

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

**HMI modul pro ovládání osvětlení v budově s komunikací přes
IP/UDP/KNXIP**

**HMI Module for Lighting Control in a Building with Communications over IP
/ UDP / KNXIP**

Zadání bakalářské práce

Student: **David Barabasz**
Studijní program: B2649 Elektrotechnika
Studijní obor: 2601R004 Měřicí a řídicí technika
Téma: **HMI modul pro ovládaní osvětlení v budově s komunikací
přes IP/UDP/KNXIP
HMI Module for Lighting Control in a Building with Communications
over IP / UDP / KNXIP**

Zásady pro vypracování:

1. Seznámení se s problematikou řízení provozu budov.
2. Seznámení se s problematikou KNX.
3. Seznámení se s problematikou komunikace prostřednictvím protokolu TCP/IP (UDP/IP).
4. Návrh HMI modulu s komunikací přes IP/UDP/KNXIP/KNX.
5. Realizace navrženého HMI modulu.
6. Zhodnocení dosažených výsledků.

Seznam doporučené odborné literatury:

- [1] *Building automation: control devices and applications*. Homewood, Illinois: American Technical Publishers, Inc., c2008, 407 p. ISBN 0826920004.
- [2] MERZ, Hermann, Thomas HANSEMANN and Christof HUBNER. *Building automation*. New York: Springer, 2009, p. cm. ISBN 978-3-540-88828-4 .
- [3] KOZIEROK, Charles M. *The TCP/IP guide: a comprehensive, illustrated Internet protocols reference*. 06. vyd. San Francisco: No Starch Press, c2005, lxxiv, 1539 p. ISBN 1-59327-047-X.
- [4] FRANK, Karlheinz. *EIB: Grundlagen Gebäudesystemtechnik*. 4., aktualis. Aufl. Berlin: Huss-Medien, 2007. ISBN 978-3-341-01540-7.
- [5] HOLLIFIELD, Bill R., et al. *The High performance HMI handbook: a comprehensive guide to designing, implementing and maintaining effective HMIs for industrial plant operations*. 1st ed. Houston, TX: Plant Automation Services, 2008. ISBN 978-0977896912.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. David Vala**

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2014



doc. Ing. Jiří Koziorek, Ph.D.
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Třanovicích, dne 7. 5. 2014



.....
David Barabasz

Abstrakt

Tato bakalářská práce se zabývá HMI modulem sběrnice systému KNX. KNX je celosvětový standard pro řízení budov. Hlavním cílem bylo navrhnout a realizovat inteligentní vypínač ke spínání světla a vizualizace komunikace. Inteligentní vypínač bude obsahovat tlačítko pro zapnutí, vypnutí světla, a také displej, na kterém bude vizualizována komunikace na sběrnici. Led dioda bude signalizovat zapnutí/vypnutí světla.

Abstract

This bachelors work deals with the HMI module of the KNX bus system. KNX is the worldwide standard for building management. The main aim was to design and implement an intelligent switch for switching lights and visualization of communication. The intelligent switch will contain a button to turn lights on and off, and a display, on which communication with the bus will be visualized. The LED will indicate the on/off status of the light.

Klíčová slova

KNX, sběrnice, Ethernet, Raspberry Pi, HMI modul.

Keywords

KNX, bus, Ethernet, Raspberry Pi, HMI module.

Seznam použitých symbolů a zkratek

KNX	Celosvětový standard pro řízení provozně technických funkcí v budovách
EIB	Evropská instalační sběrnice - European Installation Bus
EIBA	European Installation Bus Association
TP	Twisted Pair (Kroucený pár)
PL	Silové vedení(Silové vedené)
RF	Rádiový přenos
IP	Internetový protokol
LAN	Lokální počítačová síť
ISO/OSI	Referenční model popisující komunikaci počítačových sítí pomocí vrstev.
U	Účastník
L	Linie
PL	Páteřová linie
HL	Hlavní linie
NZ	Napájecí zdroj
LS	Liniová spojka
OS	Oblastní spojka
LO	Liniový opakovač
EEPROM	Electrically Erasable Programmable Read-Only Memory,Elektricky mazaná paměť
ETS4	KNX software k programování, diagnostiku a návrh KNX systému.
PVC	Polyvinylchlorid
UDP	User Datagram Protocol
TCP	Transmission Control Protocol
cEMI	Format zápisu
GPIO	General-purpose input/output
router	Aktivní síťové zařízení, přeposílající datagramy směrem k cíli.
Multicast	Více směrové vysílání, jeden vysílač – více příjemců

Obsah

1.	Úvod	1
2.	Inteligentní řízení budov	2
3.	KNX (KONNEX)	3
3.1.	Topologie sítě	5
3.1.1.	Napájecí zdroj	8
3.1.2.	Liniová spojka	8
3.1.3.	Oblastní spojka	9
3.2.	Adresování	10
3.2.1.	Individuální adresa	10
3.2.2.	Skupinová adresa	10
3.2.3.	Vlajka adresy	11
3.3.	Přenosová média	11
3.3.1.	KNX.TP	11
3.3.2.	KNX.PL	12
3.3.3.	KNX/RF	12
3.4.	Průběh komunikace, druhy telegramů	12
3.5.	Datový telegram TP1	12
3.6.	Ethernet KNXnet/IP	15
3.6.1.	KNX IP Router	15
3.6.2.	KNX/IP telegram	16
4.6.3	cEMI telegram:	18
4.	Ethernet	23
4.1.	Vrstvy ISO/OSI modelu	24
4.1.1.	Fyzická vrstva	24
4.1.2.	Linková vrstva	24
4.1.3.	Síťová vrstva	24
4.1.4.	Transportní vrstva	24
4.1.5.	Relační vrstva	24
4.1.6.	Prezentační vrstva	24

4.1.7.	Aplikační vrstva	24
4.2.	TCP a UDP protokoly	25
4.2.1.	TCP protokol	25
4.2.2.	UDP protokol	25
5.	Návrh HMI modulu	26
5.1.	Raspberry Pi	26
5.1.1.	Operační systém	27
5.1.2.	GPIO	28
5.1.3.	Návrh uživatelského modulu	28
6.	Realizace HMI modulu	30
6.1.	Programování	30
6.1.1.	Knihovny	30
6.1.2.	Multicast komunikace	32
6.1.2.1.	Odesílání	32
6.1.2.2.	Přijímání	32
6.1.3.	Skládání telegramu	32
6.1.3.1.	Telegram pro zapnutí světla	33
6.1.3.2.	Telegram pro vypnutí světla	33
6.1.3.3.	Telegram pro zjištění stavu světla	34
6.1.3.4.	Odesílání telegramů	34
6.1.3.5.	Přijímání telegramů	34
6.2.	Tlačítka	35
6.3.	Displej	36
6.4.	Soubor	37
6.5.	Indikační led dioda	38
6.6.	HMI modul	38
7.	Závěr	40
	Použitá literatura	41

1. Úvod

Tato bakalářská práce se zabývá realizací inteligentního vypínače (HMI modulu) pro inteligentní ovládání osvětlení.

Inteligentní řízení budov je stále rozvíjející se odvětví poskytující větší komfort, větší bezpečí a také úsporu energií. Řízením budov se zabývá společnost KNX a prvořadým úkolem bakalářské práce bylo seznámit se s řízením inteligentních budov a KNX sběrnici. Seznámit se s topologií sítě, komunikací po sběrnici a přístroji komunikující prostřednictvím KNX sběrnice. Dále bylo třeba se seznámit s komunikací prostřednictvím Ethernetové sítě, zjištění formátu telegramu a jeho pozdější využití k realizaci vypínače.

Nezbytnou součástí bylo navrhnout HMI modul ke komunikaci a využít celosvětově rozšířenou sběrnici KNX k ovládání osvětlení. S návrhem modulu souvisí rovněž výběr mikroprocesoru disponující možnosti připojit síťovou kartu a komunikovat po Ethernetové síti. Ethernetová síť je v tomto případě využívána pro svou vysokou rychlost přenosu dat a její rozšíření.

Realizace HMI modulu vyžadovala naprogramování aplikace s funkční komunikací mezi HMI modulem připojeným k Ethernetové síti a KNX zařízením komunikujícím po KNX sběrnici. Fyzické propojení Ethernetové sítě a KNX sběrnice zajišťuje KNX IP Router. Vytvořená aplikace pro komunikaci mezi HMI modulem a KNX akčním členem spínající světlo běží na počítači Raspberry Pi. Inteligentní vypínač obsahuje vypínač ke spínání světla a také displej, na kterém probíhá vizualizace komunikace a stavu světla.

2. Inteligentní řízení budov

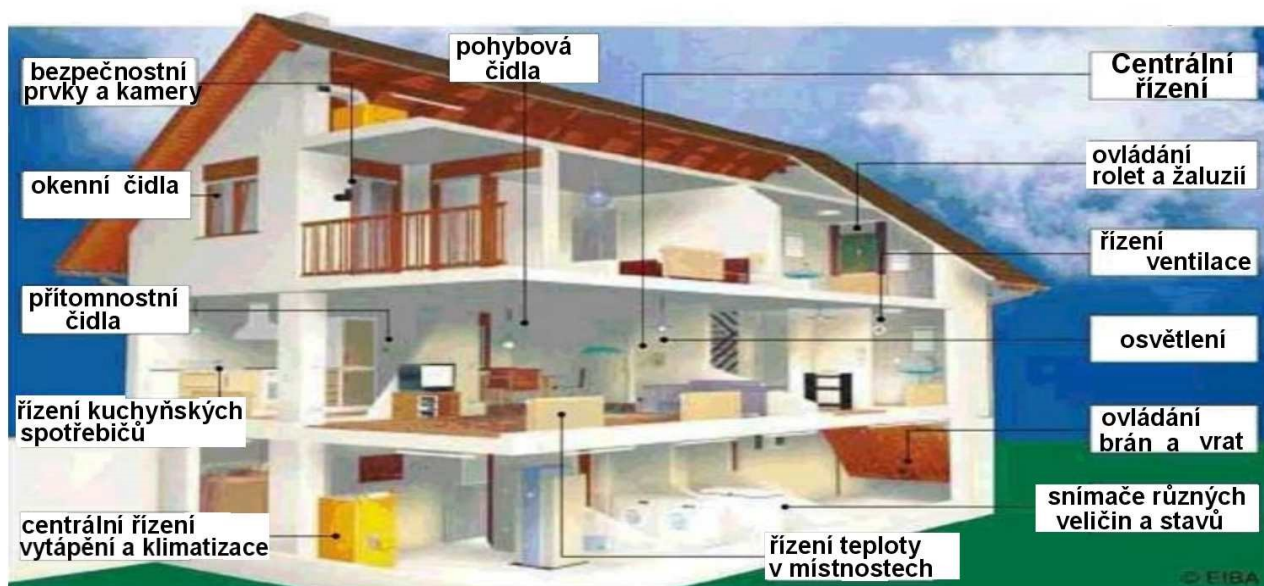
V současné průmyslově rozvinuté společnosti narůstá podíl automatizace jak v bytové zástavbě, tak i v účelových stavbách. Automatizace přináší větší pohodlí, bezpečnost a hospodárnost stavebnictví. Automatizace budov je samostatný a významný obor automatizační techniky. Automatizační prvky jsou snímače, akční členy, regulátory. Pro komunikaci mezi přístroji musí být přístroje propojeny komunikačním systémem pro oboustrannou výměnu dat. K tomuto účelu se používají sběrnice. (1)

Značný podíl automatizovaných funkcí se stává určitým standardem. Za samozřejmost můžeme požadovat regulace spotřeby energie. Optimalizace vytápění jednotlivých místností, snížení spotřeby v nočním režimu. Měření spotřeby energií a přizpůsobení větrání, vytápění povětrnostním podmínkám snižuje energetickou náročnost budovy. (1)

Dalším příkladem je řízení osvětlení. V mnoha případech se venkovní osvětlení spíná instalovaným detektorem pohybu. Světlo vybavené dalším senzorem osvětlení se sepne jen tehdy, pokud je dostatečné šero. Zapínání a vypínání osvětlení celého obytného prostoru představuje pro sběrnice systémy velkou výhodu oproti konvenční elektroinstalaci. Centrálním vypínačem je možné spustit alarm, vypnout zásuvky, spustit nebo zvednout rolety. Systém dokáže poskytnout zpětnou vazbu, informovat o stavu okna, zobrazit rozsvícená světla. Uživatel může použít k ovládání budovy chytrý telefon, tablet, a to i ze vzdálených míst přes internet. (1)

V soukromé bytové výstavbě získaly automatizované funkce velký význam především v oblastech:

- úspory energií,
- komfortu,
- a bezpečnosti. (1)



Obr. 1: Návrh inteligentního řízení budovy (2)

3. KNX (KONNEX)

KNX je celosvětový standard pro řízení provozně technických funkcí v budovách. Tvůrcem a vlastníkem této technologie je asociace KNX. Díky certifikaci produktů na základě standardu KNX je zaručena vzájemná kompatibilita výrobků různých firem, což představuje vysokou úroveň flexibility. (3)

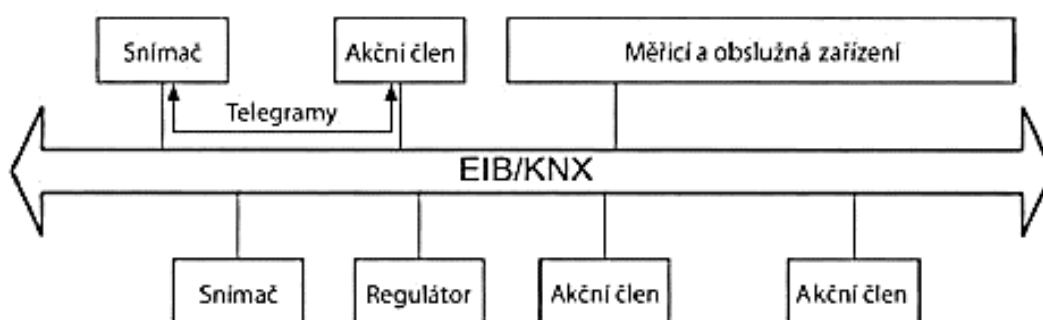


Obr. 2: Logo asociace KNX

„Instalace systémů řízení osvětlení a elektricky ovládaná instalace zastínění je klasickou oblastí elektroinstalačních profesí. Pro zajištění tohoto oboru vyvinulo mnoho evropských firem systémové komponenty, které umožňují automatický provoz funkcí tvorby prostředí.

Vůdčí výrobci elektroinstalační techniky se v rámci EIBA dohodli na vývoji vzájemně kompatibilní směrnice. Na základě této dohody vznikla Evropská instalační sběrnice EIB. Po sloučení EIBA s dalšími evropskými organizacemi do sdružení Asociace KNX se používá pro tyto systémy označení KNX/EIB. Jedná se o standardizovanou sběrnici, která zajišťuje výměnu dat mezi různými zařízeními a systémy nezávisle na výrobci.

Evropská instalační sběrnice KNX/EIB je průmyslový komunikační systém, který se v systémové technice budov používá pro síťové informatické spojení zařízení (snímačů, akčních členů, regulačních a řídicích zařízení, Obslužných a měřicích zařízení). Implementace KNX/EIB je přizpůsobena elektrotechnické instalaci, čímž jsou zajištěny funkce a automatizované procesy v budově.“ (1)



Obr. 3: Informatické zasíťování zařízení systémové techniky budov sběrnici KNX/EIB

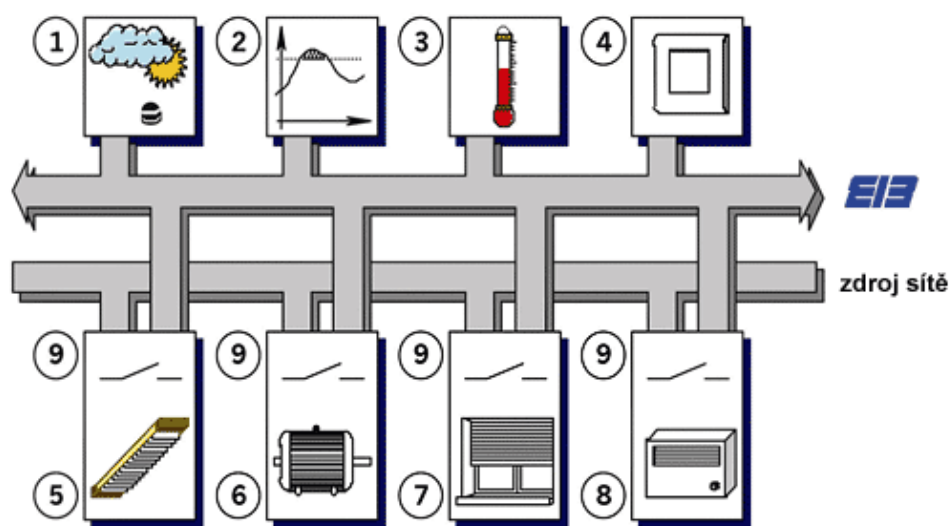
„Data určená pro vzájemnou komunikaci se vkládají do datového telegramu a prostřednictvím instalační sběrnice se digitálně přenášejí. Sběrnici je možno realizovat různým fyzikálně - technickým způsobem. V případě KNX/TP je to kabel Twisted Pair, tj. kroucený pár metalických vodičů, u KNX.PL je to silový kabel a v případě KNKRF se používá radiového

spojení. Lze rovněž použít přenos dat optickým kabelem. Výměna informací probíhá přímo mezi jednotlivými účastníky, kteří mají realizovat nějaké funkce.

KNX/EIB se používá např. pro realizaci ovládání a řízení osvětlení a nastavení žaluzií. Snímač, vydá příkaz a vyšle ho sběrnici jako datový telegram k akčnímu prvku, ten pak příjem pokynu potvrdí potvrzujícím telegramem - zpětným hlášením a provede příkaz.

Komplexnost KNX/EIB byla koncipována tak, aby pokryla všechny technické systémy budov a aby ji kvalifikovaný elektroinstalatér byl schopen naprogramovat a uvést do provozu. Přenosová rychlost je nízká, je však dostatečná pro povely sepnutí nebo regulační řídicí povely. U protokolu KNX/EIB dosahuje rychlost přenosu 9,6 kbit/s.

V elektrotechnických oborech je technologie Evropské instalační sběrnice již po několik let stabilní součástí profesní kvalifikace. Z tohoto důvodu má KNX/EIB své stálé místo v automatizaci budov, kde instalační technika spolu s komponentami technických systémů budov je v popředí, např. při regulaci osvětlení.“ (1)



Obr. 4: Příklad sběrnicevého systému (4)

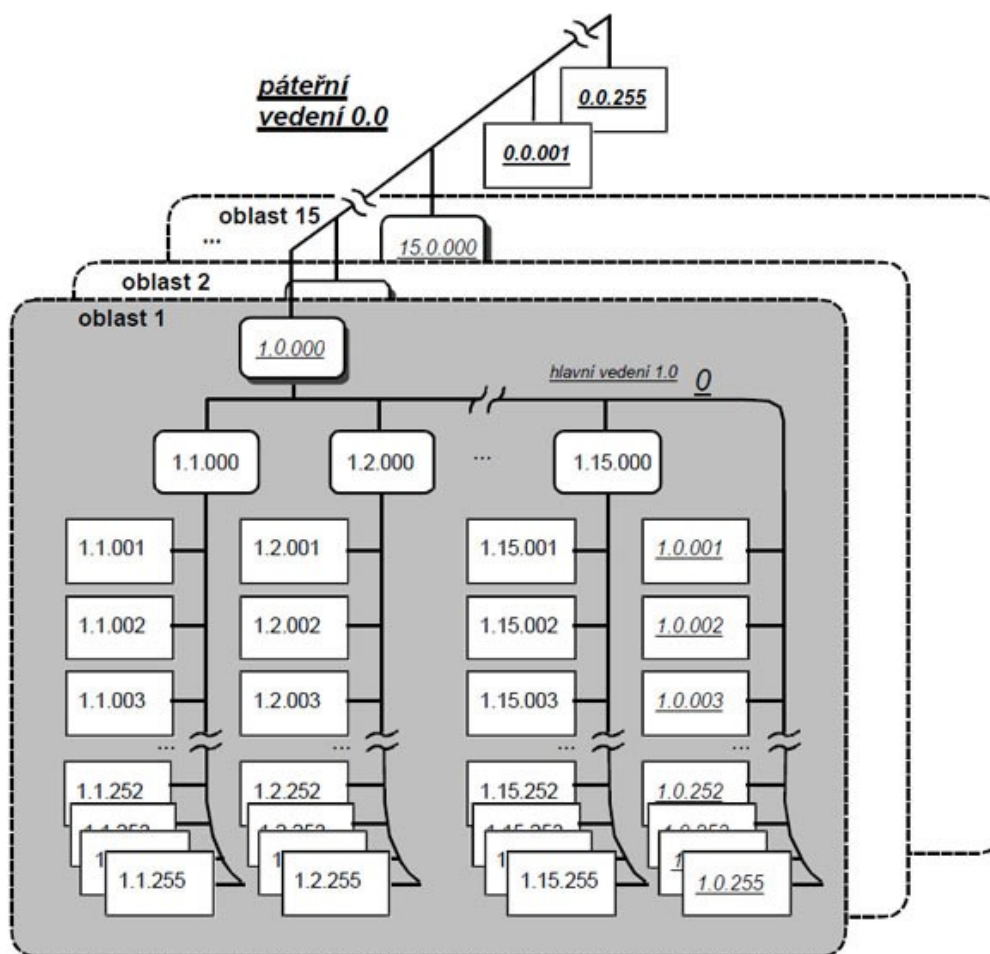
Legenda k Obr. 4:

- 1,2,3,4 – senzory, které jsou připojeny na datové sběrnici
- 5, 6, 7, 8 – spotřebiče, které jsou připojeny k spínači 9 a rozvodné síti 230V

„Z automatizační úrovně se informace např. pro vizualizaci a systém plánování výroby, předává na úroveň managementu (vrcholového řízení). V těchto případech se přenáší velké objemy dat a musí být rovněž po delší dobu k dispozici. Komunikace na vyšších rovinách automatizace probíhá převážně prostřednictvím sítí LAN.“ (1)

3.1. Topologie sítě

„Evropská instalační sběrnice se dostala na hranici svých možností, když bylo třeba přenášet současně se spínacími povely též velké množství analogových signálů. Na druhé straně se rozšíření KNX/EIB z velké části zakládá na řemeslně přívětivém zpracování. V době, kdy se tato sběrnice vyvíjela, nebyly řízení a regulace provozně technických zařízení středem pozornosti. Součástí sběrnic, resp. sítí, jsou konkrétním způsobem fyzicky vzájemně spojeny. Toto uspořádání se nazývá topologií sběrnic nebo sítí. Jestliže účastník chce přistoupit na sběrnici i nebo na síť, musí se tak stát podle určitých pravidel, aby se vysílané signály při současném přístupu více účastníků nepřekrývaly a nevznikaly tak chyby. Toto řeší spojový přístupový protokol kanálu.“ (1)



Obr. 5: Topologie systému KNX (5)

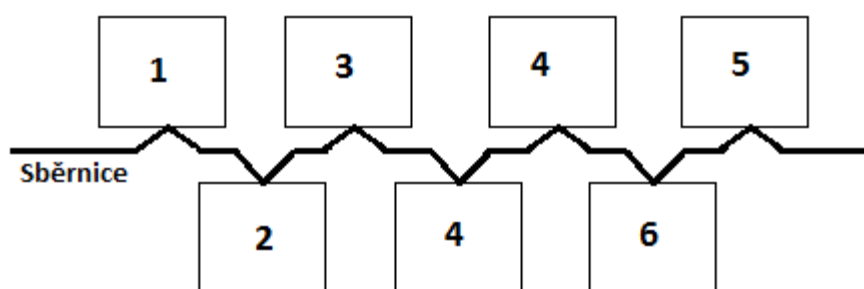
Jako u konvenční elektroinstalace, tak i instalace KNX/EIB musí být připojena na síť dodávající energii. Vedle toho ještě existuje komunikační síť instalační sběrnice KNX/EIB. Obě sítě jsou od sebe galvanicky oddělené, což vyžaduje i použití různých typů vedení. Pro instalaci KNX/EIB musí být kromě napájecí silové sítě vyprojektována a zřízena komunikační síť. Aby mohla konfigurace instalace KNX/EIB plnit funkce požadované zákazníkem, musí projektant

nejdříve rozhodnout, jaká systémová zařízení, jaké snímače a akční členy budou zapotřebí. Potom stanoví, do kterých míst budovy se budou jednotlivé přístroje KNX/EIB montovat a jak se sběrnice - prostřednictvím vedení - spojí do jednoho komunikačního systému. (1)

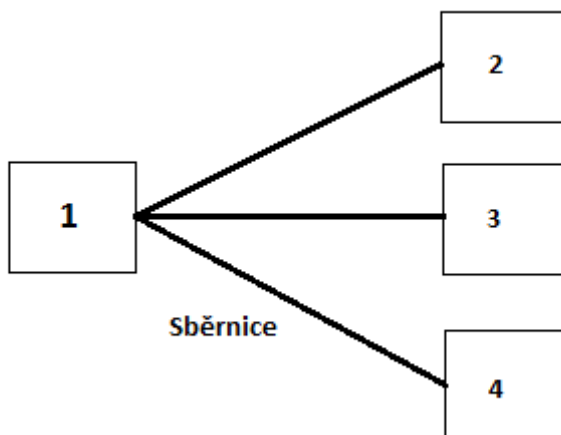
Je dovolen rozvod sběrnice:

- Lineární
- Paprsková
- Větvená, stromová.

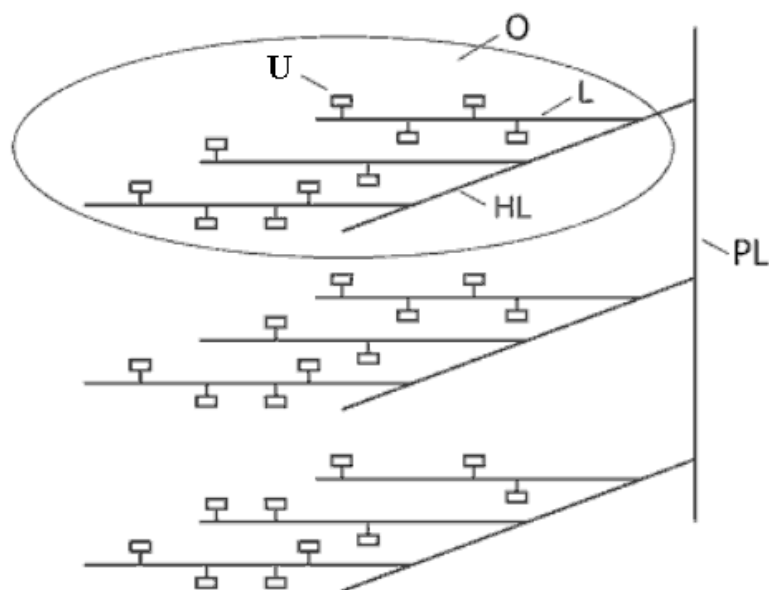
Kruhová struktura není povolena – dojde k uzavření smyčky.



Obr. 6: Lineární topologie KNX/EIB



Obr. 7: Paprsková topologie KNX/EIB



Obr. 8.: Stromová topologie KNX/EIB (1)

„Rozeznáváme následující hierarchii:

- účastník (U), jednotlivé účastnické stanice jsou spojeny vedením, neboli linií (L),
- více vedení – linií se spojuje prostřednictvím hlavní linie (HL) a společně vytvářejí oblast (O),
- více oblastí je společně napojeno na páteřní linii neboli na páteřovou linii (PL).

Pod pojmem oblasti, si lze představit např. jedno podlaží v domě. Na chodbě každého podlaží jsou vodiče, na které jsou ve stanovených místech připojeny přístroje na sběrnici.

Menší konfigurace sestávají často jen z několika účastníků, kde je každému přiřazena jedna oblast a jedna linie. Neexistuje účastník, který by nebyl přiřazen linii nebo oblasti.

Počet účastníků, linií a oblastí závisí např. na prostorovém rozmístění sítě a na počtu instalovaných přístrojů. Za normálních okolností pozůstává síť z několika jednotlivých liniiových segmentů a může mít až 64 účastníků. U komplikovaných konfigurací s velkou plošnou rozlohou a větším počtem účastnických stanic se síť strukturuje do oblastí, které zahrnují jednu až několik linií. Jedna linie se čtyřmi liniiovými segmenty může mít až 256 připojených účastníků. Dalším důvodem pro větší počet linií je např. zálohování za účelem větší dostupnosti sítě nebo za účelem snížení zátěže sběrnice na linii, na níž jsou připojeny řídicí jednotky.

Liniové segmenty, linie a oblasti jsou datově - technicky vzájemně spojeny pomocí propojovacích jednotek - liniiových zesilovačů, liniiových spojek a oblastních spojek; tyto přístroje jsou datově i technicky vzájemně kompatibilní, takže mohou být vzájemně propojeny. Snímače, akční členy a rozhraní jsou v systému KNX/EIB reálnými účastnickými stanicemi, které se účastní datové komunikace. Je jim přidělena individuální adresa, na níž jsou umístěny. Každá linie (segment, hlavní linie neboli páteřní linie) musí být vybavena - z elektrických a komunikačních důvodů - vlastním zdrojem napájení, vybaveným tlumivkou.“ (1)

3.1.1. Napájecí zdroj

Napájecí zdroj (NZ) zajišťuje, aby každý účastník v linii byl připojen na napájení 24 V DC. Tím, že zdroj je vybaven tlumivkou, umožňuje datovou komunikaci v linii. Zdroj je napojen na síť (sít' 230 V) a zajišťuje sběrnici stabilní napětí, obvykle 30 V DC. To však neznamená, že všechny přístroje, které jsou na sběrnici připojeny, mají vstupní napětí 30 V; se vzrůstající vzdáleností od zdroje napětí na sběrnici klesá. Jmenovité napětí na sběrnici činí 24 V.



Obr. 9.: Napájecí zdroj 640mA s integrovanou tlumivkou a dvěma výstupy 30V DC (6)

Jsou zdroje napětí s maximální hodnotou proudu 640 mA, které zajišťují napájení nejvýše 64 účastníků. Zdroje s maximálním proudem 320 mA, které slouží k napájení max. 32 účastníků anebo se jmenovitým proudem 160 mA pro napájení až 16 účastníků. Volba zdroje závisí na skutečném anebo plánovaném počtu účastníků v každé linii. Koncepce vlastního zdroje v každé linii má tu výhodu, že v případě výpadku některého ze zdrojů vypadnou z komunikace jen účastníci, kteří jsou v příslušné linii zařazeni. Ostatní účastníci pokračují dále v činnosti. V případě výpadku hlavní linie nebo dokonce páteřní linie je pochopitelně přerušen provoz v celém rozsahu linií i oblastí. Propojovací jednotky jako liniová spojka (LS) a oblastní spojka (OS) jsou vždy napájeny z nadřazené linie. (1)

- liniové spojky (LS) jsou vždy napájeny ze zdroje z nadřazené linie,
- oblastní spojka (OS) je vždy napájena ze zdroje z páteřní linie,
- liniový zesilovač neboli opakovač (LO) je napájen ze zdroje z nadřazeného segmentu.

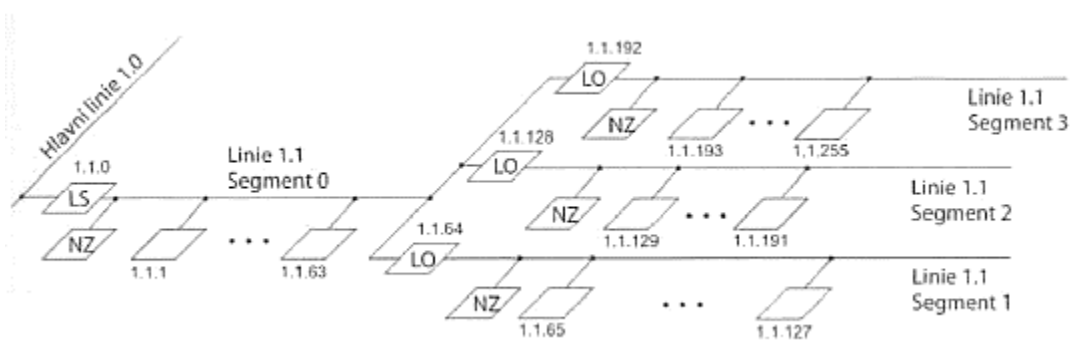
3.1.2. Liniová spojka

Liniová spojka je systémový přístroj a může sloužit jako liniový opakovač (LO) nebo jako liniová spojka (LS), nebo jako oblastní spojka (OS). To se provede v rámci projektu přiřazením speciální individuální adresy a volbou odpovídajícího aplikačního programu (zajištění spojení, resp. zesílení a). Unie, resp. liniové segmenty, které jsou prostřednictvím LS, OS a LO propojeny, jsou elektricky soběstačné. Galvanické oddělení zůstává zachováno i při použití LS, OS a LO. (1)



Obr. 10: Liniová spojka (6)

Liniový opakovač



Obr. 11: Linie se třemi liniovými opakovači (LO) a čtyřmi segmenty (1)

„Aby se mohla linie, která sestává jen z jednoho liniového segmentu (s číslem 0) rozšířit na více než 64 účastnických stanic nebo aby se eventuálně mohla i prostorově rozšířit, může se jedné liniové spojky využít jako liniového opakovače (Repeater). Takto mohou být připojeny maximálně tři další liniové segmenty (s čísly 1-3).

Liniové spojky (LS) a oblastní spojky (OS) mohou zajišťovat funkci filtrace. To znamená, že telegramy, které odesílá vysílač po linii, mohou být dále přesměrovány, jestliže se přijímač nachází vně linie, na které je připojen vysílač.“ (1)

3.1.3. Oblastní spojka

„K informatickému spojení mezi oblastmi se používá hlavních linií oblastí, které se prostřednictvím oblastních spojek napojují na páteřní linii.

Telegramy používané v normálním provozu mohou být přeneseny až přes šest LO, LS a OS. Ke kontrole počtu přenosů je v telegramu tzv. přepravní pole - routingové číslo. Vysílač má k dispozici přepravní pole, což je čítač s počtem přenosů $n = 6$. Jestliže některá spojka LO, LS nebo

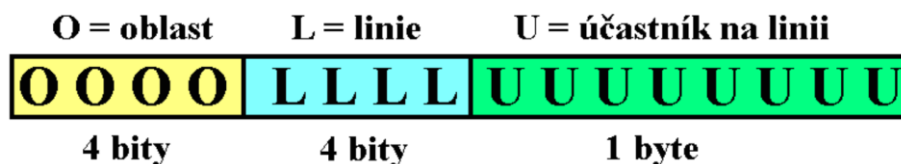
OS přeneše datový telegram, hodnota čítače n se sníží o 1. Při poklesu na nulu se již další přenos neuskuteční.“ (1)

3.2. Adresování

„Všechny přístroje KNX/EIB, které se účastní komunikace s daty (snímače, akční členy, spojky) musí mít jednoznačnou individuální adresu. Programátor ji vloží programem ETS4 v době uvedení do provozu a uloží se dlouhodobě v EEPROM. Účastnické stanice, zahrnuté do jedné skupiny si mezi sebou vyměňují informace. Účastnické stanice jedné takové skupiny mají ještě vedle své individuální adresy také skupinovou adresu, na které jsou v rámci techniky komunikace dosažitelné.“ (1)

3.2.1. Individuální adresa

Individuální adresa se po uvedení do provozu využívá k diagnostice, opravě chyb atp.



Obr. 12: Individuální adresa KNX (4)

3.2.2. Skupinová adresa

Komunikace mezi přístroji v jedné instalaci probíhá prostřednictvím skupinových adres.

Skupinová adresa se u určitého projektu vyskytuje nejméně dvakrát. Jednou u snímače a jednou u akčního členu. Přiřazením stejné skupinové adresy snímači a akčnímu členu se tyto navzájem funkčně spojí. Dříve se používalo dvouúrovňového adresování

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	H3	H2	H1	H0	U10	U9	U8	U7	U6	U5	U4	U3	U2	U1	U0
X	Hlavní skupina				Podskupina										

Obr. 13 Skupinová adresa (dvouúrovňová adresa) (1)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	H3	H2	H1	H0	M2	M1	M0	U7	U6	U5	U4	U3	U2	U1	U0
X	Hlavní skupina				Střední skupina			Podskupina							

Obr. 14 Skupinová adresa (tříúrovňová adresa) (1)

3.2.3. Vlajka adresy

Aby účastníci KNX při příjmu telegramu mohli poznat, zda se jedná o individuální, nebo skupinovou adresu, existuje bit v 6. datovém bytu – vlajka adresy příjemce.

0b – individuální adresa.

1b – skupinová adresa

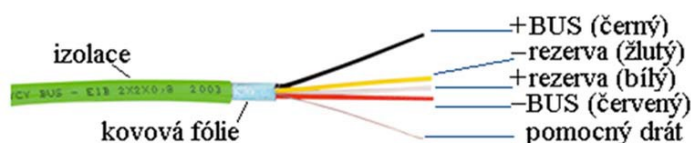
3.3. Přenosová média

Pro přenos dat mezi účastnickými stanicemi může být využito více médií, která jsou mezi sebou kompatibilní:

- Twisted Pair – kroucený pár – KNX.TP
- Power Line – silové vedení - KNX.PL
- Radiový přenos - KNX/RF
- Ethernet - KNXnet/IP
- Optická vlákna.

3.3.1. KNX.TP

Nejčastěji se používá vedení krouceným párem – Twisted Pair(KNX.TP). Pro klasický kroucený pár je typické vedení YCYM 2x 2 x 0,8. Má zelené opláštění PVC a sestává se ze dvou párů žil, jejich vodiče mají průměr 0,8mm. Jsou dvojici zkroucené a odstíněné. Pár žil s červeným vodičem (+) a s černým vodičem (-) se používá pro napájení účastníka energií a současně pro přenos dat. Páru žil se žlutým a bílým vodičem se používá jako rezervy, např. pro přídatné napájení účastníka. Vedení TP se může pokládat pod omítkou v suchém, vlhkém i mokřém prostředí. Platí pro ně stejné instalační podmínky jako pro vedení silnoproud. Rychlost přenosu dat na této sběrnici je 9600 bit.s⁻¹. (1)



Obr. 15: Popis svěrnicevého kabelu YCYM (7)

Tímto kabelem se propojují všechny KNX sběrnicevé přístroje – účastníci sběrnice (snímače, akční členy a kontroléry) a také veškeré pomocné přístroje (komunikační rozhraní, napájecí zdroje).

3.3.2.KNX.PL

Pro komunikaci se využívají přímo vodiče nízkého napětí 230/400V. Výhodou je přenos dat přímo po silových vodičích, využívá se této komunikace v prostorech, kde není možné položit datovou sběrnici KNX. Přenosového média přímo po vodičích můžeme využít v prostorech s klasickou elektroinstalací, bez větších zásahu do stávající elektroinstalace. Rychlost přenosu dat této sběrnice je 240 bit.s^{-1} .

3.3.3.KNX/RF

Pro komunikaci se využívá vysokofrekvenční přenos. Jedná se o rádiový – bezdrátový přenos. Tohoto média můžeme využít, když nemáme možnost drátově propojit dvě zařízení. Přenosová rychlost je $16,34 \text{ kbit.s}^{-1}$.

3.4. Průběh komunikace, druhy telegramů

„Informace mezi účastníky KNX/EIB se vyměňují prostřednictvím telegramů. Rozlišují se datové telegramy a zpětná hlášení (potvrzovací telegramy).

Spouštěcím impulsem k odeslání datových telegramů jsou jednotlivé události, např. Stisknout levé horní tlačítko čtyřnásobného tlačítkového snímače", (Existují však přístroje KNX/ EIB, které datové telegramy samy odesílají.) Důsledkem této události je, že určitý přístroj, odešle datový telegram s určitou skupinovou adresou.

Všichni příjemci, kteří patří do této skupiny, potvrzují (současně) příjem datového telegramu zpětným hlášením. Protože by následovalo přetížení větším počtem zpětných hlášení, zpětná hlášení proběhnou jako souhrnná. Pokud se účastník nachází mimo linii vysílače, bude telegram potvrzen liniíovou spojkou.“ (1)

3.5. Datový telegram TP1

Datový telegram obsahuje binární informace v sedmi binárních polích, která jsou strukturována takto:

- kontrolní pole (1 byte),
- adresa odesílatele (2 byty),
- adresa příjemce (2 byty + 1 bit),
- přepravní pole (routingový čítač) (3 bity),
- údaj o délce datového pole (4 bity),
- datové pole (od 1 do 16 bytů),
- zkušební pole (1 byte).

Tabl. struktura darového telegramu (8)

Kontrolní pole (Control field)	Zdrojová adresa (Source address)	Cílová adresa (Target address)	Routingový čítač (Routing counter)	Délka (Length)	Užitečná data (Useful data)	Ověřovací byte (Check byte)
8 bit	16 bit	16 + 1 bit	3	4	až 16 x 8 bit	8 bit

„Nejkratším datový telegram obsahuje 9 bytů dat (se 2 byty uživatelských dat), nejdelší může mít 23 bytů dat (se 16 byty uživatelských dat). Nejčastěji datové telegramy, kam patří spínací telegramy, mají délku 9 bytů. Po přijetí datového telegramu účastnická stanice odpovídající skupiny odešle ihned po uplynutí 13 bitových dob přestávky příslušné potvrzující zpětné hlášení, Vzniká sumární úhrnný telegram. Dominantní nulové bity v tomto případě přepíšou recesivní jedničkové bity jako při přístupu na sběrnici.“ (1)

„**Kontrolní pole** - dává informaci o důležitosti neboli přenosové prioritě. Ta je důležitá, jestliže začne vysílat více účastníků současně. Tuto prioritu lze nastavit každému skupinovému objektu prostřednictvím ETS, standardní nastavení je nízká priorita. Obsahuje také informaci, zda se jedná o opakovaný telegram. Tím se zamezí, aby účastníci vykonávali příkazy opakovaně.

Control field								
	7	6	5	4	3	2	1	0
Frame type			repeat flag		priority	priority		
FT	0	r	1	p1	p0	0	0	L_Data-Frame
	1	1	1	0	0	0	0	L_Poll_Data-Frame
	x	x	0	0	x	x	0	Acknowledgement Frame

FT = 0 : L_Data_Extended Frame
 FT = 1 : L_Data_Standard Frame

r = 0 : repeated L_Data-Frame
 r = 1 : not repeated L_Data-Frame

p1	p0	
0	0	system priority
1	0	urgent priority
0	1	normal priority
1	1	low priority

Zdrojová adresa - udává fyzickou adresu přístroje, který informaci na sběrnici zaslal.

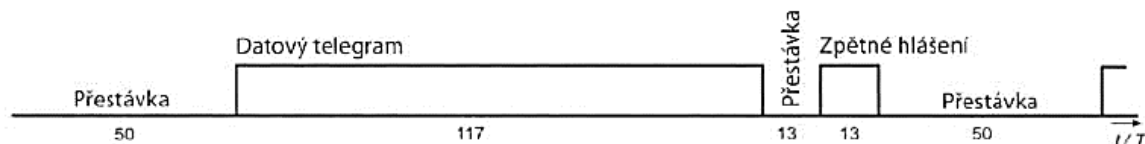
Cílová adresa – je to zpravidla skupinová adresa, může se však jednat i adresu fyzickou.

Routingový čítač - udává počet routerů (průchodů liniovou spojkou), přes které bude telegram vyslán.

Délka - velikost užitečné informace.

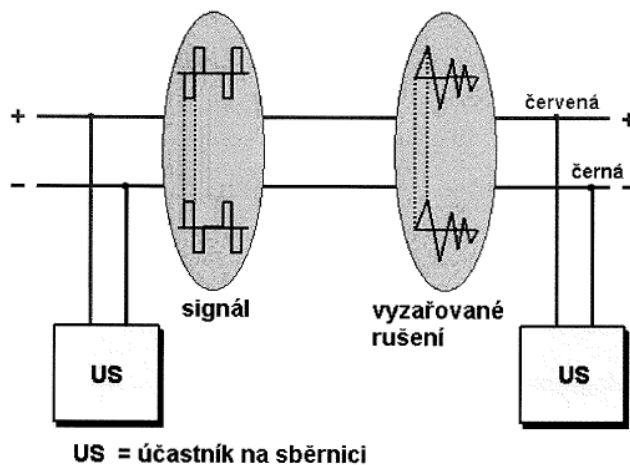
Užitečná data - tato část telegramu obsahuje předávaná data EIS (EIB Interworking Standard) a vlastnosti komunikačních objektů, tzv. vlajky (flag), které určují způsob komunikace.

Ověřovací byte - informace ve formě paritních bitů, která slouží k ověření správnosti doručeného telegramu využívající příčného a podélného zabezpečení. Kombinací těchto dvou zabezpečení vzniká zabezpečení křížové.“ (8)



Obr. 16: Přenos telegramu KNX (1)

Data se přenášejí symetricky po obou vodičích krouceného páru. Sběrníkový přístroj vyhodnocuje rozdíl napětí mezi oběma vodiči. Vyzařované rušení působí na obě jádra vodičů ve stejné polaritě a neovlivní proto rozdíl napětí signálu.



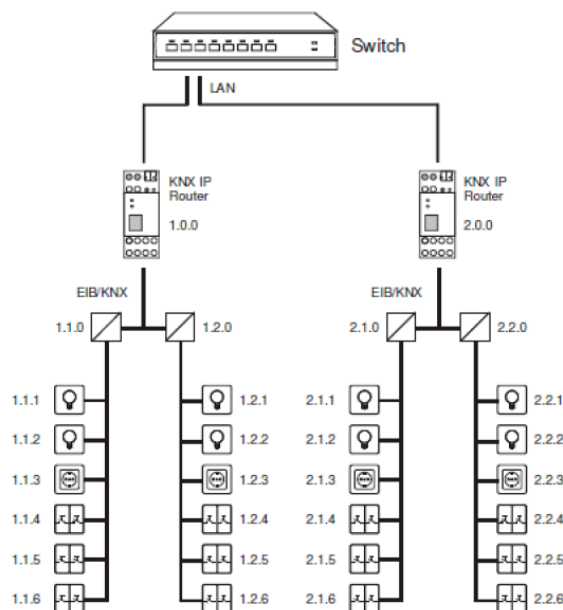
Obr. 17: Symetrický přenos (9)

Příklad telegramu s bitem pro vypnutí:

Telegram média TP1: BC 11 04 00 01 E1 00 80 CS

3.6. Ethernet KNXnet/IP

Poslední a také nejmladší technologií je přenos prostřednictvím IP telegramů, z čehož vyplývá, že k přenosu se využívají běžné Ethernetové sítě. Nejčastěji se používá pro monitorování, vizualizace a vzdálenou správu, nebo také můžeme nahradit náhrada hlavních a páteřních linií systému TP Ethernetovou linkou, která je mnohem rychlejší. Ovšem za použitím spojek, které byly navrženy pro tento účel. (1)



Obr. 18 Náhrada liniových spojek tzv. „IP Routers“ (10)

Výhodou tohoto uspořádání je, že všechny vertikální obousměrné komunikace mezi centrálou budovy a KNX je omezena rychlosti pouze na sekundární linie (Ethernet je nejméně 1000 krát rychlejší). K centrálnímu řízení budovy může být souběžně připojeno několik rozhraní, čímž se násobí celková přenosová rychlost. (9)

3.6.1. KNX IP Router

Principiálně pracuje komunikace stejně jako na směrovač dat na TP hlavní linii: když IP router chce odeslat telegram napříč liniemi, odešle skupinovou adresu IP adresou po Ethernetu. Všechny ostatní IP routery jsou připojeny k tomuto kupónovému adresování a jsou schopny přijmout a vyhodnotit tento telegram.“ „Software pro zařízení KNX IP se skládá ze dvou protokolů. IP protokol a UDP je nezbytný pro komunikaci přes Ethernet, protože KNX IP je založen na komunikaci bez připojení. Singlecast a Multicast zprávy jsou posílání přes UDP.“ (9)

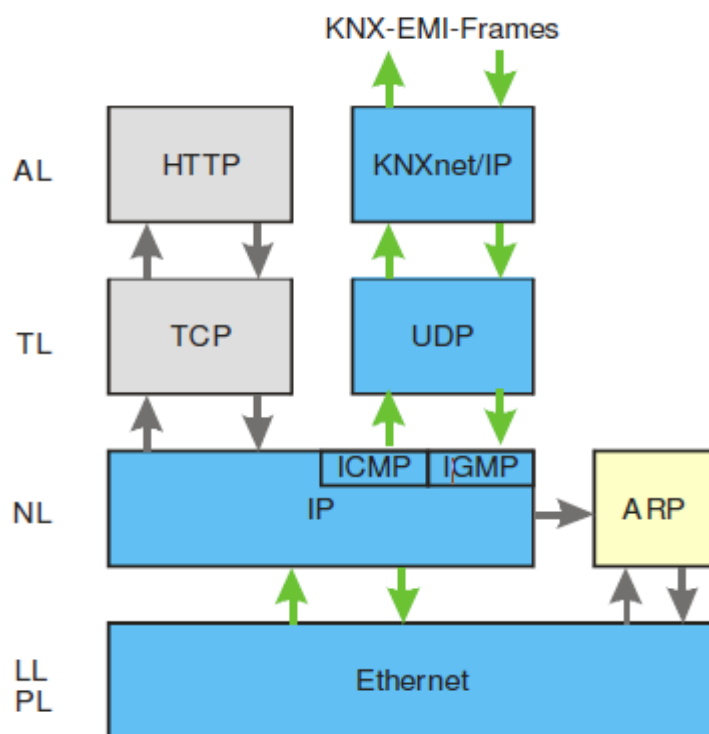


Obr. 19: KNX IP router od firmy Weinzierl (10)

3.6.2.KNX/IP telegram

KNX IP telegram se shoduje s vrstvy v ISO/OSI modelu a obsahuje následující vrstvy:

- Aplikační vrstva – KNXnet/IP
- Transportní vrstva – uživatelský datový protokol
- Síťová vrstva – internetový protokol, adresa
- Fyzická vrstva – Ethernet



Obr. 20 Sklání KNXnet/IP telegramu (10)

Každé zařízení na Ethernetové síti má přiřazenou fyzickou adresu (MAC adresa). Fyzická adresa je unikátní na světě, a je pevně vázaná na zařízení. Délka je 6 Bytů. Příklad MAC adresy: 00-50-C2-55-40-00.

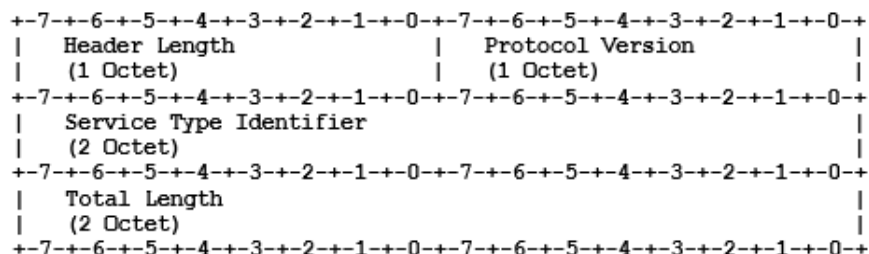
IP adresa slouží ke komunikaci na síťové vrstvě, je přidělována administrátorem. Délka je 4 byty. Příklad IP adresy: 192.168.1.1

Při vysílání telegramu více účastníkům sběrnice se využívá více směrové vysílání (multicast). Znamená to jeden vysílač-více příjemců. Pro skupinové vysílání je přiřazená speciální oblast IP adres: 224.0.0.0 – 239.255.255.255. KNXnet/IP má přiřazenou multicast adresu na 224.0.23.

Transportní vrstvu přestavuje pro skupinové vysílání UDP protokol.

Pro přenos telegramu se využívají dva módy: Tunneling-tunelování a Směřování. Tunneling slouží pro vytvoření komunikace z bodu do bodu. Využívané pro první uvedení TP1 zařízení do

provozu. Směřování se používá k připojení k více linkám, filtrování telegramu a pro skupinovou komunikace přes multicast.



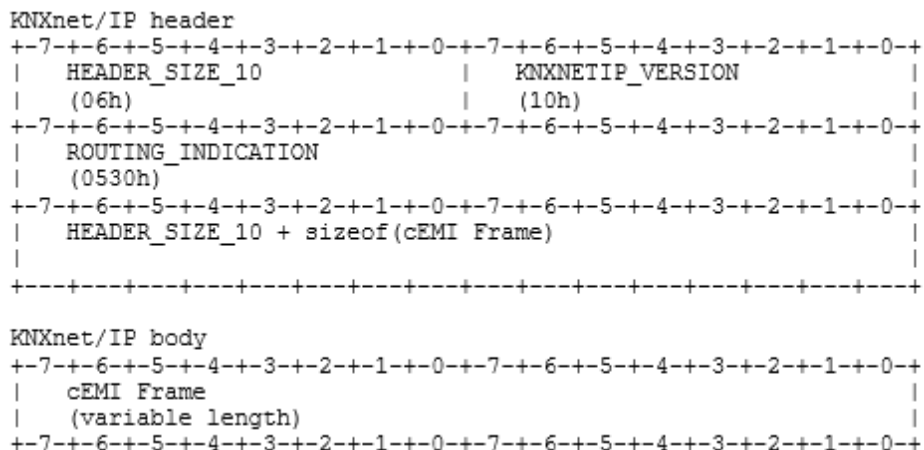
KNXnet/IP hlavička v binárním formátu (11)

Header length: Délka IP hlavičky, pro protokol 1.0 je délka 06h.

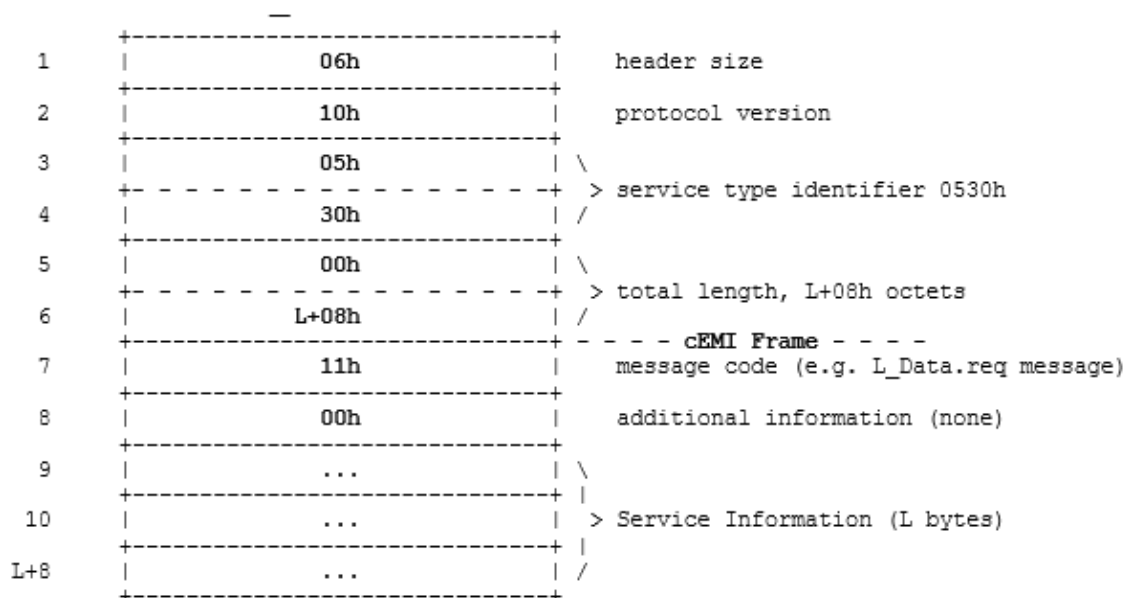
Protocol version: Informace verze revize KNXnet/IP. Jediný platná verze je 1.0 (reprezentován v hexa 10h)

KNXnet/IP service: Definuje druh akce, které mají být provedeny. Routing_indication: 0530h. 0530h se používá pro posílání telegramů přes IP síť.

Total Length: Celková délka telegramu. Délka IP hlavička společně s délkou cEMI telegramem.



Kompletní telegram KNX/IP (Datagram) (12)



Příklad binárního formátu telegramu KNX/IP (12)

4.6.3 cEMI telegram:

KNX IP medium využívá cEMI Data Link Layer Indication (L_Data.ind). Telegram pro UDP transportní vrstvu na Multicast adresách. Zpráva generuje strukturu střední implementace KNX zprávy. Telegram přenášený přes KNX je přenášen jak cEMI (Data Link Layer, L_Data.ind = cEMI message code (29h) telegram. (13)

Message Code	Additional Info Length	Additional Information	Control field 1	Control field 2	Src. High	Src. Low	Dest. High	Dest. Low	NPDU	
MC	AddIL	...	Ctrl1	Ctrl2	SAH	SAL	DAH	DAL	L	TPCI/APCI & data
1 octet	1 octet	var. length	1 octet	1 octet	2 octets		2 octets		1 octet	var. length

29h

formát zprávy: cEMI Data Link Layer Indication, (13)

1. byte : Message code: Kód cEMI zprávy: code L_Data.ind: 29h.

2. byte: Additional info length: délka dodatečné informace.

Additional Information: cEMI zpráva může obsahovat dodatečně informace. Informace o PL, RF médiu atp.

Message Code	Additional Info Length	Additional Information								Service Information
MC	AddIL	Type ID	Len	information	Type ID	Len	information
1 octet	1 octet	1 octet	1 octet	dep. on inf. type	1 octet	1 octet	dep. on inf. type	var. length



AddIL (= number of octets of Additional Information)

cEMI zpráva obsahující dodatečné informace (13)

3. byte: Control field 1: kontrolní pole

1 octet							
Ctrl1							
7	6	5	4	3	2	1	0
FT	OR	RS	PL	AC			

Typ vlajky telegramu (FT): Určuje typ telegramu, který je používán pro přenos nebo příjem telegramu.

0: rozšířený telegram

1: standartní telegram

Opakující se vlajka (příznak),(R): Opakování, neplatí pro všechna media

0: opakovat telegram pokud dojde k chybě

1: neopakovat

Systém vysílání vlajky (příznaku),(SB): Určuje, zda je telegram přenášen systémovým broadcastem nebo broadcastem. Použitelné jen na otevřených médiích.

0: systémový broadcast

1: broadcast

Priorita (P): Určuje prioritu vysíleného nebo přijímaného telegramu.

11b	low	Povinné pro dlouhé telegramy
01b	normal	Výchozí nastavení pro krátké telegramy
10b	urgent	Vyhrazeno pro urgentní telegramy
00b	system	Vyhrazeno pro vysokou prioritu, systémovou konfiguraci a řídicí postupy

Potvrzení žádosti o vlajku (příznaku) (A): Toto určí, zda je požadavek na L_Data.reg. Neplatí pro všechny média.

0: potvrzení není požadováno

1: potvrzení je požadováno

Potvrzující vlajka (příznak) (C): Indikuje chyby v přenášeném telegramu.

0: žádná chyba

1: chyba

5. byte: Control field 2:

Typ cílové adresy (AT):

0: individuální (jednotlivá)

1: společná (group)

1 octet							
Ctrl2							
7	6	5	4	3	2	1	0
AT	HC					EFF	

Hop Count (HC): Hodnota binárního kódování, liniový opakovač se rozhodne, jestli přepoše telegram dál, nebo ne, s každým dalším opakovaním se číslo snižuje. Normálně se nastavuje na 6.

Rozšířený format telegramu (EFF):

0000b: standartní telegram

01xxb: LTE telegram

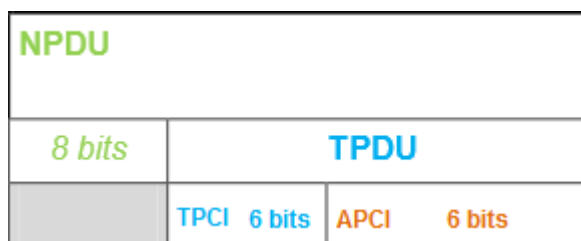
6. – 7. byte: Source: Zdrojová adresa zařízení, které telegram vysílá.

7. – 8. byte: Destination: Cílová adresa zařízení, kterému je telegram vysílán. Pokud se jedná o skupinovou adresu, oslovují se všichni účastníci sběrnice, kteří mají tuto adresu přiřazenu. V případě fyzické adresy se oslovuje (aktivuje) jen jeden přístroj (účastník). Toho se využívá pro zasílání systémových telegramů (programování sběrnice). To, jestli se jedná o adresu skupinovou, nebo fyzickou určuje 8 bit v 9. bytu.

1 - skupinová adresa.

0 - fyzická adresa .

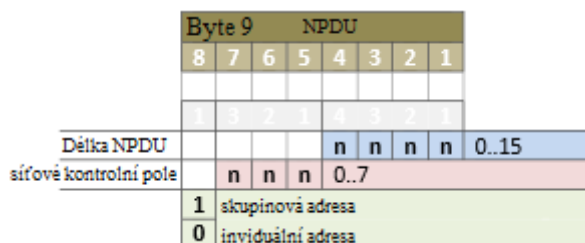
9. - 11. byte: NPDU:



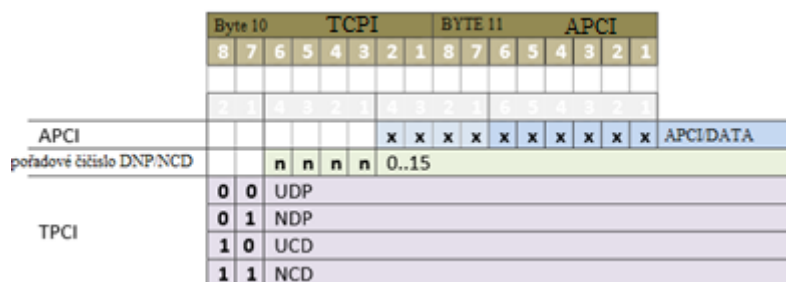
Formát NPDU (14)

9. byte:

Délka: Informace o délce maximální délka je 255, TPCI byte není zahrnut.



NPDU byte (15)



APDU byty (15)

10. – 11. byte:

TPCI: (Transportní vrstva):

Pro UDP: 000000b

APCI (Aplikační vrstva):

Pro skupinové adresy data:

0000b GroupValue_Read

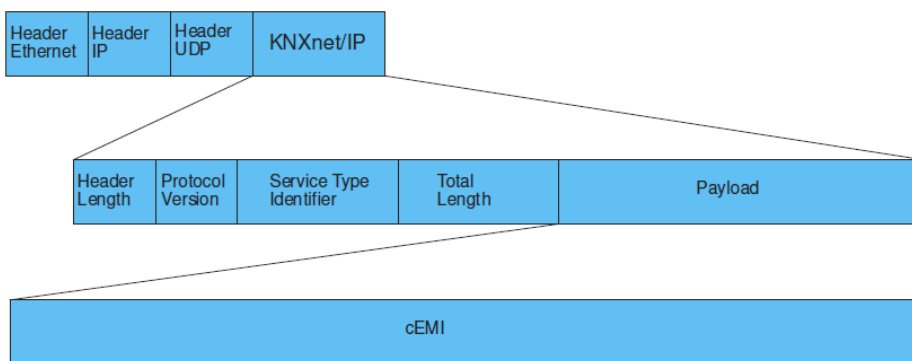
0001b GroupValue_Response

0010b GroupValue_Write

APCI/data:

000000b – bit pro vypnutí

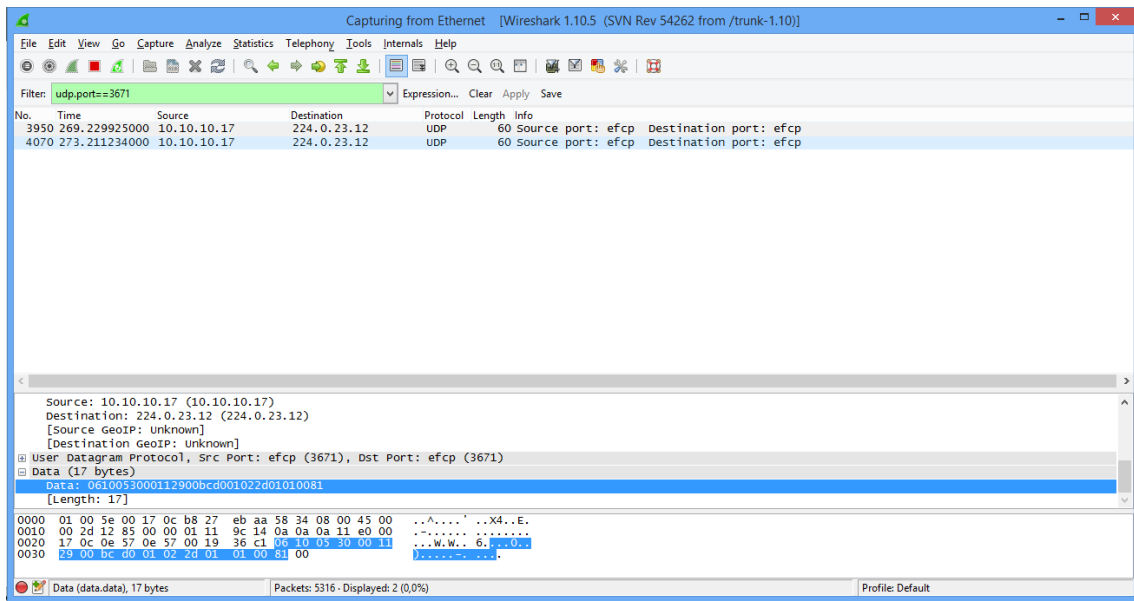
000001b – bit pro zapnutí



Obr. 21 KNXnet/IP telegram na základě UDP (16)

Příklad telegramu s bitem pro vypnutí:

Telegram média KNX/IP: 06 10 05 30 00 11 29 00 BC D0 11 04 00 01 01 00 80“ (16)



Obr. 22: Wireshark – odhycený datový telegram KNXnet/IP

4. Ethernet

„Ethernet je v současnosti nejrozšířenější technologií pro budování počítačových sítí typu LAN a stále více používány v rostoucí škále aplikací. Především díky své univerzální kompatibilitě jsou nízko úroňové ovladače Ethernet dostupné pro téměř všechny integrované operační systémy používané v modulárních počítačích. Nejpopulárnější je pravděpodobně Linux, který bez problémů pracuje na architekturách x86, MIPS i ARM. Stále rostoucí počet mikrokontrolérů však podporuje také konektivitu přes Ethernet i bez výhod integrovaného operačního systému, jakým je například Linux.“ (17)

„V dnešní době se využívá Ethernetové rozhraní s RJ-45 konektorem, který nalezneme skoro ve všech produktech využívajících Ethernetové komunikace. Běžná přenosová rychlost dosahuje 100 až 1000MB/s . Ethernet je stále vyvíjející se technologií, přicházející stále s vyšší rychlosti přenosu.“ (18)

„To představuje integrované prostředí bez hranic, alespoň pokud jde o data. Zesílený důraz na tvorbu, shromažďování a přenos dat, přináší významné výhody pro tuto úroveň propojitelnosti a přenositelnosti dat.

Proto je dnes obchodně i technicky výhodnější integrovat konektivitu přes Ethernet do nových platform.“ (17)



Obr. 23: Ethernetový kabel s koncovkou RJ-45

„Ethernet je přenosovou technologií zajišťující přenos dat v ISO/OSI modelu. ISO/OSI model je referenční komunikační model, který rozděluje vzájemnou komunikaci mezi počítači do sedmi souvisejících vrstev. Úkolem každé vrstvy je poskytovat služby následující vyšší vrstvě a nezatěžovat vyšší vrstvu detaily jak je daná služba realizovaná. Pro přenos dat mezi vrstvami se využívají pakety. V každé vrstvě se pak k paketu doplňují další informace (formátování, adresa), které jsou nezbytné pro úspěšný přenos po síti.“ (19)



Obr. 24: Model ISO/OSI vrstev (19)

4.1. Vrstvy ISO/OSI modelu

4.1.1. Fyzická vrstva

„Definuje prostředky pro komunikaci s přenosovým médiem a s technickými prostředky rozhraní. Dále definuje fyzické, elektrické, mechanické a funkční parametry týkající se fyzického propojení jednotlivých zařízení. Je hardwarová.“ (19) Ethernet realizuje tuto vrstvu dráty a signály.

4.1.2. Linková vrstva

„Zajišťuje integritu toku dat z jednoho uzlu sítě na druhý. V rámci této činnosti je prováděna synchronizace bloků dat a řízení jejich toku. Je hardwarová.“ (19)

4.1.3. Síťová vrstva

„Definuje protokoly pro směrování dat, jejichž prostřednictvím je zajištěn přenos informací do požadovaného cílového uzlu. V lokální síti vůbec nemusí být, pokud se nepoužívá směrování. Je hardwarová ale, když směrování řeší PC s dvěma síťovými kartami je softwarová.“ (19) Jedna se IP protokol, IP adresu.

4.1.4. Transportní vrstva

„Definuje protokoly pro strukturované zprávy a zabezpečuje bezchybnost přenosu (provádí některé chybové kontroly). Řeší například rozdělení souboru na pakety a potvrzování. Je softwarová.“ (19) Tato vrstva je realizovaná TCP/IP nebo UDP protokolem.

4.1.5. Relační vrstva

„Koordinuje komunikace a udržuje relaci tak dlouho, dokud je potřebná. Dále zajišťuje zabezpečovací, přihlašovací a správní funkce. Je softwarová.“ (19)

4.1.6. Prezentační vrstva

„Specifikuje způsob, jakým jsou data formátována, prezentována, transformována a kódována. Řeší například háčky a čárky, CRC, kompresi a dekompresi, šifrování dat. Je softwarová.“ (19)

4.1.7. Aplikační vrstva

„Je to v modelu vrstva nejvyšší. Definuje způsob, jakým komunikují se síť aplikace, například databázové systémy, elektronická pošta nebo programy pro emulaci terminálů. Používá služby nižších vrstev a díky tomu je izolována od problémů síťových technických prostředků. Je softwarová.“ (19) Tato vrstva je realizována v našem případě KNXnet/IP.

Ethernet používá sběrniceovou technologii. V jednom okamžiku může vysílat jenom jeden účastník, a komunikaci přijímají všichni. Každé zařízení má přiřazenou hardwarovou adresu (MAC adresa). Informace ze sběrnice využívají jen zařízení, kterým jsou určena.

Mezi přenosová media Ethernetu patří:

- Koaxiální kabely
- Kroucená dvoulinka, nejrozšířenější druh Ethernetové kabeláže.
- Optická vlákna.

4.2. TCP a UDP protokoly

4.2.1. TCP protokol

Protokol TCP je protokol transportní vrstvy. Slouží k propojení dvou aplikací. IP protokol síťové vrstvy, nižší vrstvy, slouží k propojení dvou počítačů. Protokol TCP používá protokol síťové vrstvy protokolu IP, proto můžeme najít označení TCP/IP. Při komunikaci s TCP je nutné navázat spojení. Spojení navazuje TCP klient, který se připojuje k TCP serveru. Přenos dat probíhá pouze mezi těmito dvěma zařízeními. Data jsou potvrzovaná. Při selhání doručování dat jsou data vyslána znovu. Data vyslána přes TCP jsou přijímána ve stejném pořadí, jak byla vyslána. (20)

Potvrzovaný protokol.

Výhody TCP:

- Snadná detekce nedoručení dat
- Garance správnosti pořadí přijímaných dat.
- Nemůže vzniknout duplicita dat.
- Zajištěna správnost dat.

Nevýhody TCP:

- Příliš mnoho řídicích informací- Hlavička TCP obsahuje mnoho informací.
- Velká zátěž pro síť – vysílání potvrzovacích paketů.

4.2.2. UDP protokol

Protokol UDP je protokol transportní vrstvy. Slouží ke komunikaci dvou aplikací. Využívá IP jako síťový protokol pro spojení dvou počítačů. Identifikuje aplikace pomocí portu. UDP port je jednoznačné označení aplikace. UDP není spojová služba. Nenavazuje spojení. Nepotvrzuje se doručení UDP datagramů, jedná se o nepotvrzovaný protokol. Data mohou být doručována ve špatném pořadí. Data se mohou duplikovat. (20)

Výhody UDP:

- Malá hlavička dat
- Malé zatížení sítě

Nevýhody UDP:

- Nezabezpečený přenos dat, data se mohou ztratit, duplikovat, předbíhat.

UDP se využívá při různých „real-time“ přenosech multimediálních dat, například videokonference. Při přenášení velkých dat by bylo jejich potvrzování pro síť velice náročné. K odesílání dat na multicastových adresách se také využívá UDP protokolu. UDP se rovněž využívá ke komunikaci s DNS serverem. (20)

5. Návrh HMI modulu

K realizaci inteligentního vypínače byl vybrán mikropočítač Raspberry Pi. Jeho výhodou je, že je připraven pro práci na síti, protože disponuje síťovou kartou přímo na desce plošných spojů.

5.1. Raspberry Pi

Raspberry Pi je počítač velikostí kreditní karty o rozměrech 85.60mm x 56mm x 21mm, ke kterému je možné připojit monitor a klávesnici. Je to malý počítač, který lze použít v mnoha projektech v oblasti elektroniky, a pro další aplikace, které zvládá klasický stolní počítač: práci s daty, psaní textu, hraní her. Také zvládá přehrávat video ve vysokém rozlišení. O funkčnost Raspberry Pi se stará SoC Broadcom BCM2835 obsahující procesor ARM1176JZFS, který běží na taktu 700Mhz. Raspberry Pi disponuje 256 MB operační pamětí. Na desce také najdeme 2 USB 2.0 porty, 10/100 megabitová síťovou kartu, čtečku paměťových karet, 3,5mm zvukový výstup, kompozitní video výstup a HDMI. Pro napájení slouží microUSB konektor. Cena počítače se pohybuje pod hranici 1000 Kč. (21)

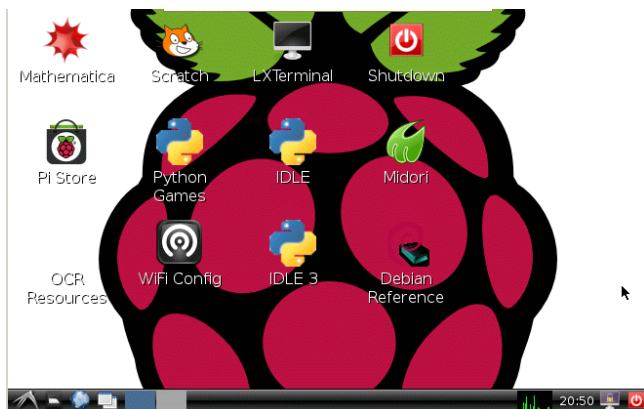


Obr. 25: Raspberry Pi (22)

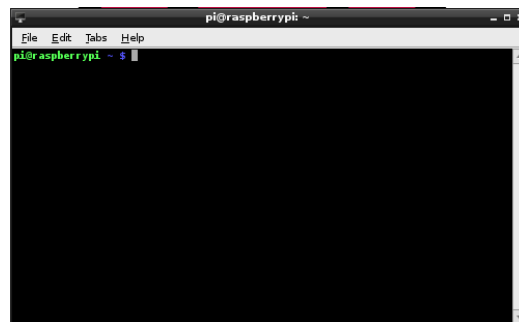
Díky univerzálnosti Raspberry jej lze využít k mnoha aplikacím. Jedná se o multifunkční zařízení s větším rozsahem využití. Jedna z mnohých je komunikace pomocí Ethernetu, díky síťové kartě, kterou disponuje. Raspberry Pi bootuje z SD karty, na kterou je možné nahrát pár linuxových distribucí. Raspberry Pi běžící na linuxové distribuci je možné ho používat jako běžný stolní počítač, a k tomu je možné připojit další periférie k GPIO konektoru. Může se jednat o tlačítka, diody, displeje atd. Připojením k televizi a připojením síťového úložiště získáme domácí multimediální centrum. Operační systém Linux zajišťuje stabilitu a jistotu provozu bez nepotřebných pádů. Další výhodou je spotřeba elektrické energie nepřesahující 5W. O napájení Raspberry se postará jakákoli nabíječka k chytrému telefonu s konektorem microUSB. Je možno také k napájení použít konvenční USB port.

5.1.1. Operační systém

Operační systém pro Raspberry byl vybrán z kompatibilních linuxových distribucí s názvem Debian. Systém je nahrán na SD kartě, z které nabootuje při startu. Systém v sobě zahrnuje rovněž grafickou podobu operačního systému, avšak v případě provozu HMI modulu není využit.



Obr. 26: grafické prostředí debian



Obr. 27: Příkazový řádek

Programování aplikace probíhalo v programovacím jazyku C#. Jedná se o objektově orientovaný programovací jazyk vytvořený firmou Microsoft zároveň s platformou .NET. Aplikace běží na rozhraní .NET Framework. Skompilované aplikace mají příponou „.exe“ a jsou nativně spustitelné jen v operačním systému Windows.

Linux nepodporuje spuštění aplikací s příponou „.exe“, proto je spuštěn pod linuxem prostřednictvím programu Mono, který umožňuje spouštět aplikace napsané v jazyku C#.

Mono:

Mono je projekt vedený firmou Novell. Mono umožňuje spuštění programu vytvořených v programovacím jazyku C# pod operačními systémy Linux, UNIX, Mac OS X, Solaris. Ke spuštění programu s příponou „.exe“, bylo potřeba tento program doinstalovat. Instalace probíhá přes příkazový řádek.

Instalace:

Po otevření příkazové řádky zadáme příkaz pro instalaci mona:

```
sudo apt-get install mono-complete
```

Tímto příkazem se mono nainstaluje a umožní spouštět programy z Windows. Napsaný program můžeme zkompileovat přímo na počítačích Windows v programu Microsoft Visual Studio, který byl využit k programování aplikace, nebo jej můžeme zkompileovat přímo v linuxovém prostředí, prostřednictvím příkazu:

```
gmcs HelloWorld.cs
```

Po kompilaci můžeme aplikaci spustit jednoduchým příkazem (spuštění aplikace s příponou „.exe“ v linuxové distribuci):

```
mono HelloWorld.exe
```

5.1.2. GPIO

Raspberry Pi disponuje GPIO portem s piny, které mohou plnit úlohu vstupních i výstupních pinů. Na obrázku č. 28 můžeme vidět pojmenování jednotlivých pinů.



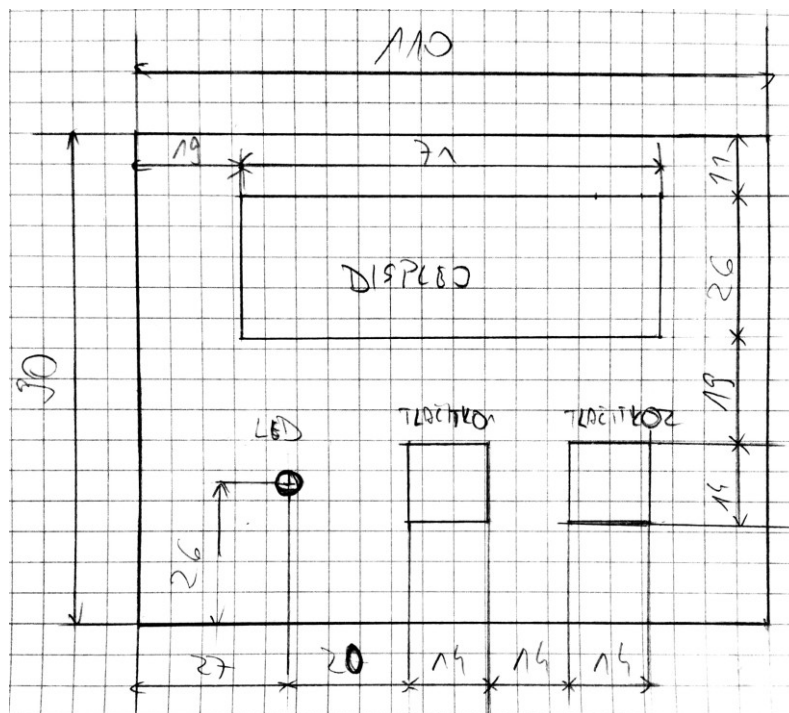
Obr. 28: GPIO piny na RaspberryPI

Přístup k pinům lze provést příkazy přes příkazový řádek nebo použít naprogramované knihovny. Práce a přístup k pinů v aplikaci napsané v jazyku C# bylo umožněné díky knihovně RaspberryPiDotNet.

5.1.3. Návrh uživatelského modulu

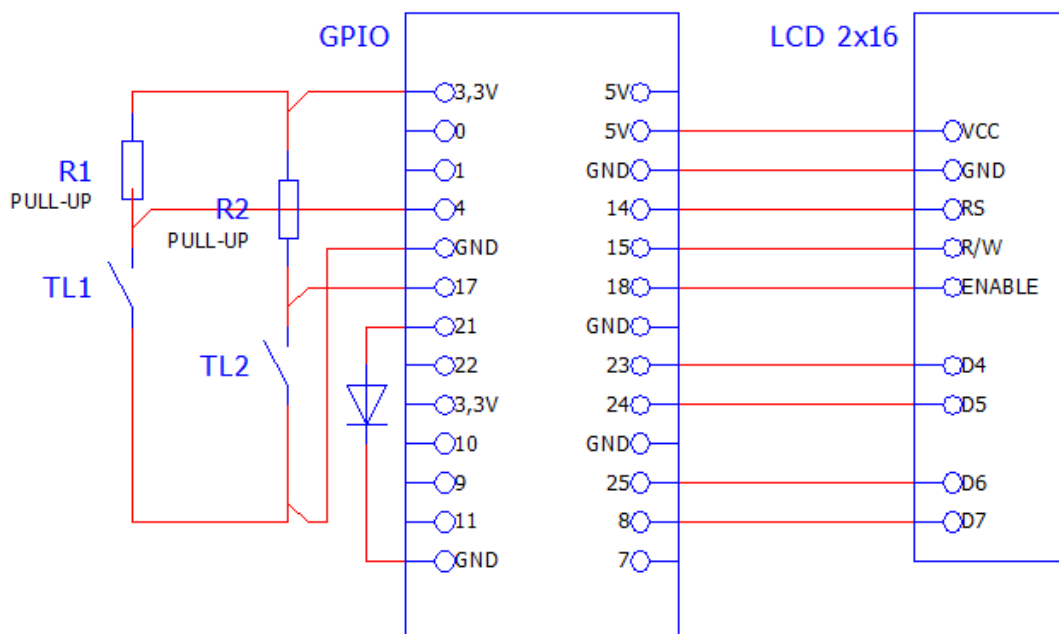
Před realizací byl navržen uživatelský panel pro ovládání světla. Navržena byla krabička s rozměry 120x70x50 mm. Rozměry byly předem stanoveny tak, aby se dovnitř vešel Raspberry Pi společně s displejem, tlačítky, led diodou a vnitřním zapojení kabelu. Na vrchní straně krabičky byly vytvořeny návrhy umístění displeje, tlačítek a led diody. Navrhnutý byly asi 4 verze, nakonec po zakoupení konkrétního typu krabičky byl vybrán jeden a přizpůsoben na rozměry krabičky 110x90x50. Dále bylo navrženo schéma zapojení ovládacích tlačítek, indikační diody a led displeje.

Návrh rozmístění prvků na krabičce:



Obr. 29: Návrh rozmístění displeje, tlačítek a led diody

Návrh schématu zapojení:



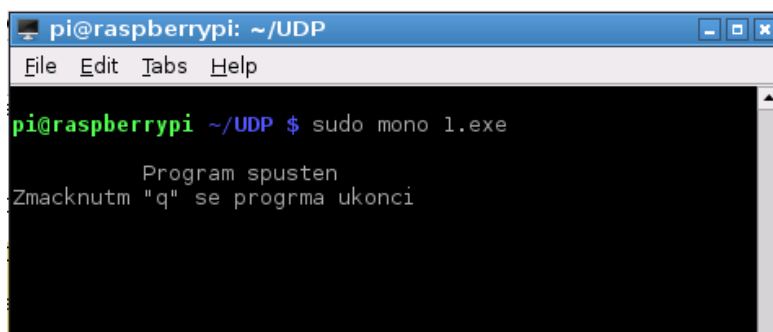
Obr. 30: Navržené schéma zapojení displeje, tlačítek a led diody

6. Realizace HMI modulu

Hlavním realizačním úkolem bylo naprogramovat aplikaci. Aplikace umožňuje komunikaci mezi Raspberry Pi připojeného k Ethernet a akčním členem připojeným na KNX sběrnici. Dalším úkolem bylo připojit tlačítka, lcd displej a led diodu k GPIO konektoru. Do krabičky bylo následně vyvrtány otvory pro prvky čelního panelu. Následně byl panel osazen tlačítky, displejem a led diodou.

6.1. Programování

Aplikace je napsána v programovacím jazyku C#. Jedná se o konzolovou aplikaci. Spouští se po startu na pozadí. Aplikaci je také možné spustit v konzoli na popředí. Zapnutí/vypnutí světla se provádí jedním tlačítkem. Stisk tlačítka světlo zapne, při opakovaném stisku se vypne. Druhé tlačítko umožňuje přepínat mezi normálním módem a pokročilým módem. Všechny stavy a průběhy jsou zobrazovány na dvouřádkovém displeji. Normální mód pouze zobrazí, zda byl telegram vyslán nebo přijat, a zda světlo svítí nebo ne. Pokročilý mód zobrazí tělo KNX telegramu společně se skutečným stavem světla. Telegram je vysílán na multicastové adrese: 224.0.23.12 a multicastovém portu 3671.



```
pi@raspberrypi: ~/UDP
File Edit Tabs Help

pi@raspberrypi ~/UDP $ sudo mono 1.exe

Program spusten
Zmáknutím "q" se progrma ukonci
```

Obr. 31: Konzole se spuštěným programem

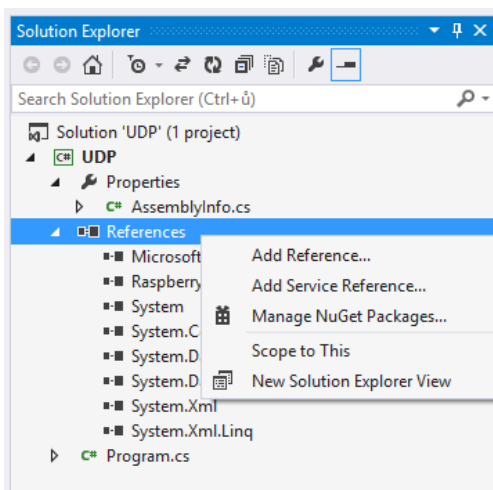
6.1.1. Knihovny

Na začátku programu jsou přiřazeny knihovny. Jedná se o systémové knihovny, knihovny umožňující ukládání a čtení souboru, spouštění vláken, odesílání a přijímání dat, a také práce s GPIO konektorem.

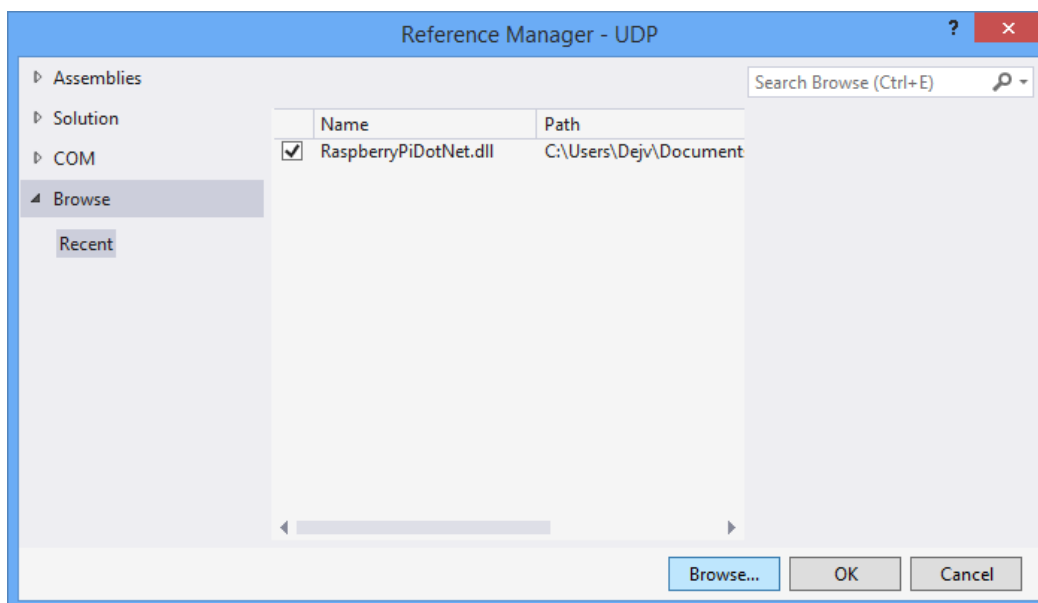
Knihovny programu:

<code>using System;</code>	- systémová Knihovna
<code>using System.IO;</code>	- knihovna umožňující práci se soubory
<code>using System.Collections.Generic;</code>	
<code>using System.Linq;</code>	
<code>using System.Text;</code>	
<code>using System.Net;</code>	
<code>using System.Net.Sockets;</code>	- knihovna umožňující odesílání paketů
<code>using System.Threading;</code>	- knihovna pro práci s vlákny
<code>using RaspberryPiDotNet;</code>	- knihovna pro práci GPIO konektorem
<code>using RaspberryPiDotNet.MicroLiquidCrystal;</code>	- knihovna pro práci s lcd displejem

V programu musí být přiřazena reference na knihovnu RaspberryPiDotNet.dll. Přidání reference se provede ve stromu programu. Pravým kliknutím myši na References a kliknutí na Add Reference. Rozbalí se okno, ve kterém klikneme na browse a vybereme umístění dll knihovny. Potvrdíme a referenci na knihovnu můžeme vidět v záložce References.



Obr. 32: Přidáním reference na dll knihovnu



Obr. 33: Přidáním reference kliknutím na tlačítko Browse

Nyní můžeme vytvořit instanci lcdProvide třídy `RaspPiGPIOMemLcdTransferProvider`, která nám poslouží k inicializaci lcd displeje:

```
static RaspPiGPIOMemLcdTransferProvider lcdProvider;
```

dále vytvoříme instanci lcd třídy Lcd sloužící pro výpis textu na displej.

```
static Lcd lcd;
```

Vytvoření instance `button`, `button1` a `led` třídy `GIOMem` budou sloužit k ovládní pinů GPIO konektoru:

```
static GIOMem button, button1, led;
```

V programu si nadefinujeme další statické proměnné potřebné k běhu programu. Vytvořené jsou statické proměnné, protože je potřebujeme používat ve více vláknech.

Vytvoření nové instance vlákna v instanci `butt1` s parametrem názvem metody `butt`.

```
Thread butt1 = new Thread(butt);
```

Spuštění vlákna dojde příkazem:

```
butt1.Start();
```

6.1.2. Multicast komunikace

6.1.2.1. Odesílání

Pro odesílání dat na multicast adrese se musíme nejdříve připojit do multicast skupiny. Začneme založením nové instance `UdpClient` do instance `client`.

```
client = new UdpClient();
```

Dále dojde k vložení multicast adresy (224.0.12.23) do instance `multicastaddress` a následné připojení do multicas skupiny:

```
IPAddress multicastaddress = IPAddress.Parse(MULTICAST_ADDRESS);  
client.JoinMulticastGroup(multicastaddress);
```

6.1.2.2. Přijímání

Kód pro přijímání dat z multicastové skupiny je podobný kódu pro přijímání. Musíme ještě přidat instanci `localEp` s parametry umožňující přijímání z jakékoli adresy, avšak pouze z určitého portu, následně zavolat metodu `SetSocketOption`, neboť `localEp` a `SetSocketOption` jsou nezbytné pro příjem dat.

```
client = new UdpClient();  
IPEndPoint localEp = new IPEndPoint(IPAddress.Any, MULTICAST_PORT);  
IPAddress multicastaddress = IPAddress.Parse(MULTICAST_ADDRESS);  
client.Client.SetSocketOption(SocketOptionLevel.Socket,  
SocketOptionName.ReuseAddress, true);  
client.JoinMulticastGroup(multicastaddress);
```

6.1.3. Skládání telegramu

Sepnutí světla je realizováno vysláním datového telegramu, který má v sobě datový bit, který realizuje logickou 1 nebo 0. Při log. 1 světlo svítí. Aplikace umožňuje editaci cílové adresy

akčního členu světla. Editace cílové adresy se provádí v příkazovém řádku zmáčknutím klávesy „u“. Změna se zapisuje do souboru a volá se při každém spuštění programu.

6.1.3.1. Telegram pro zapnutí světla

Telegram nesoucí informaci o zapnutí světla se skládá podle předem daného složení KNX/IP telegramu. KNX IP hlavička obsahuje nejprve délku hlavičky 06h. Jediná platná verze revize KNXnet/IP je 1.0 (10h). Nadefinovaný KNXnet/IP service je Routing indication 0530h, který se používá k odesílání telegramů na IP síť. V IP hlavičce vyzývá doplnit celkovou délku telegramu včetně dat, v našem případě se jedná o 17 bytů (0011h).

Následuje cEMI zpráva, v prvním bytů se definuje, o jakou zprávu se jedná. Bude se jednat o L_Data.ind sloužící k přenosu dat v telegramu (29h). Délku dodatečné informace je zvolena 0h. V prvním kontrolním poli je zvolen standartní telegram, telegram se neopakuje, dojde-li k chybě a je přenášen pomocí broucastu. Priorita telegramu je nastavena na low a nepožaduje se potvrzení. Výsledné 1. kontrolní pole: (BCh).

V druhém kontrolním poli je nastavena cílová adresa jako společná, routingové číslo je nastaveno na 5. a jedná se o standartní telegram (0h). 2. Kontrolní pole: (D0h).

Zdrojová adresa zařízení je nastavena na 0.1.2 (01 02h), cílová adresa 1.1.1(09 01h).

Další byte určuje délku dat, budeme přenášet pouze jeden bit, takže délka se bude rovnat 1h. Byte přenášející užitečnou informaci se doplní příslušným bitem podle toho, zda se jedná o zapisování nebo pouze čtení stavu. Z aplikační vrstvy, pro zápis se volí 0001b. Výsledný byte nesoucí informaci 1bit k zapnutí světla (81h).

KNX IP hlavička: 06 10 05 30 00 11

cEMI zpráva: 29 00

Kontrolní pole 1: 1011 1100 v hexa: BC

Kontrolní pole 2: 1101 0000 v hexa: D0

Zdrojová adresa: 0.1.2 v hexa 01 02

Cílová adresa: 1.1.1 v hexa 09 01

Data: 01 00 81

Výsledný telegram: 06 10 05 30 00 11 29 00 BC D0 01 02 09 01 01 00 81

6.1.3.2. Telegram pro vypnutí světla

Telegram pro vypnutí světla je stejný s telegramem pro zapnutí světla s tím rozdílem, že poslední bit je změněn a to z 1 na 0.

KNX IP hlavička: 06 10 05 30 00 11

cEMI zpráva: 29 00

Kontrolní pole 1: 1011 1100 v hexa: BC

Kontrolní pole 2: 1101 0000 v hexa: D0

Zdrojová adresa: 0.1.2 v hexa 01 02

Cílová adresa: 1.1.1 v hexa 09 01

Data: 01 00 80

Výsledný Telegram: 06 10 05 30 00 11 29 00 BC D0 01 02 09 01 01 00 80

6.1.3.3. Telegram pro zjištění stavu světla

Telegram pro zjištění stavu světla je využíván při startu aplikace. Aplikace si zjistí, jaký je stav světla a podle toho, v jakém stavu se nachází, vybere vždy ten opačný. V případě rozsvíceného světla, světlo vypne. Telegram se liší v tom, že v aplikační vrstvě se dosadí 0000b místo 0001b. Jednička na konci znamená zápis, to znamená zapsání do akčního členu akci - vypnout nebo zapnout. Při dosazení 0 akční člen odešle svůj stav.

Telegram: 06 10 05 30 00 11 29 00 BC D0 01 02 09 01 01 00 00

6.1.3.4. Odesílání telegramů

Odeslání telegramu probíhá zavoláním metody Send a zadáním parametrů:

- ZAP – telegram pro zapnutí světla,
- ZAP.lenght – délka telegramu,
- MULTICAST_ADDRESS – multicastová adresa,
- MULTICAST_PORT – multicastový port.

```
client.Send(ZAP, ZAP.Length, MULTICAST_ADDRESS, MULTICAST_PORT);
```

Stejně tak můžeme vyslat telegram pro vypnutí světla, bude se lišit pouze v záměně telegramu ZAP za VYP:

```
client.Send(VYP, VYP.Length, MULTICAST_ADDRESS, MULTICAST_PORT);
```

6.1.3.5. Přijímání telegramů

Pro přijímání dat se zavolá metoda Receive s parametrem, který určuje příjem na kterékoli adrese, avšak na definovaném KNX portu.

```
Byte[] data = client.Receive(ref localEp);  
string hex = BitConverter.ToString(data).Replace("-", string.Empty);
```

Data následně převedeme na datový typ string a můžeme vypsát na displej, do konzole, uložit atp.

6.2. Tlačítka

Tlačítka jsou připojena k GPIO konektoru. K tlačítkům je připojen pull-up rezistor. V aplikaci se čeká, kdy se změní hodnota tlačítka z true na false.

Vytvoření nové instance třídy v proměnné `button`, s parametry nastavující GPIO pin 04 jako výstupní.

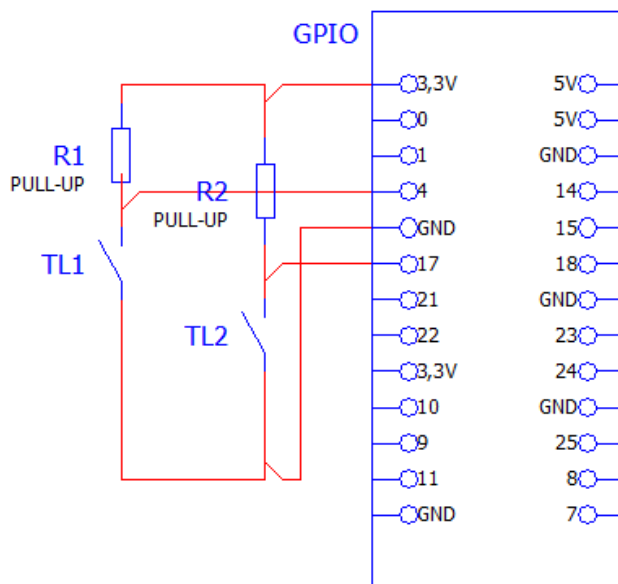
```
GPIO Mem button = new GPIO Mem(GPIO Pins.GPIO_04, GPIO Direction.In);
```

Stav tlačítka se zjišťuje metodou `Read`. Stav tlačítka se zapíše do proměnné `m`.

```
m = button.Read();
```

Zjištění stavu tlačítka je realizováno v samostatném vlákne, při přechodu hodnoty tlačítka z true do false se odešle telegram. Jaký telegram aplikace vyšle, závisí na stavu světla. Aplikace si vždycky zjišťuje po odeslání telegramu, zda došlo doopravdy k požadované akci. Je-li tomu tak, nastaví se tlačítko pro odeslání opačné akce. Po stisknutí tlačítka se připravený telegram vyšle a vše se opakuje znova. Níže je zobrazeno schéma zapojení tlačítek.

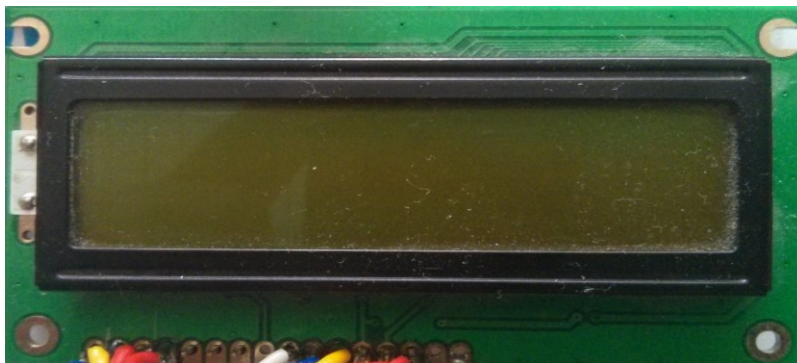
Druhé tlačítko slouží k přepínání mezi normálním a pokročilým módem. Normální mód vizualizuje vysílání a přijímání telegramu. Na displeji zobrazí například „odesláno: zap“, nebo „přijato: vyp“. Pokročilý mód zase zobrazuje tělo KNX telegramu, uživatel tak může mít přehled, který telegram vysílá a který naopak vypínač přijal.



Obr. 34: Chéma zapojení tlačítek

6.3. Displej

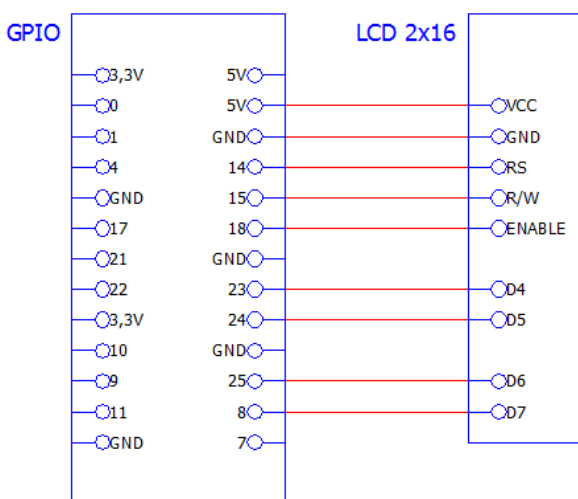
Displej je dvouřádkový s řadičem HD44780. Displej je připojen ke GPIO konektoru, z kterého je také napájen. Zobrazování na displeji probíhá díky knihovně RaspberryPiDotNet. Knihovna umožňuje vypisovat text na displej. Displej přijímá data přes čtyři datové vodiče a tři jsou řídicí.



Obr. 35: Display s řadičem HD44780

Piny displeje:

1. Ground
2. VCC (+3.3 to +5V)
3. Contrast adjustment (VO)
4. Register Select (RS). RS=0: Command, RS=1: Data
5. Read/Write (R/W). R/W=0: Write, R/W=1: Read
6. Clock (Enable). Falling edge triggered
7. Bit 0 (Not used in 4-bit operation)
8. Bit 1 (Not used in 4-bit operation)
9. Bit 2 (Not used in 4-bit operation)
10. Bit 3 (Not used in 4-bit operation)
11. Bit 4
12. Bit 5
13. Bit 6
14. Bit 7



Obr. 36: Schéma zapojení displeje k GPIO pinům

Založením nové instance `RaspPiGPIOMemLcdTransferProvider` v instanci `lcdProvider` s parametry za sebou sepsaných pinů se provede inicializace lcd displeje.

Piny jsou přiřazovány v pořadí: `rs`, `rw`, `enable`, `d4`, `d5`, `d6`, `d7`.

```
RaspPiGPIOMemLcdTransferProvider lcdProvider = new RaspPiGPIOMemLcdTransferProvider(  
    GPIOPins.GPIO_14,  
    GPIOPins.GPIO_15,  
    GPIOPins.GPIO_18,  
    GPIOPins.GPIO_23,  
    GPIOPins.GPIO_24,  
    GPIOPins.GPIO_25,  
    GPIOPins.GPIO_08);  
Lcd lcd = new Lcd(lcdProvider); - vložení nové instance LCD s parametrem  
lcdProvider do instance lcd;
```

Po přiřazení pinů a vytvoření instance `lcd` přijde na řadu nastavení typu displeje (16 znaků, 2 řádky). Nastaví se zavoláním metody `Begin`. Poté se vymaže displej zavoláním metody `Clear`. Nastavení pozice kurzoru na 4. pozici nultého řádku se provede metodou `SetCursorPosition`. Následně se metodou `Write` vypíše „Zdravím“

```
lcd.Begin(16, 2); //nastavení typu displeje 16x2  
lcd.Clear(); //vymazání displeje  
lcd.SetCursorPosition(4, 0); //nastavení kurzoru na 4 pozici nultého  
řádku  
lcd.Write("Zdravim"); //výpis „Zdravím“ na displej
```

Signalizace zapnutého a vypnutého světla je indikováno na displeji v pravém dolním rohu. Písmeno „X“ znamená zapnuté světlo, pomlčka „-“ znamená vypnuté světlo. Indikace světla je také indikováno led diodou.

6.4. Soubor

Aplikace ukládá do souboru aktuální cílovou adresu. Cílová adresa se do souboru zapíše vždy po novému vložení adresy. Aplikace si pak sama načte po startu naposledy používanou cílovou adresu.

```
using (StreamReader sr = new StreamReader(@"adresa.txt"))
```

Vytvoří se instance „sr“ s vloženou novou instancí `StreamReader` a parametrem názvu textového souboru, z kterého se bude adresa načítat.

Načte se první polovina čísla adresy zavoláním metody `ReadLine`:

```
ddd = Convert.ToByte(sr.ReadLine());
```

Načte se druhá polovina čísla adresy:

```
ddd1 = Convert.ToByte(sr.ReadLine());
```

6.5. Indikační led dioda

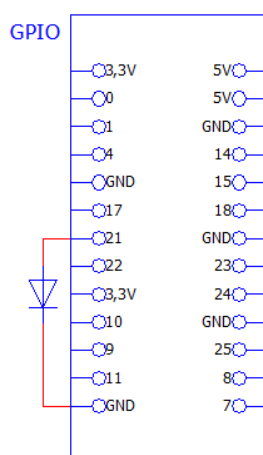
Led dioda indikuje reálný stav světla. Když je světlo zapnuto, led dioda svítí. Aplikace využívá zpětně zasílané telegramy indikující, ve kterém stavu se akční člen nachází, k zapínání a vypínání led diody.

Nejprve se musí přiřadit pin pro led diodu a nastavit pin jako výstupní.

```
led = new GPIOMem(GPIOPins.GPIO_22, GPIODirection.Out);
```

Zapnutí diody se provádí zavoláním metody Write s parametrem true. Výsledkem je rozsvícení diody.

```
led.Write(true);
```



Obr. 37: Schéma zapojení led diody

6.6. HMI modul



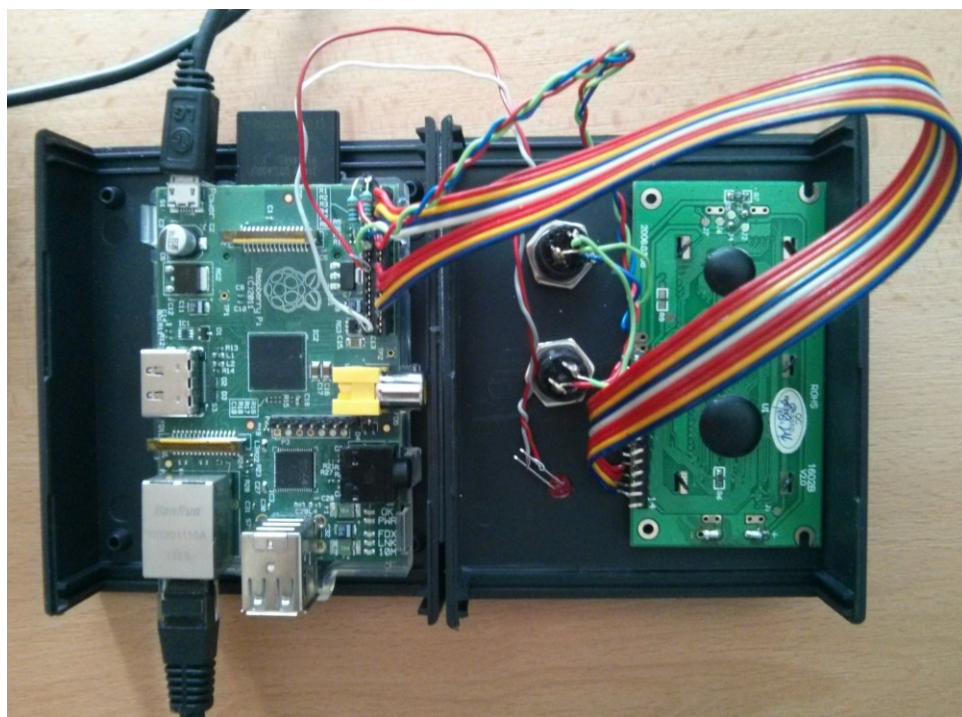
Obr. 38: Zapojené a spuštěné Raspberry Pi

Realizovaný funkční HMI modul:



Obr. 39: HMI modul

Funkčnost byla otestována na reálném zapojení. Komunikace mezi HMI modulem a akčním členem probíhala bez problému. Akční člen vždy vykonal požadovanou akci. Níže je zobrazeno vnitřní zapojení HMI modulu.



Obr. 40: Podhled dovnitř HMI modulu

7. Závěr

Cílem bakalářské práce bylo realizovat inteligentní vypínač (HMI modul) pro komunikaci přes KNXnet/IP. Předmětem bylo navrhnout zařízení, na kterém bude spuštěná aplikace, která bude vykonávat veškerou komunikaci a vizualizaci. Navržen byl malý počítač Raspberr PI disponující Armovým procesorem, umožňující spuštění linuxové distribuce Debian. K tvorbě aplikace byl použit programovací jazyk C#. Spuštění aplikace s příponou „.exe“ pod Linuxovou distribucí umožňuje program Mono. Aplikace se automaticky spustí po startu Raspberry. Skládání telegramu a následné vyslání je realizováno v naprogramované aplikaci.

Byl navržen uživatelský modul, který obsahuje displej, tlačítka a led diodu. Inteligentní vypínač obsahuje dva tlačítka. První tlačítko odesílá telegram na sběrnici, zapíná a vypíná tím akční člen na knx sběrnici a spíná světlo. Adresa akčního členu je předem nadefinována v programu a uložena v souboru, ze kterého se při spuštění programu načte poslední používaná adresa. Editace adresy rovněž umožňuje program, provádí se tak v konzoli. Komunikace probíhá přes Ethernetovou síť. Raspberry je připojen k síti kabelem zapojeným v síťové kartě. Telegramy ze sítě přijímá KNX IP Router, který dokáže komunikovat jak na síti, tak na knx sběrnici. Router telegramy předává z jedné sběrnice na druhou. Akční člen připojen na knx sběrnici telegram přijme, vyhodnotí, provede žádanou akci a odešle potvrzující telegram o stavu akčního členu zpátky. Raspberry jej přijme a na displeji se zobrazí stav světla. Stav světla je také indikován led diodou. Displej vizualizuje komunikaci mezi vypínačem a akčním členem. Na displeji se zobrazí vždy aktuální situace, vysílání či přijímání telegramu. Přepnutím druhým tlačítkem se dostaneme do pokročilého módu, který zobrazuje tělo KNXIP telegramu. Tlačítka, displej a led dioda jsou GPIO konektor mikropočítače. Vše je poskládáno do navržené krabičky. Raspberry Pi je napájeno z adaptéru dodávajícího 5V.

Použitá literatura

- [1.] MERZ, Hermann, HANSEMANN, Thomas a HÜBNER, Christof. *Automatizované systémy budov: sdělovací systémy KNX/EIB, LON a BACnet*. Praha : Grada Publishing, a.s., 2008. str. 264. ISBN 978-80-247-2367-9.
- [2] Inteligentní domy. *elektron*. [Online] [Citace: 6. 5 2014.] <http://www.elektrontrade.cz/inteligentni-domy/>.
- [3] Stránky asociace KNX. [Online] [Citace: 5. 5 2014.] <http://www.knx.org/knx-en/knx/association/what-is-knx/index.php>.
- [4] Inteligentní elektroinstalace na výstavě AMPER 2010. *tzb info*. [Online] [Citace: 5. 5 2014.] <http://www.tzb-info.cz/veletrh-amper/6388-inteligentni-elektroinstalace-na-vystave-amper-2010>.
- [5] Architektura, komunikace a adresování sběrnice KNX/EIB. *ElektroPrůmysl.cz*. [Online] [Citace: 6. 5 2014.] <http://www.elektroprumysl.cz/elektroinstalace/architektura-komunikace-a-adresovani-sbernice-knx-eib>.
- [6] ABB. [Online] [Citace: 5. 5 2014.] <http://www.abb.cz/>.
- [7] Solar Vable. *Solar*. [Online] [Citace: 6. 5 2014.] <http://www.solarcable.no/product/59698>.
- [8] SKOPAL, Dušan. *Ovládání provozně technických funkcí pomocí sběrnicevého systému KNX. Ostrava, 2012. Bakalářská práce. VŠB - Technická univerzita Ostrava*.
- [9] *Základní kurz KNX*.
- [10] Weinzierl, Dr.-Ing. Th. “KNX IP only” – A New Class of KNX Devices. www.weinzierl.de. [Online] [Citace: 14. 1 2014.]
- [11] Association, KNX. *03_08_02 Core v01.05.01 AS.pdf*. 2013.
- [12] Association, KNX. *03_08_05 Routing v01.05.01 AS.pdf*. 2013.
- [13] Association, KNX. *03_06_03 EMI_IMI v01.03.03 AS.pdf*. 2013.
- [14] KNX. [Online] [Citace: 6. 5 2014.] <http://www.knx.org>.
- [15] PDF KNX Twisted Pair Protokollbeschreibung. *mikrocontroller.net*. [Online] [Citace: 5. 5 2014.] http://www.mikrocontroller.net/attachment/151008/KNX_Twisted_Pair_Protokollbeschreibung.pdf.
- [16] Heiny, Dipl.-Ing. (FH) Florian. Workshop. [Online] [Citace: 10. 2 2014.] www.weinzierl.de.
- [17] Lance Hemmings. Ethernet a USB společně propojují průmyslové systémy. *Electronic Engineering magazine*. [Online] [Citace: 14. 1 2014.] <http://infocube.cz/cs/ethernet-a-usb-spolecne-propojuji-prumyslove-systemy/>.
- [18] Zaneta, Havlickova. Ethernet. [Online] [Citace: 4. 5 2014.] <http://home.zcu.cz/~zaneta/>.

[19] Počítačové sítě - Model ISO/OSI. *počítačové sítě*. [Online] [Citace: 6. 5 2014.]
<http://site.the.cz/index.php?id=4>.

[20] Protokol UDP 1.část. *BUILDER*. [Online] [Citace: 6. 5 2014.]
<http://www.builder.cz/rubriky/c/c--/protokol-udp-1-cast-156226cz>.

[21] Introduction: What is a Raspberry Pi? *Raspberry Pi*. [Online] [Citace: 14. 1 2014.]
<http://www.raspberrypi.org/faqs#introWhatIs>.

[22] My Goal. *Raspberry Pi School*. [Online] [Citace: 6. 5 2014.]
<http://www.raspberrypischool.org.uk/?p=52>.

Seznam Příloh

A	Zobrazování displeje	I
B	Obsah DVD	II

A. Zobrazování displeje



Obr. 40: Odesláno Zapnutí



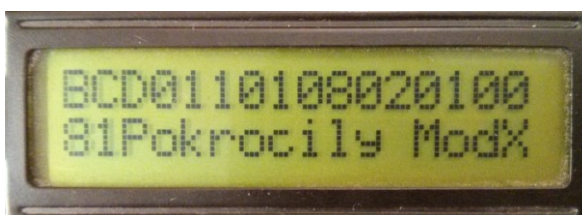
Obr. 40: Přijato Zapnutí



Obr. 40: Odesláno Vypnutí



Obr. 40: Přijato Vypnutí



Obr. 40: Pokročilý mód – světlo svítí



Obr. 40: Pokročilý mód – světlo je vypnuté

B. Obsah DVD

Zde je popsán obsah přiloženého DVD.

- Program napsaný v jazyce C# s názvem HMImodul.cs
- Spustitelná aplikace s názvem HMImodul.exe
- Knihovna RaspberryPiDotNet.dll
- Image SD karty
- Fotodokumentace k HMI modulu
- Bakalářská práce ve formátu PDF