omrežje v stavbi ima kljub vsemu svoje individualne lastnosti, ki jih je potrebno upoštevati pri njegovi izgradnji oz. že prej pri samem načrtovanju. Najprej je potrebno odgovoriti na par pomembnih vprašanj.

Vprašanja glede omrežja:

- **V kakšni zgradbi oz. stavbi se bo omrežje uporabljalo?** (nova izgradnja, obnova objekta, stara stavba...)
- **Vrste funkcionalnosti in za kakšen namen?** (zasebna uporaba, profesionalna uporaba...)
- **Katere naprave bodo povezane v sistem?** (senzorji, kamere, motorji, industrijski stroji, hišne naprave....)
- **Kateri medij bo uporabljen za povezavo naprav?** (zvita parica, optični kabel, RF, električno napajalno omrežje, IR...)
- **Kateri komunikacijski standard(i) bo uporabljen(i)?** (LonWorks, EHS/KNX, X10, CAN, EIB, WLAN, TCP/IP....)

Vprašanja glede upravljanja omrežja:

- **Kakšen hišni strežnik bo uporabljen?** (PC, posebna izvedba, na Windows platformi, na OSGi platformi, ...)
- **Kje in kaj bo uporabljeno za centralno oz. pomožne upravljalne enote za upravljanje?** (TV, mobilni telefon, poseben vgrajen LCD zaslon, dlančnik, ...)
- **Kakšen bo grafični uporabniški vmesnik (GUI)?** (standarden, WIN Media Center, CECED, OSGi...)
- **Ali bo sistem dostopen tudi navzven (npr. preko svetovnega spleta)?**
- **Če bo - Kdo bo ponudnik infrastrukture in storitev (sistem agregator)?** (Telekom, Siol...)

Upoštevati je potrebno seveda tudi stroške za izgradnjo oz. postavitev sistema, kar seveda pomembno vpliva na večino odgovorov na zgoraj postavljena vprašanja. Trenutno je za komunikacijski medij še vedno najcenejša uporaba zvite parice. Seveda pa je uporaba tega medija primerna le za novogradnje ali pri temeljitih obnovah stavb. V že obstoječih stavbah pa je bolj primerna uporaba vedno bolj popularne komunikacije preko električnega napajalnega omrežja ali brezžične komunikacije. Vendar ima velika večina ljudi do slednje odpor zaradi strahu pred prevelikim elektromagnetnim onesnaženjem bivalnega okolja.

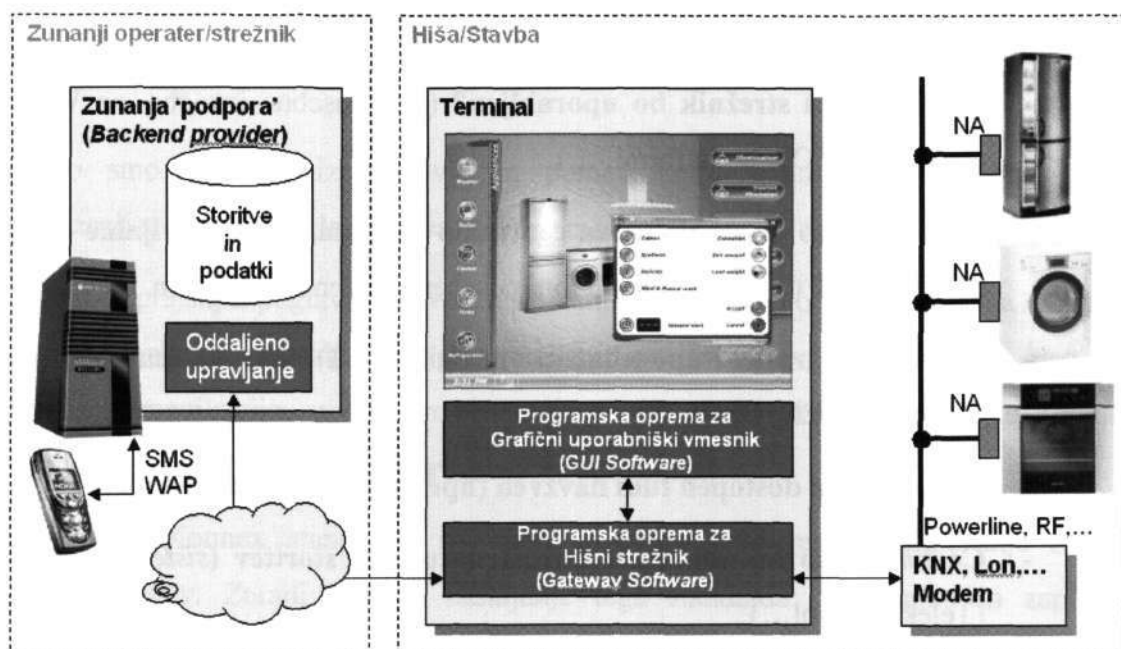
Ko odgovorimo na vsa zgoraj zastavljena vprašanja, lahko pričnemo z načrtovanjem omrežja oz. sistema. V naslednjih poglavjih si pogledjmo nekaj teh možnosti.

7.1 Sestave različnih sistemov

Najbolj na obliko sistema vpliva izbira hišnega strežnika - odvisno od sposobnosti, ki jih ima. Več inteligence, kot je premore, manj je potrebno zunanjih komponent.

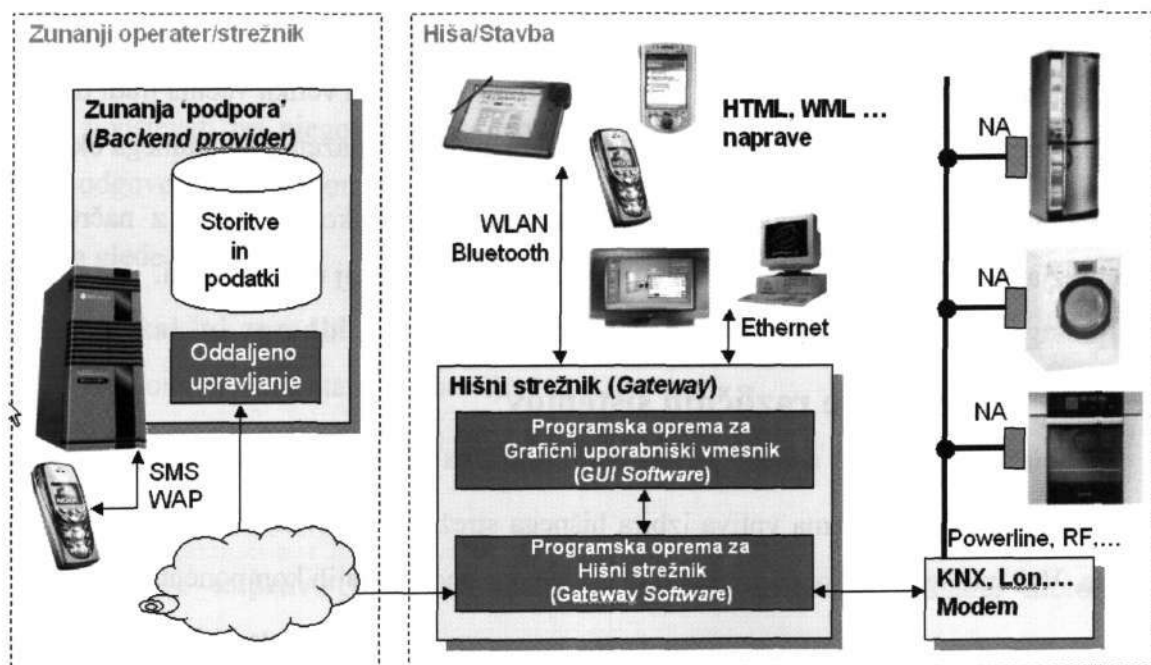
Poglejmo si nekaj možnih scenarijev [15]:

1. **Hišni terminal** - multimedijski terminal z možnostjo daljinskega upravljanja naprav v stavbi (Slika 7.1).



Slika 7.1: Scenarij uporabe sistema s hišnim terminalom

2. **Hišni inteligentni strežnik** - popolni nadzor stavbe s podporo za različne prikazovalnike v sami stavbi. Zunanji operater je potreben le za uporabo storitev (Slika 7.2).



Slika 7.2: Scenarij uporabe sistema z inteligentnim hišnim strežnikom

3. **'Osiromašen' hišni strežnik** - strežnik z minimalno inteligenco. To je poceni izvedba, ki le posreduje informacije naprej do zunanjega strežnika, kjer se nahaja vsa inteligenca sistema (Slika 7.3)



Slika 7.3: Scenarij uporabe sistema z 'osiromašenim' hišnim strežnikom

7.2 Izvedbe omrežnih vmesnikov povezljivih naprav

V tem poglavju si oglejmo, kako izgleda izgradnja oz. razvoj omrežne naprave.

Prvi pogoj je seveda, da je naprava elektronsko upravljana. Da postane naprava omrežno povezljiva, pa je seveda potrebno razviti nekakšen (fizičen) povezovalni element med elektronskim upravljanjem naprave in omrežjem. Ta povezovalni element pogosto imenujemo omrežni vmesnik oz. *Network Adapter (NA)* – od tu kratica NA, ki jo uporabljam v nadaljevanju. Ker sta za hišne gospodinjske naprave trenutno najprimernejša standarda LonWorks in Konnex, ter uporaba električnega napajalnega omrežja za komunikacijski medij, si najprej pogledjmo lastnosti uporabe tega komunikacijskega medija in obe izvedbi NA.

7.2.1 'Power Line' komunikacija

'Power line' komunikacija [6] uporablja obstoječe električno napajalno ožičenje oz. omrežje znotraj doma, stavbe ali po potrebi tudi širšega zunanjšega območja (distributerji električne energije), za prenos podatkov med napravami. Z dobro načrtovanimi rešitvami za uporabo te komunikacije lahko naprave uporabljajo obstoječe omrežje brez kakršnihkoli modifikacij (dobra in poceni rešitev za obstoječe domove).

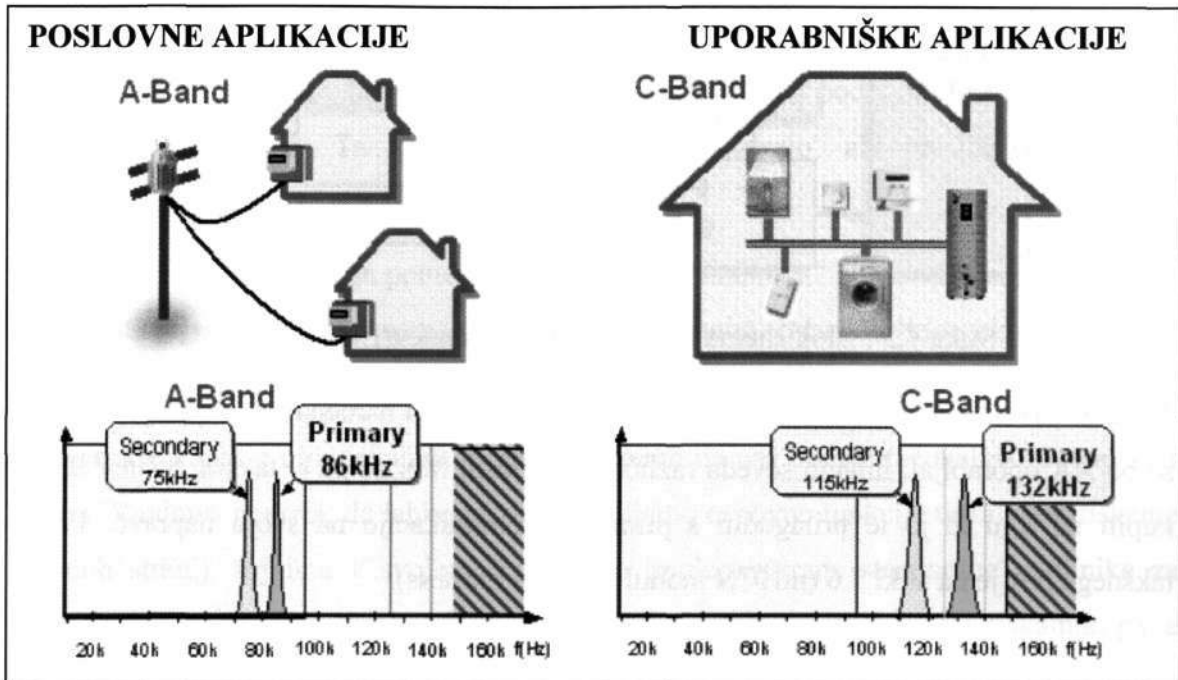
Pri razvoju takšne komunikacije pa je bilo kar nekaj izzivov. Električno omrežje je bilo postavljeno seveda za napajanje naprav z električno energijo in ne za prenos podatkov – to pomeni, da je na takšnem omrežju zelo velika količina šumov in je za uspešen prenos sporočil potreben zelo dovršen transceiver.

Na kvaliteto prenosa podatkov vpliva količina in vrsta na omrežje priključenih električnih naprav (TV-ji, računalniki, gospodinjske naprave, sušilniki las,...) v nekem trenutku. Kvaliteta je odvisna tudi od razdalje (dolžina žic) med sprejemniki in oddajniki ter od topologije omrežja (arhitektura ožičenja) v stavbi.

Pri načrtovanju uporabe 'Power line' komunikacije je potrebno upoštevati **predpisan frekvenčni spekter**. Ta mora biti skladen s pravili za posamezno področje (državo). To je še posebej pomembno, če želimo zagotoviti skupno komunikacijsko platformo za naprave, ki se tržijo na svetovnem trgu. Evropa ima za to področje že predpisane stroge zakone (CENELEC), ostale države (Severna Amerika, Azija, Afrika in Avstralija) pa si prizadevajo k upoštevanju podobnih pravil.

Frekvenčni pas na električnem omrežju, predpisan za uporabo komunikacije, je v Evropi omejen med 9kHz in 148.5kHz (slika 7.4). Ta spekter je naprej razdeljen na območja (*bands*) za specifične aplikacije:

- A-band: 9-95 kHz za dobavitelje električne energije.
- B-band: 95-125 kHz za potrošniško uporabo brez protokolov.
- C-band: 125-140 kHz za potrošniško uporabo z CENELEC protokolom.
- D-band: 140-148.5 kHz za potrošniško uporabo brez protokolov.

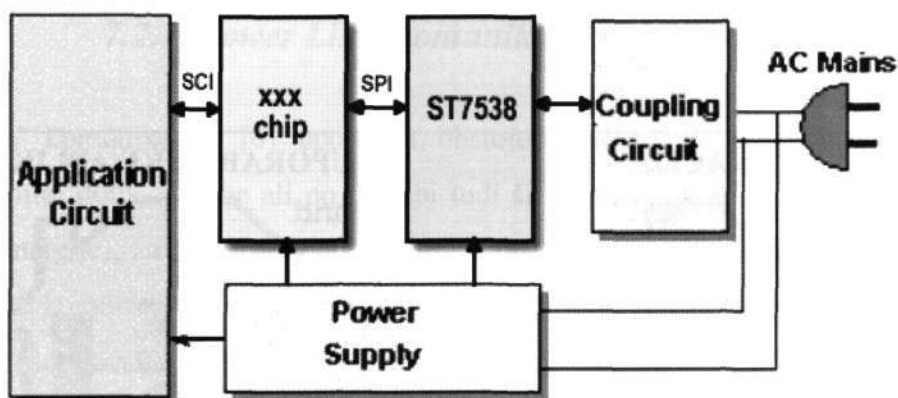


Slika 7.4 : Izbira frekvenčnega pasu

Ostale posebnosti in predpise glede komunikacije po električnem omrežju je mogoče najti v standardu CENELEC EN50065.

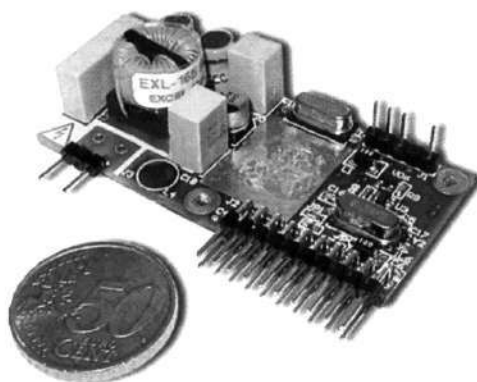
7.2.2 Konnex (EHS) omrežni vmesnik

Osnovna zgradba omrežnega vmesnika (v nadaljevanju NA) je prikazana na sliki 7.5. Zadostuje uporaba 8-bitnega mikrokontrolerja, ki omogoča SPI in SCI komunikacijo, ter dovolj pomnilniškega prostora za implementacijo Konnex (ali EHS) komunikacijskega protokola (min. 24k). SPI komunikacija je potrebna do 'power line' transceiverja. SCI (lahko je uporabljena tudi npr. I2C, UART,...) komunikacija pa je primerna za izmenjavo podatkov z upravljavskim delom elektronike v napravi, če je za to uporabljen poseben mikrokontroler (kot je prikazano v primeru na sliki 7.5). Dodati je seveda potrebno še sklopno vezje (*Coupling Circuit*), ter napajalni del (*Power Supply*).



Slika 7.5 : Blok shema NA za uporabo Konnex(EHS) standarda

To je le ena od možnosti razvoja takšnega NA. Odvisno od namena in aplikacije za katero se bo NA uporabljal, imamo seveda različne možnosti. Možno pa je takšno rešitev že tudi kupiti na trgu in jo le prilagoditi s primerno komunikacijo na svojo napravo. Primer takšnega NA je na sliki 7.6 (mHCN modul podjetja Emness).



Slika 7.6: Primer NA za Konnex/EHS komunikacijo

Pri implementaciji programske opreme imamo zopet različne možnosti. Prva in najzahtevnejša je kompletna implementacija Konnex oz. EHS komunikacijskega standarda. To pomeni popolno razumevanje vseh nivojev komunikacije in celotne arhitekture Konnex omrežja ter komunikacijskega protokola. Ta standard je vse prej kot enostaven, zato je za takšno rešitev potreben precejšen razvojni čas, ki zahteva svoje stroške – rezultat pa je lasten produkt z nižjo končno ceno.

Druga možnost je nakup programskega paketa za Konnex/EHS. Ta vsebuje že pripravljene knjižnice za ta standard. Trenutno takšne rešitve ponujajo podjetja kot so Emness (Bolgarija), Prosyst (Nemčija), Trialog (Francija), Sipro (Italija),... Seveda je v teh primerih potrebno uporabiti pomoč teh podjetij pri sami implementaciji, kar ni poceni in na

koncu plačati tudi licenčnino za vsak produkt, ki vsebuje programski paket (*stack*) tega podjetja.

Če želimo razviti CECED kompatibilno napravo, je seveda potrebno implementirati še CECED objekte. To so v bistvu točno oblikovana sporočila s podatki, ki so predpisani po CECED specifikaciji. Ta sporočila se nato 'zapakirajo' ali preslikajo v izbran komunikacijski protokol – v tem primeru Konnex.

Kaj CECED implementacija pomeni za končnega uporabnika? Na to vprašanje je najlažje odgovoriti s primerjavo z nam vsem dobro poznanim računalniškim okoljem, ki ga vsakodnevno uporabljamo.

Primer: Če kupimo nov tiskalnik in ga priključimo na naš domač računalnik, operacijski sistem Windows poskrbi, da lahko takoj uporabljamo osnovne funkcije tiskalnika (tiskanje izbranih strani), saj ima v svojem sistemu že implementirane standardne gonilnike za večino naprav. Če želimo uporabljati dodatne funkcije, ki jih omogoča nov tiskalnik, pa je ponavadi priložen še instalacijski CD z gonilniki in aplikacijami, ki omogočajo uporabo ostalih funkcij (izbira barv, dvostransko tiskanje, povečave,...).

Točno to zagotavlja tudi uporaba standardnih CECED sporočil – uporabo osnovnih funkcij npr. pralnega stoja (*start, stop, pause*). Če želimo uporabljati še ostale funkcije naprave (novi pralni programi, trenutno stanje,...) je potrebno ravno tako namestiti še dodatne gonilnike za napravo (sveženj).

Vključevanje takšnih naprav v omrežje je po zaslugi prej omenjenega kompleksnega komunikacijskega protokola zelo enostavno, saj Konnex podpira 'prikluči in uporablaj' način in zaradi tega ne zahteva posebne opreme in izkušenih instalaterjev. Takšna naprava se torej v omrežje povsem samostojno 'zapiše'.

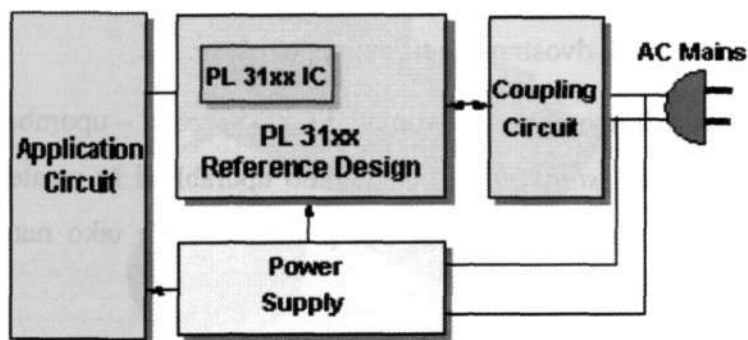
Postopek (*enrolment*) poteka takole:

1. Napravo priključimo v omrežje s priključitvijo na električno omrežje.
2. Vse kar je potrebno narediti s strani uporabnika je, da v določenem času pritisne na za to namenjen gumb na napravi in gumb na hišnem strežniku (če smo natančnejši, je to – *Plug, Press & Play* metoda)

3. Za oba (strežnik in napravo) pomeni pritisk določenega gumba znak za inicializacijsko fazo – '*Enrolment*' naprave. Po točno določenem postopku strežnik dodeli zahtevane naslove novi napravi.
4. Naprava je tako povezana v omrežje in ima svoj naslov preko katerega je dosegljiva.

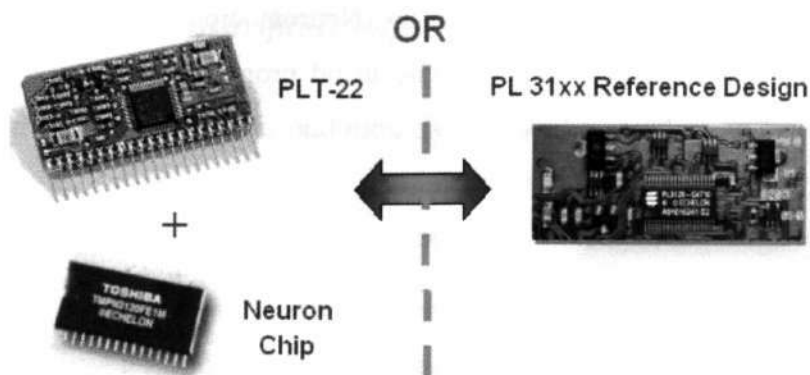
7.2.3 LonWorks omrežni vmesnik

Osnovna zgradba omrežnega vmesnika (v nadaljevanju NA) je prikazana na sliki 7.7. Uporabljen je mikrokrmilnik PL31xx, ki že vsebuje *Neuron* procesor in '*Power line*' transceiver. Torej je bilo potrebno dodati le še sklopno vezje (*Coupling Circuit*), napajalni del (*Power Supply*), ter povezava do elektronskega upravljanja naprave (*Application Circuit*).



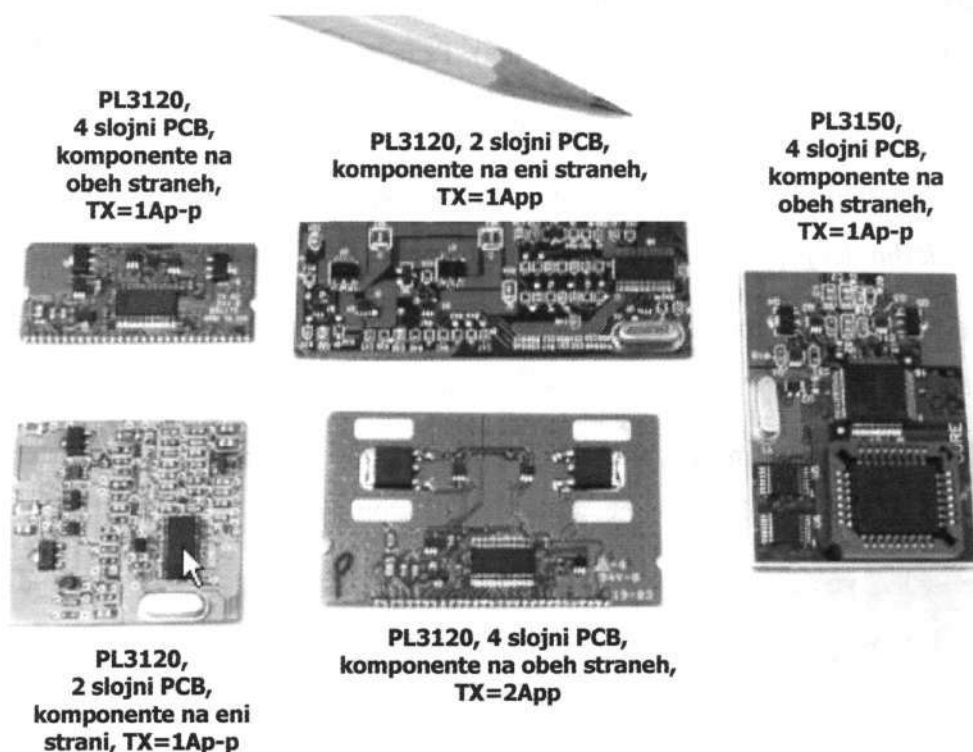
Slika 7.7 : Blok shema NA za uporabo LonWorks tehnologije

Pred obstojem mikrokontrolerja PL31xx je bilo potrebno v vezje dodati primeren *Neuron* procesor in '*Power line*' transceiver (PLT-22), kar je poleg visoke cene pomenilo tudi večjo končno velikost produkta, kar je lepo prikazano na sliki 7.8.



Slika 7.8: Primerjava stare in nove izvedbe NA

K razvojnemu orodju za delo z PL31xx spada tudi dokumentacija, ki vsebuje priporočljive referenčne načrte vezja ter priporočljive elektronske komponente za uporabo. Nekaj teh je prikazanih na sliki 7.9. Najlažji razvoj Lon NA je torej, če enostavno sledimo vsem priporočilom, saj na ta način podjetje Echelon zagotavlja tudi uspešno certificiranje LonWorks naprave pri LonMark organizaciji.



Slika 7.9: Referenčni načrti za uporabo PL31xx

Po izdelavi prototipa Lon NA je potrebno pričeti z razvojem programske opreme. Neuron čip že vsebuje *firmware* za komunikacijo po električnem omrežju z uporabo LonTalk protokola. Dodati je potrebno le vmesnik med delom elektronike, ki upravlja napravo in

NA. Programski jezik, ki se uporablja na Neuron čipu, se imenuje *Neuron C*. Programiranje v tem jeziku je zelo enostavno in od programerja ne zahteva natančnega poznavanja delovanja mikrokrmilnika. Ni se potrebno ubadati z nastavljanjem registrov, števecv in prekinitev.

Primer: Za občutek podajam vrstico, ki je v programskem jeziku Neuron C potrebna za uporabo oz. nastavitve serijske (UART) komunikacije mikrokrmilnika [10]:

```
IO_8 sci [baud (SCI_2400)] [twostopbits] io-object-name;
```

Na takšen način zagotovimo komunikacijo med upravljavsko elektroniko in komunikacijskim delom elektronike in s tem že imamo povezljivo napravo.

Naslednji korak pri razvoju je dodajanje te naprave v omrežje. LonWorks tehnologija potrebuje za ta namen posebno orodje (strojna oprema: *NodeBuilder* in programska oprema: *LNS*, *LonMaker*...). To orodje priključimo na omrežje in z njim lahko zaznamo vse naprave, ki za komunikacijo uporabljajo LonTalk protokol. Vsaka LonWorks naprava ima svoj edinstven ID (identifikacijska številka). Ta je zapisan v vsakem Neuron čipu.

Vključevanje naprave v omrežje izgleda takole:

1. v primernem programskem okolju (*LonMaker*) preko primerne strojne opreme (*NodeBuilder*) sprožimo poizvedovanje na omrežje.
2. Na LonWorks napravi stisnemo gumb (*service pin*), ki v omrežje pošlje svoj unikaten ID
3. *NodeBuilder* sprejme podatke in jih posreduje programski opremi, s katero lahko zaznani napravi določimo ime, mesto in funkcionalnosti v omrežju.

Ker pa je vedno več zahtev po rešitvah 'priključi in uporablaj', že obstajajo tudi takšne rešitve za Lonworks naprave. Prvi koraki v to smer so bili narejeni tudi pri CECED organizaciji, ki je že izdala dokument za preslikavo CECED sporočil na LonTalk protokol (*CECED LON Mapping*).

7.2.4 Upravljanje naprav in sistema

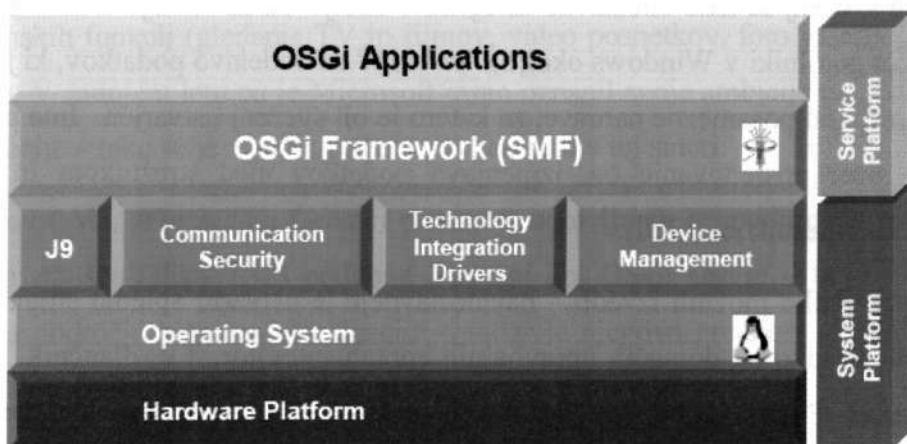
Naprave po izbranem komunikacijskem mediju (žice, RF...) povežemo v omrežje. Kot že omenjeno, je v takšen sistem priključen tudi hišni strežnik, ki omogoča upravljanje naprav in uporabo ostale funkcionalnosti sistema (odvisno od vrste uporabljenega strežnika).

Zgradba hišnega strežnika je v bistvu zelo podobna zgradbi osebne računalnika. Razlika med njima je predvsem zaradi namena uporabe: osebni računalnik uporabljamo za vse - od službenega dela, do igranja igrice, gledanja filmov, programiranja,...ipd. Nanj priključujemo nove in nove komponente in malo je takšnih, ki se jim pod vsemogočimi obremenitvami ne bi kdaj pa kdaj 'sesulo' programsko okolje. Tega si pri hišnem strežniku enostavno ne moremo privoščiti. Delovati mora 24 ur na dan in 7 dni v tednu – zanesljivo in brezhibno. To je še posebej pomembno, če je v omrežje vključen npr. varnostni sistem stavbe. Trenutno se na tem področju kažeta dve zanimivi rešitvi *OSGi* in *.NET Framework*.

7.2.4.1 Hišni strežnik na osnovi *OSGi* ogrodja

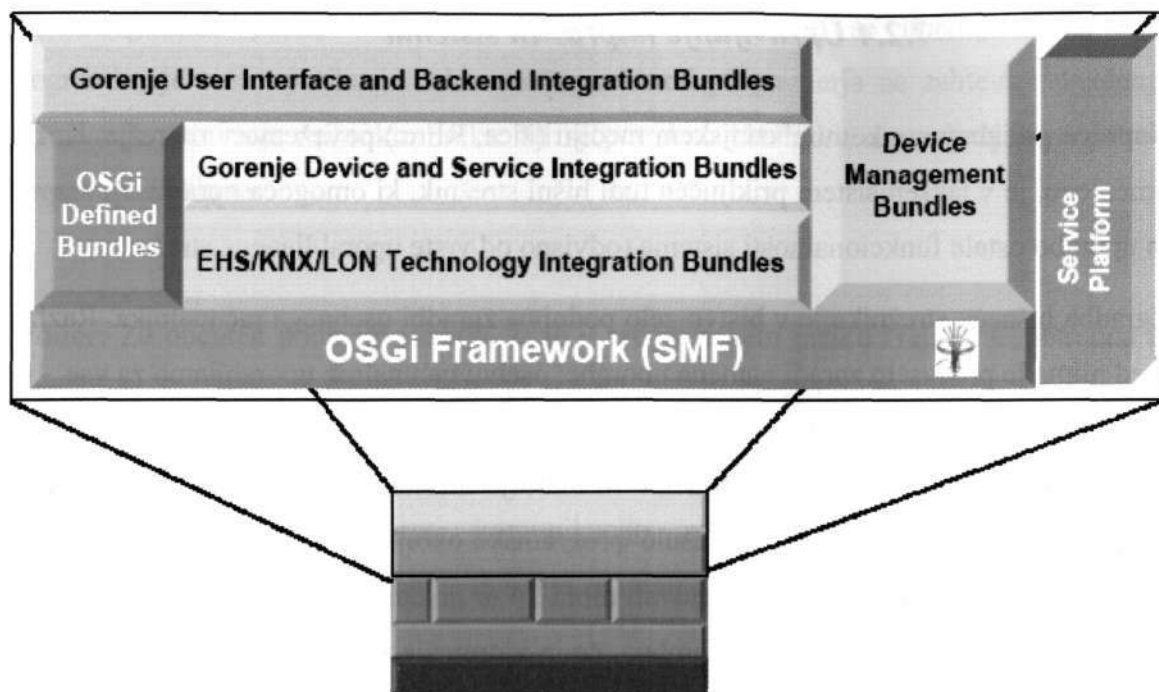
Kot je opisano v poglavju *OSGi*, za tem standardom stojijo velika svetovna podjetja [14], ki ustvarjajo tehnično dovršeno programsko okolje, ki bo omogočalo interoperabilnost različnih standardov in komunikacijskih medijev med različnimi proizvajalci naprav.

Na sliki 7.10 prikazuje osnovno blokovno zgradbo programskega okolja za *OSGi* hišni strežnik.



Slika 7.10: Zgradba *OSGi* hišnega strežnika

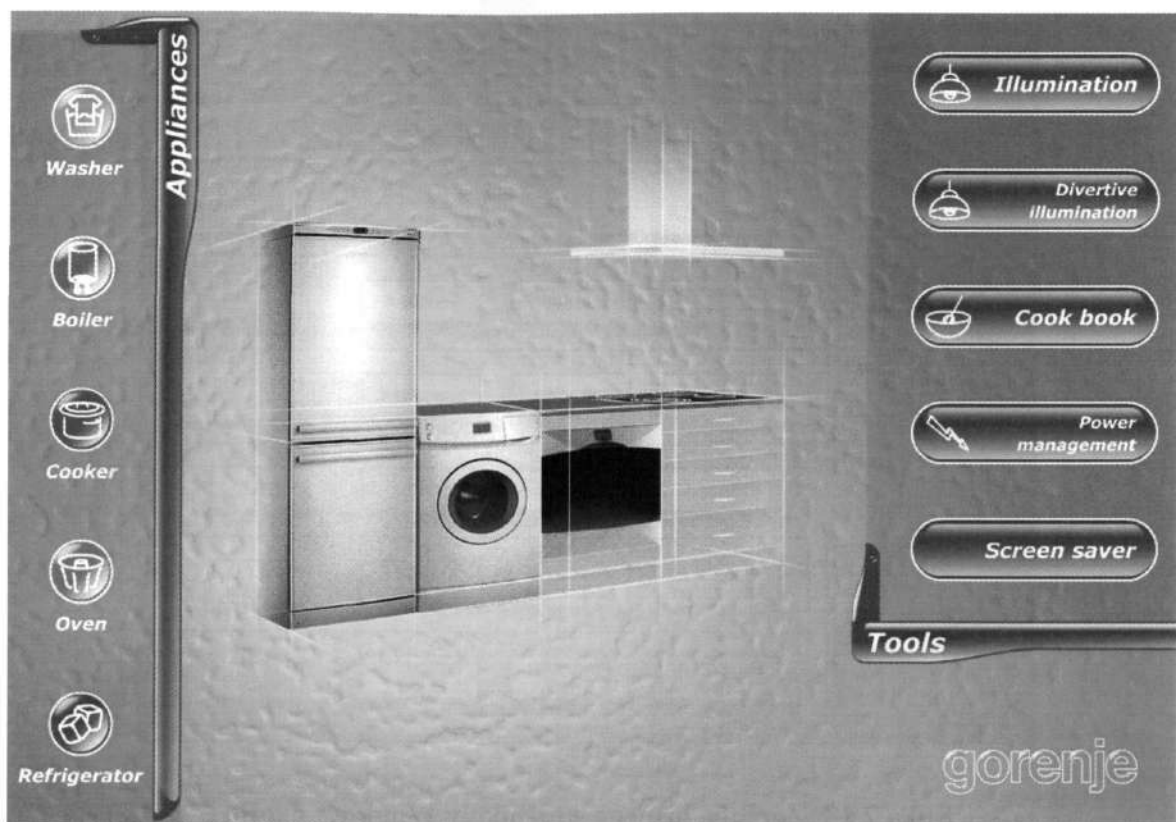
Slika 7.11 natančneje razdrobi del storitvene platforme *OSGi* okvirja.



Slika 7.11: Zgradba storitvene platforme OSGi okvirja

Hišni strežnik je v bistvu končna naprava, kateri mora proizvajalec povezljivih naprav le zagotoviti ustrezen 'paket' programske opreme, ki jih želi priključiti na ta strežnik. V primeru OSGi predstavlja ta paket programske opreme tako imenovan *OSGi sveženj*. To je aplikacija napisana v programskem jeziku Java. Struktura svežnja je točno definirana. V objekte znotraj njega mora proizvajalec zapisati lastnosti in obnašanje svoje naprave v omrežju. Ta sveženj se nato enostavno integrira v okolje OSGi okvirja v hišnem strežniku (podobno kot gonilniki v Windows okolju), ter 'skrbi' za obdelavo podatkov, ki jih strežnik dobi iz omrežja od posamezne naprave, za katero je bil sveženj ustvarjen. Ima pa še drugo nalogo, in sicer posredovanje oz. izmenjavo podatkov med strežnikom in grafičnim uporabniškim vmesnikom (GUI).

Tukaj je spet veliko možnih izvedb – najenostavneje je ustvariti spletno stran, ki je nato lahko uporabna tako na domačih upravljalnih enotah (monitor, TV, dlančnik...) kot tudi preko interneta. Primer take strani, ki smo jo v Gorenju ustvarili za upravljanje našega demonstracijskega sistema, je na sliki 7.12.

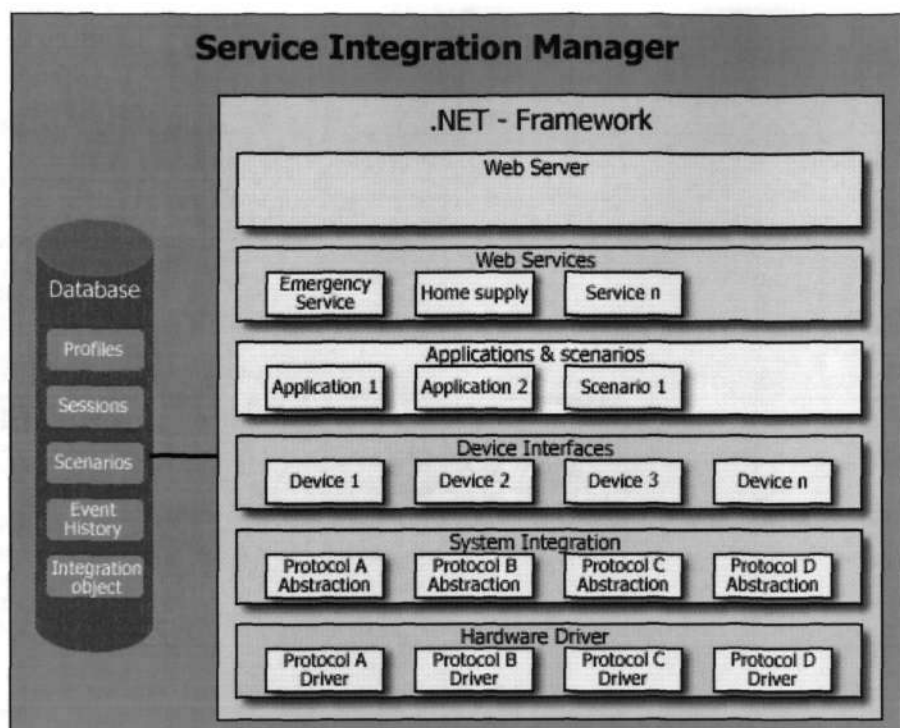


Slika 7.12: Primer GUI za upravljanje naprav v omrežju

7.2.4.2 Hišni strežnik na osnovi .Net Framework-a

Windows Media Center je bil ustvarjen pred nekaj leti, vendar šele v zadnjih mesecih prihaja na trg v večjem zamahu. Ustvarjen je bil predvsem za namene uporabe različnih multimedijskih funkcij (gledanje TV in filmov, video posnetkov, foto galerije, poslušanje glasbe...). V zadnjem letu pa je Microsoft jasno oznanil svoje ambicije na področju hišne avtomatizacije – tako se je zgodba pričela odvijati tudi v tej smeri.

Microsoftov (.Net, *Win Media Center*) pristop k upravljanju gospodinjskih naprav ima s tehničnega stališča (slika 7.13) podobne lastnosti kot OSGi sistem – ta je mogoče bolj dovršen za področje hišne avtomatizacije, vendar je njegova prednost v tem, da je med končnimi uporabniki operacijski sistem Windows zelo razširjen in so ljudje že navajeni uporabe tega okolja. Zaradi tega obstaja velika verjetnost, da se bo ta rešitev prej 'prijela' na trgu.



Slika 7.13: Zgradba hišnega strežnika v Windows okolju

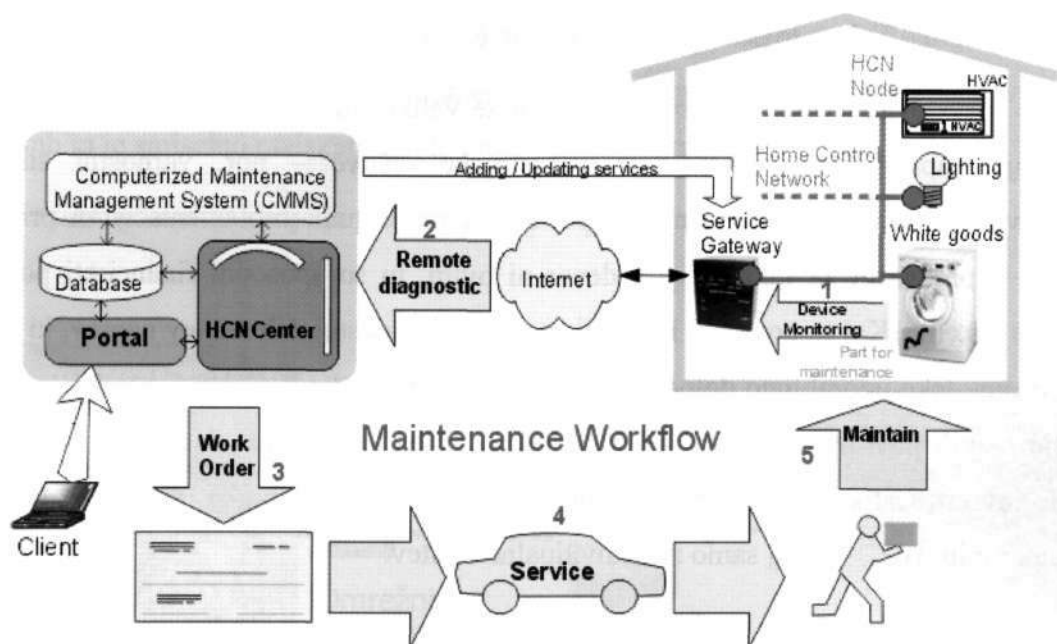
Ima pa še eno prednost in sicer že ustvarjen skupni GUI – grafični uporabniški vmesnik (*Win Media Center*), v katerega je zelo enostavno dodati nove stvari, izgled pa ostane enoten. (slika 7.14). Najenostavneje je dodati kar HTML spletno stran, ki se enostavno integrira v okolje. Izmenjava podatkov pa lahko poteka npr. v obliki XML sporočil.



Slika 7.14: GUI - Windows Media Center

7.2.5 Primer uporabe storitve vzdrževanja

Poglejmo si še možen mehanizem za zagotavljanje storitev v avtomatiziranem domu [14].



Slika 7.15: Primer storitve servisiranja naprave

Na primeru so prikazani vsi nivoji komunikacije od naprave do ponudnika storitve (Slika 7.15). Koraki postopka si lahko sledijo nekako takole:

1. Gospodinjaska naprava, ki je zaznala okvaro, obvesti hišni strežnik z natančno definiranim sporočilom, ki opisuje problem.
2. Hišni strežnik se poveže z zunanjim storitvenim strežnikom (na sliki CMMS) in mu posreduje sporočilo naprave.
3. CMMS avtomatsko izda delovni nalog servisnemu podjetju, z vsemi potrebnimi informacijami o okvari, lokaciji ter možni rešitvi okvare
4. Serviser opravi servisni poseg v enem zamahu, saj točno pozna situacijo že prej, preden pride do stranke.

To je le ena izmed možnih storitev. Na tem primeru je lepo vidno, kakšne koristi prinaša sistem povezanih naprav, saj se v tem primeru stroški in čas servisiranja zelo zmanjšajo.

7.3 Primerjava lastniško usmerjenega sistema z odprtim

Skozi poglavja naloge so nekajkrat omenjene lastniške rešitve v primerjavi z rešitvami odprtega sistema. Tokrat le povzemimo ugotovitve.

Gledano v svetovnem merilu, je bilo narejenih že veliko implementacij avtomatizacije in omreženja naprav. To so predvsem industrijske rešitve – npr. varnostni sistemi, nadzorovanje strojev v proizvodnjah...ipd. Nekaj pa je tudi implementacij za privatne domove – predvsem za ljudi, ki jim denar ni ovira, in so sposobni financirati povsem lastniške sisteme. Kaj to pomeni? Poiskali so npr. nekaj obstoječih rešitev/naprav, ki so na trgu, ki prav tako uporabljajo določene komunikacijske standarde (X10, CAN, EIB....), ali pa celo razvili povsem svoj protokol za komunikacijo. Te so nato povezali v omrežje in razvili povsem lastniške načine upravljanja glede na želje uporabnika (npr. PC s povsem nestandardnim GUI) – torej samo za individualno rešitev.

Slabosti takšnih sistemov so predvsem:

- drag razvoj in implementacija
- precejšen čas od želje po sistemu do dejanske realizacije
- popolna odvisnost od razvijalca, tako pri servisiranju kot pri morebitnih spremembah – velik strošek

Ravno zaradi teh naštetih slabosti, se je pričel razvoj odprtega sistema. Ta torej omogoča sestavo standardnih naprav v standardizirana omrežja po principu 'prikluči in uporablaj'.

Prav tako pa bo s tem omogočena masovna proizvodnja, kar drastično vpliva na znižanje cen izdelkov in končnih stroškov pri implementaciji sistema.

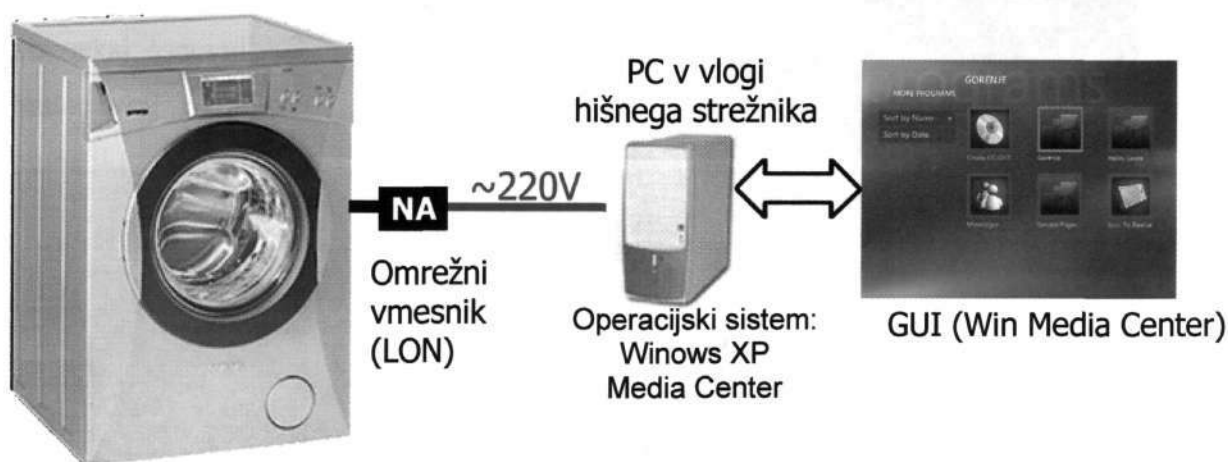
7.4 Izdelava demo primera

Kot že rečeno, smo se tudi v Gorenju odločili za razvoj povezljivih naprav za vključitev v odprte sisteme. Kot vodja projekta razvoja teh aparatov moram pokrivati in razumeti vse faze razvoja. Projekt ima močne poslovne partnerje (IBM, Echelon...), pričel pa se je v mesecu aprilu 2004. Ker je to zelo zahteven in obširen projekt, bodo prvi prototipi aparatov (pralni stroj, hladilnik in štedilnik) z implementirano LonWorks in Konnex

komunikacijo, ter s sistemsko podporo za OSGi hišni strežnik dokončani šele do konca letošnjega leta.

Vzporedno s tem projektom, delam tudi na alternativni novi rešitvi (*Win Media Center*), ki je primerna za promocijo povezljivosti. Vzrok za to delo je tudi v tem, da se je primernost uporabe Microsoftovega okolja v namene hišne avtomatizacije pokazala šele v zadnjih mesecih in je smiselno raziskati tudi te možnosti za uporabo.

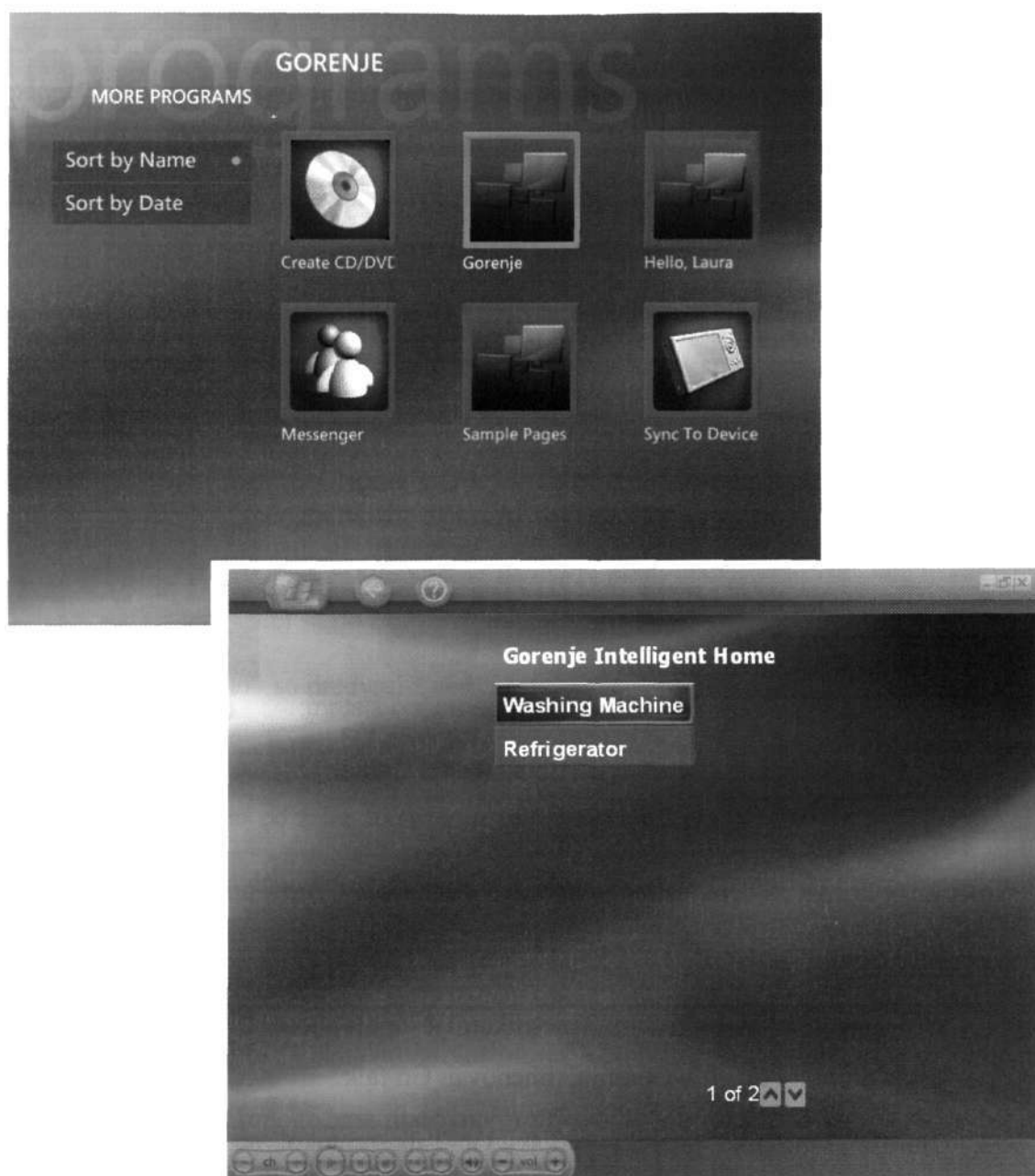
Primer takšne komunikacije je prikazan na sliki 7.16 in izgleda takole:



Slika 7.16: Primer za demonstracijo uporabe *Win Media Centra* pri upravljanju naprav

Posamezni gradniki so:

1. Med aparatom in omrežnim vmesnikom je uporabljena UART komunikacija in lasten protokol razvit v Gorenju, imenovan Go-talk.
2. Omrežni vmesnik preslika serijski Go-Talk protokol na *LonTalk* protokol za komunikacijo po električnem omrežju.
3. PC je za namen komunikacije preko *SLTA-10* adapterja (Echelon) serijsko povezan na električno omrežje.
4. Na PC-ju je uporabljen operacijski sistem *Windows XP Media Center*.
5. V GUI okolje *Media Centra* je implementirana HTML stran za upravljanje aparatov Gorenje (slika 7.17).



Slika 7.17: Primer GUI za upravljanje aparatov Gorenje

8 SKLEP

Področje hišne avtomatizacije je zelo atraktivno že vrsto let, vendar se pogoji za njen razcvet šele ustvarjajo. Precej stvari je že dorečenih, dosti standardov se šele ustvarja in iz celotne mase teh, so se pričele luščiti prave in uporabne rešitve za to področje.

V tej magistrski nalogi sem želela ustvariti bazo podatkov za nekoga, ki se na novo spušča v to področje. Ker je samo področje zelo obširno, se je temu 'prilagodil' tudi obseg naloge. Opisani so le standardi, ki so v tem trenutku najpomembnejši za povezljive hišne naprave, ter nekaj osnovnih napotkov za izgradnjo celotnega sistema. Kljub vsemu mora imeti bralec vsaj nekaj predznanja iz področja programiranja, komunikacij, protokolov in omrežij.

V vsakem poglavju so zajete osnovne značilnosti, ki bralcu omogočajo kasnejše lažje razumevanje dejanskih specifikacij standardov.

9 LITERATURA

1. Konrad Steblovnik: Internetno dosegljivi gospodinjski aparati; VITEL - 16. delavnica o telekomunikacijah - Pametne stavbe; november 2004
2. <http://www.konnex.org>, ,Konnex Organisation
3. Konnex Association: KNX open standard Specification, Februar 2004
4. <http://www.konnex.org/download.html> , Documentation download
5. Echelon Corporation: Introduction to the LONWORKS System; CA. USA, l. 1999
6. <http://www.echelon.com/products/lonworks/faq.htm> - pogosta vprašanja
7. <http://echelon.com/products/oem/transceivers/smartpl/default.htm>, Power Line Smart Transceivers
8. Echelon Corporation: LON Protocol Overview; San Jose, CA. USA, l. 1994
9. <http://www.lonmark.org/about/about.htm> , LonMark Organisation
10. Echelon Corporation: Neuron C Reference Guide; San Jose, CA. USA, l. 1999
11. <http://www.ceced.org>
12. European Committee of Household Appliance Manufacturers (CECED): Household Appliances Control and Monitoring /Application Interworking Specification, verzija 0.96, Marec 2004
13. http://www.osgi.org/osgi_technology/index.asp?section=2 , OSGi Alliance
14. <http://www.emness.com>
15. <http://www.prosyst.com>

Izjava

Izjavljam, da sem magistrsko nalogo izdelala samostojno pod vodstvom mentorja prof.dr. Andreja Žemve. Izkazano pomoč drugih sodelavcev sem v celoti navedla v zahvali.

Klančnik Laura

A handwritten signature in black ink, appearing to read 'Laura Klančnik', written in a cursive style.