



Klaus Nurkkala

**RGB-LED -VALAISTUKSEN OHJAAMINEN LINUX-  
PALVELIMELLA**

# **RGB-LED -VALAISTUKSEN OHJAAMINEN LINUX- PALVELIMELLA**

Klaus Nurkkala  
Opinnäytetyö  
Kevät 2011  
Automaatiotekniikan koulutusohjelma  
Oulun seudun ammattikorkeakoulu

## OULUN SEUDUN AMMATTIKORKEAKOULU

## TIIVISTELMÄ

Koulutusohjelma  
Automaatiotekniikka

Opinnäytetyö  
Insinöörityö

Sivuja + Liitteitä  
38 + 5

Suuntautumisvaihtoehto

Aika

Projektoinnin sv.

2011

Työn tilaaja

Työn tekijä

Cornier Oy

Klaus Nurkkala

Työn nimi

RGB-LED -valaistuksen ohjaaminen Linux-palvelimella

Avainsanat

RGB-LED-valaistus, kodin automaatio, X10, Z-Wave, KNX, Openremote, LinuxMCE, MisterHouse

Insinöörityössä toteutettiin internet-selaimen kautta ohjattava RGB-LED-tunnelmavalaisin pienitehoisen Linux-palvelimen ja Arduino-protoalustan avulla. Samalla tutustuttiin avoimiin kodin automaatio-sovelluksiin ja niihin soveltuviin tietoliikennestandardeihin.

Työ aloitettiin hankkimalla sopivat laitteet ja ohjelmistot, tutustuttiin sarjaliikenne-rajapintoihin, Arduino- ja Python-ohjelmointikieleen ja N-tyypin MOSFETien toimintaan. Arduino-ohjelmointikielellä, joka perustuu C/C++-ohjelmointikieleen, luotiin mikrokontrollerille RGB-LED valaisinohjausohjelma ja vastaavasti pienellä Linux-palvelimella otettiin käyttöön Pythonilla luotu RGB-LED-valaistusta ohjaava sarjaliikenneohjelma. Käyttöliittymäksi valittiin suoraan tietokoneelta ohjattava Python-ohjelma. Vaihtoehtoisena järjestelmänä voi käyttää internet-selaimella toimivaa jQueryä sekä HTML:ää hyödyntävää Web-ohjausjärjestelmää. Näiden avulla luotiin tunnelmavalaisin, jossa käyttäjän valittavana on yli 16,7 miljoonaa eri värisävyä. Työn loppuosa keskittyi Linuxilla toimivien kodin automaatio-ohjelmien tutkimiseen.

# SISÄLTÖ

TIIVISTELMÄ .....	3
SISÄLTÖ .....	4
LYHENTEET JA TERMIT .....	6
1 JOHDANTO .....	7
2 TIETOLIIKENNESTANDARDEJA.....	8
2.1 X10.....	8
2.2 Z-Wave.....	8
2.3 KNX.....	9
3 KÄYTETYT LAITTEET JA TOIMINTAPERIAATTEET.....	10
3.1 Guruplug plus Linux-minipalvelin.....	10
3.2 Arduino Duemilanove-protoalusta .....	11
3.3 Liiketunnistin (PIR) .....	11
3.4 Pulssinleveysmodulointi .....	12
3.5 RGB-LED .....	13
3.6 N-tyypin MOSFET .....	14
4 RGB-LED-VALAISTUKSEN TOTEUTUS .....	15
4.1 RGB-LED-valaistuksen ohjaus selaimen kautta.....	15
4.2 RGB-LED-valaistuksen ohjaus suoraan tietokoneelta.....	17
4.3 Mikrokontrollerin sovellus .....	19
4.4 Kytöntäkaavio (liite 5) .....	19
4.5 Tarkemmat asetustiedot.....	20
4.5.1 Guruplug ja ohjaavan tietokoneen udev-asetukset.....	20
4.5.2 Python-serial-ohjelma.....	21
4.5.3 Web käyttöliittymän asentaminen.....	21
4.5.4 Web käyttöliittymän sarjaliikenerajapinta.....	21
4.5.5 Valon ohjaus suoraan tietokoneelta.....	21
4.6 Lopputulos.....	21
5 AVOIMEN LÄHDEKOODIN KODIN AUTOMAATIO-SOVELLUKSIA .....	23
5.1 LinuxMCE.....	23
5.1.1 LinuxMCE Orbiter .....	24
5.1.2 LinuxMCE-tuetut tietoliikennestandardit .....	24
5.2 OpenRemote .....	25
5.2.1 OpenRemote Designer .....	25

5.2.2 OpenRemote tuetut tietoliikennestandardit.....	26
5.3 Misterhouse.....	26
5.3.1 MisterHouse tuetut tietoliikennestandardit.....	27
6 TULOKSET JA JOHTOPÄÄTÖKSET .....	28
6.1 Guruplug plus .....	28
6.1.1 LinuxMCE mini ja Guruplug.....	28
6.1.2 OpenRemote ja Guruplug.....	28
6.1.3 MisterHouse ja Guruplug .....	29
6.2 RGB-LED .....	29
6.3 Arduino-protoalusta .....	29
6.4 Tietoliikennestandardit .....	29
6.4.1 X10 .....	30
6.4.2 Z-Wave .....	31
6.4.3 KNX .....	32
6.4.4 Yhteenveto tutkituista tietoliikennestandardeista.....	32
6.5 Linux- kodin automaatio-sovellukset .....	33
7 POHDINTA .....	34
LÄHTEET.....	36
LIITELUETTELO.....	38
LIITTEET	

## LYHENTEET JA TERMIT

IR Infra Red, infrapuna

LED Light Emitting Diode, valoa emittoiva diodi

PL PowerLine, sähköjohto / voimalinja

PWM Pulse Width Modulation, pulssinleveysmodulaatio

RAM Random Acces Memory, keskusmuisti

RF Radio Frequency, radiotaajuus

RGB Red Green Blue, punainen vihreä sininen

TP Twisted Pair, kierretty parikaapeli

USB Universal Serial Bus, sarjaväylä oheislaitteiden liittämiseen tietokoneeseen

DVR Digital Video Recorder, digitaalinen videotallennin

WLAN Wireless Local Area Network, langaton lähiverkko

# 1 JOHDANTO

LED-valaisimien hyötyjä ovat pitkä käyttöikä ja energiatehokkuus. RGB-LED (Red, Green, Blue) on laite, jossa on kolme lähekkäin sijoitettua kolmen eri värin LEDiä. Tällä järjestelyllä saadaan tuotettua yli 16,7 miljoonaa eri värisävyä. RGB-LED-valoilla on mahdollista elävöittää julkisia rakennuksia, siltoja, aitoja, seiniä, patsaita, ja vastaavasti kodeissa julkisivuja, pihan eri elementtejä sekä sisätiloissa sisustuksellisia kohteita. Värivalaistuksella voidaan vaikuttaa voimakkaasti kohteen tunnelmaan ja viihtyvyyteen. Merkitys korostuu paikoissa, joissa valaistus on luonnostaan vähäinen tai muutoin edustuksellisen ulkoasun toteuttaminen on haasteellista. RGB-LED-valaistuksen käyttöä rajoittaa lähinnä mielikuvitus. (1)

Työn alussa käydään läpi X10-, KNX- ja Z-Wave -kodin automaatiotietoliikennestandardveja. Tämän jälkeen käsitellään RGB-LED-valaistuksen ohjauksessa käytettyjä laitteita ja tekniikoita, minkä yhteydessä esitellään myös varsinainen opinnäytetyön tekninen osa, RGB-LED-valaistuksen ohjaus Linux-palvelimen avulla.

Lopuksi perehdytään LinuxMCE, OpenRemote ja Misterhouse avoimien lähdekoodien kodin automaatiosovelluksiin ja käsitellään niiden yhteensopivuutta eri tietoliikennestandardien kanssa.

Tämän työn tavoitteena oli luoda Linuxilla toimiva RGB-LED-valaisinohjausjärjestelmä kodin käyttöön sekä tutkia avoimen lähdekoodin kodin automaatiosovelluksia ja niihin sopivia tietoliikennestandardveja kodin automatisointiin.

RGB-LED valaistusohjauksratkaisu on toteutettu avoimen lähdekoodin periaatteella, joten luodut ja muokatut lähdekoodit on vapaasti käytettävissä GPL-lisenssin mukaisesti.

## **2 TIETOLIIKENNESTANDARDEJA**

Tässä luvussa käydään läpi Linux kodin automaatio-ohjelmille soveltuvia tietoliikennestandardeja.

### **2.1 X10**

X10 on vanhimpia kodin automaation tietoliikennestandardeja. Pico Electronics kehitti tämän vuonna 1975 kodin sähkölaitteiden etäohjaukseen ja hallintaan.

X10-ohjausviestit voidaan lähettää joko kodin sähköverkon kautta tai langattomasti. Langattomaan tiedonsiirtoon on olemassa näihin sopivat lähettimet ja vastaanottimet. Sama pätee sähköverkon kautta viestejä lähettävälle ja vastaanottaville laitteille.

Sähköverkon kautta lähettyt viestit kulkevat purskeina vaiheen ollessa 0. Jännitteen käydessä 0 V 50 Hz:n sähköverkossa saadaan siirrettyä yksi bitti 120 kHz:n viestin kantotaajuudella. Langaton lähetin lähettää viestin Euroopan toimilaitteilla 433 MHz:n taajuudella ilman salausta. (2)

### **2.2 Z-Wave**

Z-Wave on matalaenerginen langaton tiedonsiirtoprotokolla, joka on suunniteltu kodin automaatiolle. Euroopassa Z-Wave käyttää tiedonsiirtoon matalaenergistä 868,42 MHz:n taajuutta. Tiedonsiirrolle on asetettu yhden prosentin syklin raja, mikä tarkoittaa, että viestejä voi lähettää yhden prosentin ajasta. Verkko toimii silmukkaverkkoperiaatteella, jossa laite verkossa voi lähettää ja vastaanottaa välittäjäsilman kautta tietoa ohjauskeskukseen, joten suoraa yhteyttä lähettimen ja vastaanottimen välille ei välttämättä tarvita.



Toimilaitteet toimivat paristoilla tai suoraan sähköverkon kautta. Ohjaus voi tapahtua monesta lähteestä, kuten kaukosäätimestä, ohjauspaneelistä ja ohjauskeskuksesta. Ohjauskeskus ei ole välttämätön, laitekoonpano voi koostua esim. kaukosäätimestä ja ohjattavista laitteista. (3)

## 2.3 KNX

KNX Association on KNX-teknologian luoja ja omistaja. KNX on maailmanlaajuisesti hyväksytty avoin standardi koti- ja kiinteistöohjausten sovelluksiin ulottuen valaistus ja verho-ohjauksista lämmitykseen, ilmastointiin, jäähdytykseen, valvontaan, hälytyksiin, veden hallintaan, energiakulutuksen ohjaukseen, mittarointiin, kuten myös kotitalouksien ääni- ja AV-laiteohjauksiin sekä paljon muuhun. Tekniikkaa voidaan hyödyntää niin uusissa kuin jo olemassa olevissa asuin- ja liikekiinteistöissä. (4.)

KNX on maailmanlaajuisesti hyväksytty standardi kiinteistöohjauksiin, jossa on

- yksi, valmistajista riippumaton suunnittelu ja käyttöönotto työkalu (ETS)
- kattava valikoima tuettuja viestintätapoja (TP = kierretty parikaapeli, PL = sähköjohdin, RF = langaton, IP = internet protokolla ja IR = infrapuna)
- kattava valikoima tuettuja käyttöönotto-tiloja: järjestelmä-, helppokäyttö- ja automaattitila ("Auto, Easy ja System mode") (4).

## 3 KÄYTETYT LAITTEET JA TOIMINTAPERIAATTEET

Tässä luvussa perehdytään Guruplug plus Linux-minipalvelimeen, Arduino Duemilanove -protoalustaan, liiketunnistimeen, pulssinleveysmodulointiin, RGB-LED-valoon ja N-tyypin MOSFET-transistoriin. Näitä laitteita ja tekniikoita käytettiin työn toteuttamisessa.

### 3.1 Guruplug plus Linux-minipalvelin

Guruplug plus (kuva 1) on Globalscale Technologies INC. valmistama Linux-käyttöjärjestelmällä toimiva minipalvelin. Tämän etuina ovat pieni koko, pieni tehon kulutus (5 W), edullisuus, avoimuus ja äänettömyys passiivisen jäähdytyksen ansiosta. Tärkeimmät ominaisuudet on listattu alla:

- Suoritin: Marvel Kirkwood 6281 1.2 Ghz
- Keskusmuisti: 512 MB 16 bit DDR2 800 MHz
- NAND Flash: 512 MB
- WLAN 802.11 B/G
- Bluetooth 2.1 + EDR
- Käyttöjärjestelmä: Linux kernel 2.6.32
- 1 x GigaBit Ethernet
- 2 x USB 2.0
- Mitat: 95 x 65 x 48.5 mm



KUVA 1. Guruplug plus Linux palvelin (15)

### 3.2 Arduino Duemilanove-protoalusta

Arduino Duemilanove (kuva 2) on protoalusta, jossa on Atmega328-mikrokontrolleri. Siinä on 14 digitaalista IO-nastaa, joista kuutta voidaan käyttää PWM-ulostuloina, kuutta analogisina sisääntuloina. Se sisältää myös 16:n MHz:n kiteen, USB-liitännän, virtaliitännän, ICSP-liittimen ja reset-painikkeen. (5.)

Arduinon tuottamat protoalustat perustuvat avoimuuteen. Piirikaviot, tuotanto-ohjelmistot, käyttäjien tuottamat ja jakamat sovellukset ovat vapaasti ladattavissa osoitteesta <http://www.arduino.cc/>. Edut tämän käytössä on helppokäyttöisyys, edullisuus, monet sovellukset ja ohjeet.



*KUVA 2. Arduino Duemilanove -protoalusta*

### 3.3 Liiketunnistin (PIR)

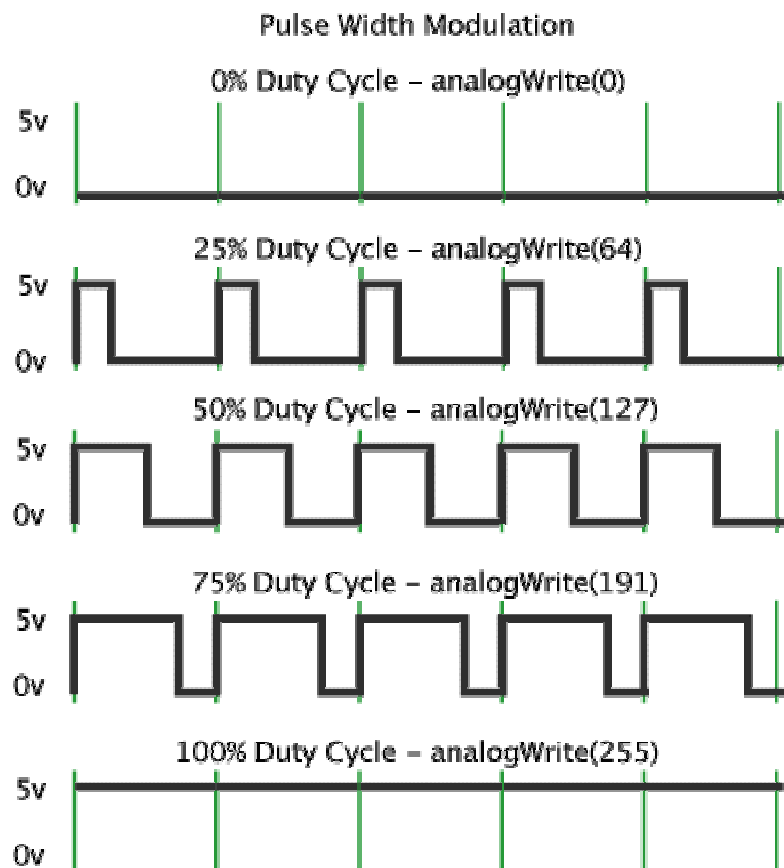
Passiivinen infrapunatunnistin (PIR) (kuva 3) on liiketunnistin, joka nimensä mukaisesti tunnistaa liikettä. (PIR) liiketunnistin toimii tekniikalla, joka tunnistaa infrapunasäteilyn voimakkuuden nopeat vaihtelut, jotka johtuvat kehon tai muun lämmönlähteen liikkeestä. Tämän perusteella saadaan muodostettua jännitesignaali, joka voidaan ohjata mm. mikrokontrollerille.



*KUVA 3. Liiketunnistin*

### **3.4 Pulssinleveysmodulointi**

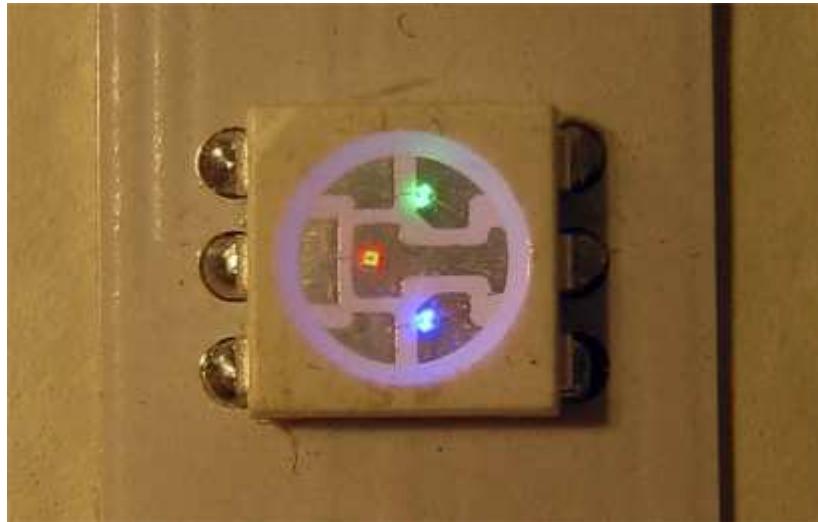
Pulssinleveysmodulointi (PWM) on modulointitapa, jossa kuormaan menevää jännitettä säädetään muuttamalla pulssisuhdetta. Digitaalisella ohjauksella, esim. mikrokontrollerilla tuotetaan kanttiaaltoa, jossa pulssin leveyttä säädetään asteittain asteikolla 0–100 % digitaalisesti portaistettuna (kuva 4.). Tällä tekniikalla muodostetaan jännitealue, joka tässä työssä oli 0–5 V. Atmega328-mikrokontrollerissa PWM-signaalin taajuus on noin 500 Hz, jolloin jakson pituus on 2 millisekuntia. Pulssin leveys on digitalisoitu 256 digitaaliseen arvoon (0–255). Arvo 0 muodostaa 0 V (kuva 4, 0 % Duty Cycle) ja vastaavasti arvo 255 5 V (kuva 4, 100 % Duty Cycle). Tämän tekniikan avulla saadaan mm. säädettyä LED-valaistuksen tehoa.



*KUVA 4. Pulssinleveysmodulointi ja pulssin leveys (Duty Cycle) (6)*

### **3.5 RGB-LED**

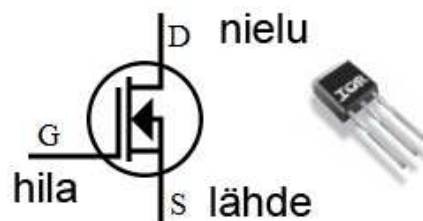
RGB-LED muodostuu kolmesta lähekkäin sijoitetusta LEDistä, joiden värit ovat punainen, vihreä ja sininen (kuva 5). Pulssinleveysmoduloinnin avulla LEDejä voidaan säätää tehoasteella 0–255, mikä tarkoittaa kokonaisuudessaan 256 eri valotehoa / väri. Värikombinaatioiden kokonaismääräksi muodostuu täten  $256 * 256 * 256 \approx 16,7$  miljoona eri värisävyä.



KUVA 5. RGB-LED-valo

### 3.6 N-tyyppin MOSFET

MOSFET on kanavatransistori. Transistorin kanavaa (nielun ja lähteen välistä) ohjataan hilan kautta. Kanava muuttuu johtavaksi, kun hilalle tuotu jännite ylittää kynnyksjännitteen. Tämän transistorin avulla voidaan esim. pienemmällä jännitteellä (5 V) ohjata suuremman jännitteen 12 V:n virtapiiriä.



KUVA 6. N-tyyppin MOSFET

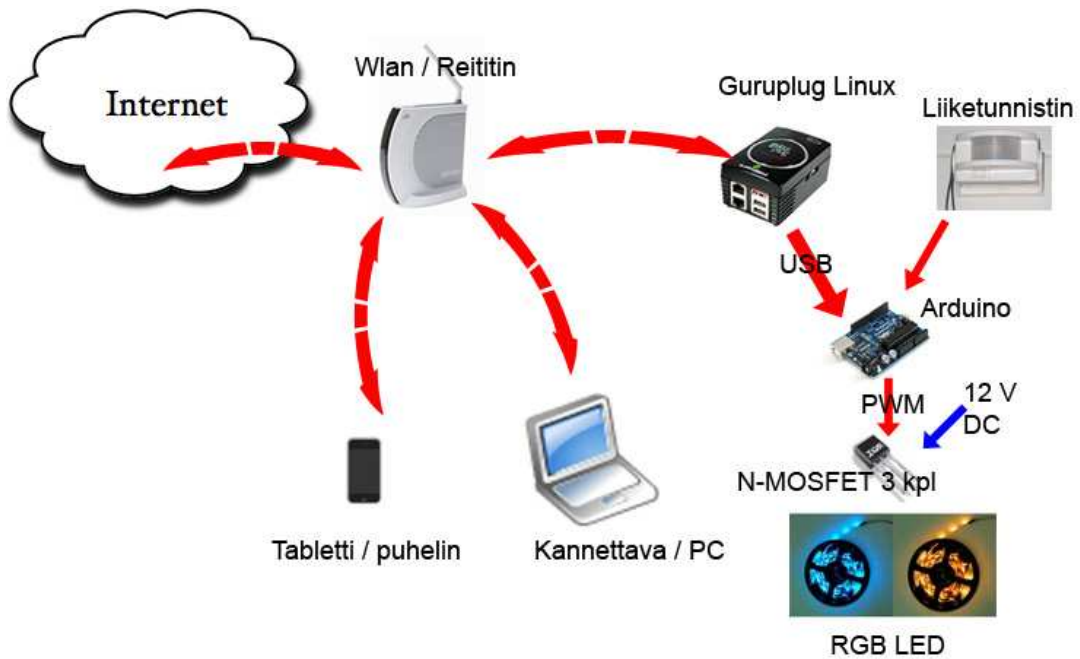
## 4 RGB-LED-VALAISTUKSEN TOTEUTUS

Tässä luvussa käydään läpi työn kokonaisrakenne, yhdistetään ja otetaan käyttöön edellisissä kappaleissa käydyt laitteet ja tekniikat. Tarkoituksena on valaista lukijalle järjestelmän kokonaisuus ja pilkkoa se ymmärrettäviin osiin. Piirikaavio ja lähdekoodit ovat liitteinä vapaasti käytettävissä GPL-lisenssin mukaisesti. Python-kielellä tehdyissä ohjelmissa muutoksia on tehty lähinnä laitteen nimeämisissä, kun taas mikrokontrollerin ohjelma on luotu kokonaan itse.

### 4.1 RGB-LED-valaistuksen ohjaus selaimen kautta

RGB-LED-valaistuksen ohjaus voidaan kytkeä internetin ja/tai intranetin kautta hallittavaksi (kuva 7). Väriohjaus tapahtuu internet-selaimen kautta (kuva 8), jossa on Java-tuki. Selaimen kautta hiiren avulla tai yksittäisten väriarvojen syöttämisellä arvoina 0–255 käyttöliittymän R, G, ja B syöttökenttiin saadaan valitut arvot välitettyä Guruplug Linux-palvelimelle, jossa on aktivoitu Apache2-web-palvelu. Guruplug ottaa selaimesta välitetyn tiedon Python-ohjelmointikielellä luotuun ohjelmaan (liite 3). Ohjelma lähettää väritiedon sarjamuodossa USB-kaapelin kautta Arduino-mikrokontrollerille, josta taas tieto välittyy ulos PWM-porteista edelleen N-tyypin MOSFETeille. Yhdessä käyttöjännitteen ja PWM:n avulla RGB-ledit muuttavat väriä käyttäjän haluamalla tavalla. Asetettu väri sammuu, mikäli liiketunnistin ei ole tunnistanut liikettä ajastetun ajan sisällä. Asetettu väri syttyy uudelleen liiketunnistimen havaittua liikettä. Uuden väritiedon syöttäminen järjestelmään käynnistää myös ledit. Mikäli uusi tieto on (0) jokaiselle värikanavalle, ledit sammuvat. Sattuminen, käynnistyminen ja muutokset tapahtuvat liukuvasti. Tavallisen valaisimen himmennin toimii samankaltaisesti, valaisuteho muuttuvat liukuvasti käyttäjän määrittämällä

tavalla. Tässä tapauksessa himmennyksen käsittelee mikrokontrolleri vakioidun ajan avulla, askel kerrallaan.



KUVA 7. RGB-LED valaistuksen ohjaus selaimen kautta rakenne



KUVA 8. RGB-LED-valaistuksen ohjaus selaimen kautta

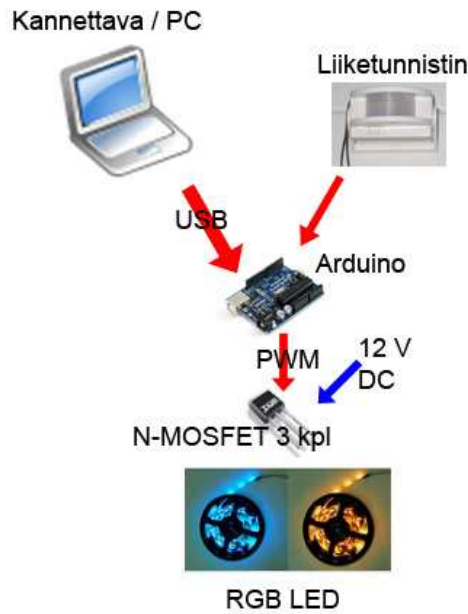


Laitteiston kokoonpano on seuraavanlainen:

- ohjaava laite (kannettava, työasema yms.), jossa internet-selain mahdollisuus
- Buffalo WHR-G54S wlan-tukiasema, joka sisältää palomuurin ja reitittimen
- Guruplug plus Linux-palvelin Debian Lenny-käyttöjärjestelmällä, python- serial- ja Apache2 -ohjelmilla
- USB-kaapeli
- Arduino Duemilanove-prototyyppi
- paristoilla toimiva liiketunnistin, josta langallinen kytkentä mikrokontrollerille
- 3 kpl N-kanavan MOSFETtia
- AC 230 V – DC 12 V muuntaja
- 5 metrinen RGB-led-sarja.

## **4.2 RGB-LED-valaistuksen ohjaus suoraan tietokoneelta**

RGB-LED-valaistuksen ohjaus voidaan kytkeä suoraan Linux-tietokoneelta hallittavaksi (kuva 9). Värin ohjaus tapahtuu ohjausikkunan kautta (kuva 10), joka on toteutettu Python-ohjelmointikielellä (liite 1). Hiiren avulla tai väriarvon numeerisella syöttämisellä ohjelmaan saadaan väriarvo välitettyä USB-kaapelin kautta Arduino-mikrokontrollerille, josta tieto välittyy ulos PWM-porteista edelleen N-tyypin MOSFETeille. Toimintaperiaate on lähes sama kuin edellisessä ohjausjärjestelmässä. Erona on, että valaistusta ohjataan suoraan tietokoneelta ja käyttöliittymä on erilainen.



KUVA 9. RGB-LED valaistuksen ohjaus suoraan tietokoneelta



KUVA 10. Väriasetuspaneeli

Rakennekokoonpano oli seuraavanlainen:

- ohjaava laite (kannettava, työasema yms.), jossa Python-ohjelman ajomahdollisuus
- USB-kaapeli
- Arduino-protoalusta

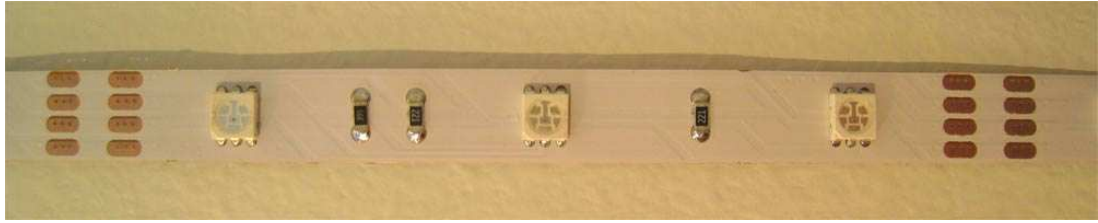
- paristoilla toimiva liiketunnistin, josta fyysinen kytkentä mikrokontrollerille
- 3 kpl N-kanavan MOSFETia
- AC 230 V – DC 12 V muuntaja
- 5 m RGB-led-sarja.

### **4.3 Mikrokontrollerin sovellus**

Ohjelman päätehtävä on vastaanottaa asetustiedot tietokoneelta ja välittää ne kullekin värikanavalle pulssileveysmoduloituna. Toisin sanoen RGB-LED-värinauhaan asettuu ohjauspaneelista säädetty väri. Mikrokontrollerin toisena tehtävänä on ottaa liiketunnistimen signaali vastaan ja käsitellä se. Mikäli havaintoalueella on liikettä, lähettää liiketunnistin + 5 V jännitepulssin, jolla nollataan ohjelman sisäinen laskuri. Valaistus sammuu kokonaan laskurin saavuttaessa laskuriin määritellyn ajan. Valaistuksen sammuminen vaatii laskurin saavuttavan määritetyn ajan ilman, että liikettä oltaisiin tunnistettu tai uutta väriasetustietoa olisi syötetty mikrokontrolleriin. Vastaavasti ohjelma sytyttää asetetun värin liukuvasti liiketunnistimen tunnistettua liikettä. Myös uuden väritiedon syöttäminen mikrokontrollerille käynnistää valaistuksen (mikrokontrollerin ohjelma, liite 2).

### **4.4 Kytkentäkaavio (liite 5)**

Käytössä oli 5 metrin pituinen RGB-LED-sarja. Käyttöjännite ledeille oli 12 V ja maksimaalinen tehon käyttö 30 W. Muuntaja antoi sopivan käyttöjännitteen. Alkuun on kytketty kytkin katkaisemaan käyttöjännitteen, minkä perässä on diodi (D10) suojaamassa laitteistoa negatiiviselta virralta. Työssä käytetty sarja oli katkottavissa kolmen RGB-LEDin sarjoihin (kuva 11). Kunkin sarjan pituus oli 10 cm. Koko viiden metrin nauha sisälsi yhteensä 150 kpl RGB-LEDejä.



*KUVA 11. RGB-LED-sarja*

Tämän kolmen RGB-LEDin sarja (SARJA 1) on kuvattu kytkentäkaaviossa diodista D1 diodiin D9, jossa D1–D3 edustavat kolmea sarjaan kytkettyä vihreää lediä, D4 – D6 kolmea sarjaan kytkettyä punaista lediä ja D7 – D9 kolmea sarjaan kytkettyä sinistä lediä. Virtaa rajoittavat vastukset R1 - R3 on sijoitettu kunkin värisarjan väliin.

RGB-LED-sarja päättyy kunkin värikanavan maa (GND) -yhteyteen. Tähän väliin on sijoitettu N-tyypin MOSFET:it Q1 – Q3. Mikrokontrollerin 3 PWM-lähtöä ohjaavat kunkin värisarjan R, G tai B maayhteyttä kunkin n-tyypin MOSFET:in hilan (piirroksessa Q1 G, Q2 G ja Q3 G) kautta.

Liiketunnistin on kuvattu piirikaaviossa kytkimenä. Liikettä tunnistettaessa + 5 V aktivoituu liiketunnistimen portin 1 kautta mikrokontrollerin tuloportille. Kun liikettä ei tunnisteta, on 0 V aktiivisena.

## **4.5 Tarkemmat asetustiedot**

Jotta ohjelmat toimisivat suoraan liitteenä olevilla lähdekoodeilla, tulee seuraavat asiat olla huomioituna.

### **4.5.1 Guruplug ja ohjaavan tietokoneen udev-asetukset**

USB-kaapelin liittäminen Linuxissa aiheuttaa muuttuvan laitteen nimeämisen. Ensimmäisellä kerralla Arduino-mikrokontrolleri voi olla liitettynä nimellä `/dev/ttyUSB0` kun taas seuraavan kerran liitettäessä laitteen nimi voi olla `/dev/ttyUSB1`. Tämä muodostaa ongelman ohjausjärjestelmän ohjelmien luomiseen. Tämä on kuitenkin korjattavissa

udev-asetuksilla. Laite nimettiin */dev/arduino*, jolloin ohjelma osaa ottaa oikeaan osoitteeseen oikeat yhteydet. Tarkemmat ohjeet liitteestä 4, kohta 1.

### 4.5.2 Python-serial-ohjelma

Sarjaliikennekeskusteluun tarvitaan python-serial-ohjelma, jonka saa Debian Linuxissa asennettua komennolla *sudo aptitude install python-serial*.

### 4.5.3 Web käyttöliittymän asentaminen

RGB-LED web-ohjauskäyttöliittymä on ladattu osoitteesta <http://www.eyecon.ro/colorpicker/>. Tiedot tallennetaan ladatut tiedostot - palvelimen web-kansioon, esim. */var/www/colorpicker*, ja muokataan *js*-kansiossa sijaitsevaa *layout.js* -tiedostoa liitteen 4 kohdan 2 mukaisesti.

### 4.5.4 Web käyttöliittymän sarjaliikenne-rajapinta

Jotta käskyt välittyisivät mikrokontrollerille, tulee *webserver.py*-ohjelma (liite 3) käynnistää komennolla *python webserver.py*.

### 4.5.5 Valon ohjaus suoraan tietokoneelta

Jotta käskyt välittyisivät mikrokontrollerille, tulee *VARI\_ASETUS.py*-ohjelma (liite 1) käynnistää komennolla *python VARI\_ASETUS.py*.

## 4.6 Lopputulos

Valaistus on näyttävämmän näköinen kun ledit eivät ole suoraan näkösällä vaan loistavat epäsuorasti esim. verholaudan takaa. 5 metrin RGB-LED-sarja asennettiin verholaudan taakse (kuva 12). Ohjauskeskus asennettiin verhojen taakse piiloon. Vain liiketunnistin jäi suoraan näkyville.



*KUVA 12. RGB-LED-sarja*

## 5 AVOIMEN LÄHDEKOODIN KODIN AUTOMAATIO-SOVELLUKSIA

Avoimen lähdekoodin kodin automaatio-sovelluksia on lukuisia. Tässä esitellään kolme kodin automaatio-sovellusta.

### 5.1 LinuxMCE

LinuxMCE (Linux Media Center Edition) on avoin ja ilmainen sovellusalausta, jossa on TV-ruutuihin suunniteltu käyttöliittymä. Ohjelmalla voidaan hyödyntää kodin teatterijärjestelmää, digitaalista videotallenninta (DVR) ja kodin automaatiolaitteita. (7)

LinuxMCE on avoimen lähdekoodin ohjelma, joka on rakennettu Kubuntu-käyttöjärjestelmän päälle. Kodin automaatio-ohjelmalla voidaan hallita tuettujen tietoliikennestandardien laitteita, joita on lukuisia. Ohjelmalla voidaan tehdä yksittäisiä painikkeita erinäisiä tilanteita varten (kuva 13), jossa ryhmäkäskeillä esimerkiksi asetetaan koti turvalliseen tilaan vaikkapa hälyttimien ja valaistuksen osalta. Halutut asetukset ovat käyttäjien muokattavissa tarpeidensa mukaisesti.



KUVA 13. LinuxMCE kodin automaation hallintakäyttöliittymä (8)

## 5.1.1 LinuxMCE Orbiter

Orbiter on yleisnimitys laitteesta, jolla hallitaan LinuxMCE:tä käyttäjätasolla. Orbiter voi olla kaukosäädin, tietokone, kännykkä tai vastaava laite, joka on yhteensopiva ohjelman kanssa (9). Internet-selain orbiter-käyttöliittymä mahdollistaa ohjelman käytön lukuisilla digitaalisilla laitteilla. Ohjaamiseen riittää laitteisto, jossa on internet-selain.



KUVA 14. LinuxMCE orbiter internet-käyttöliittymä (10)

Lista tuetuista orbiter-laitteista löytyy osoitteesta:

<http://wiki.linuxmce.org/index.php/Category:Orbiters> .

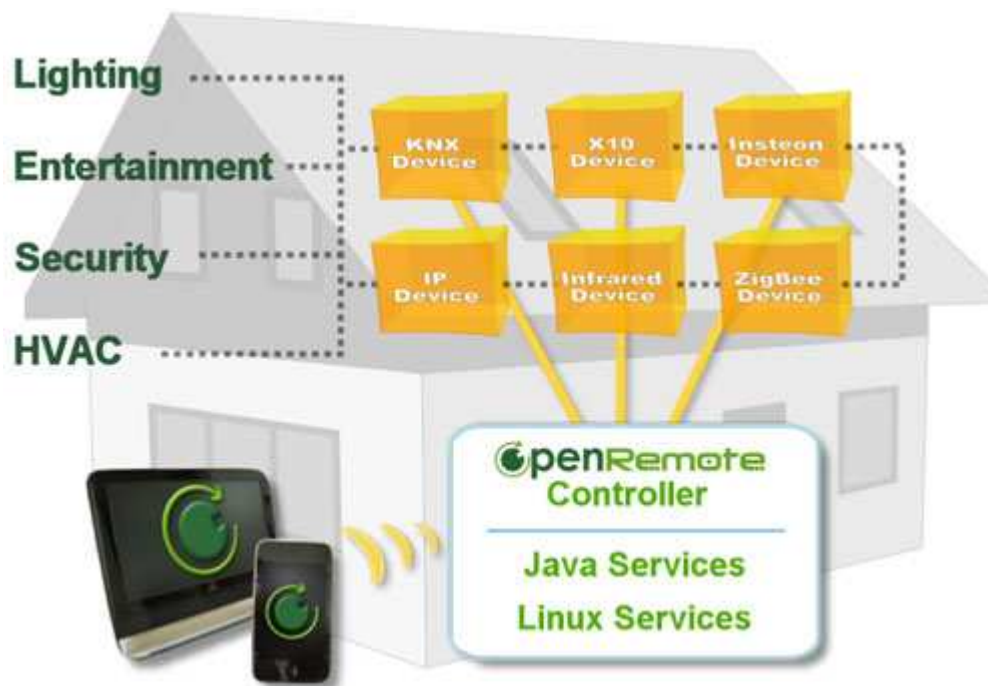
## 5.1.2 LinuxMCE-tuetut tietoliikennestandardit

LinuxMCE tukee KNX-, X10- ja Z-Wave-tietoliikennestandardeja.



## 5.2 OpenRemote

OpenRemote on avoin kodin automaatio-ohjelma, jonka voi asentaa Microsoft (XP ja 7), Mac OS X (Snow Leopard) ja Linux (Ubuntu 8 ja 10, sekä Voyage 0.5.x ja 0.6.x) käyttöjärjestelmien päälle. Ohjelma tukee yleisiä kodin automaatiolaitteita ja tietoliikennestandardeja. Ohjauspaneeli on suunniteltu käytettäväksi Apple- ja Android-kosketusnäyttöpuhelimia ja -tabletteja varten. Ohjauspaneelia voi ohjata myös internet-selaimen avulla.



KUVA 14. OpenRemote Designer (11)

### 5.2.1 OpenRemote Designer

OpenRemote Designer on internetissä toimiva ohjauspaneelien suunnitteluohjelma, jolla voi luoda erilaisia käyttäjäprofileja ja käyttöliittymiä käyttäjien preferenssien mukaan.



KUVA 15. OpenRemote Designer (12)

## 5.2.2 OpenRemote tuetut tietoliikennestandardit

OpenRemote tukee KNX- ja X10-tietoliikennestandardeja.

## 5.3 Misterhouse

Misterhouse on avoimen lähdekoodin kodin automaatio-ohjelma. Ohjelma on kirjoitettu Perl-ohjelmointikielellä, ja se toteuttaa käskyjä perustuen aikaan, verkkoon, pistorasiaan, ääneen ja sarjamuotoiseen tietoon. Ohjelma toimii Windows 95/98/NT/2k/XP:llä ja suurimmalla osalla Unixiin perustuvilla alustoilla sisältäen Linuxin ja Mac OSX:n. (13)



KUVA 14. MisterHouse internet-käyttöliittymä (14)

### 5.3.1 MisterHouse tuetut tietoliikennestandardit

MisterHouse tukee KNX-, X10- ja Z-Wave-tietoliikennestandardreja.

## 6 TULOKSET JA JOHTOPÄÄTÖKSET

### 6.1 Guruplug plus

Pienitehoinen logiikkakeskus on ideaali kodin älykkäänä ohjauskeskuksena. USB-väylään on mahdollista kytkeä erilaisia laitteita kamerasta releohjaukseen ja kosketusnäyttöpaneelisiin. USB-väylään saa kytkettyä myös X10-, Z-Wave- ja KNX-ohjauslaitteet. Vähäisen tehonkulutuksen (5 W) ansiosta laitteisto voi olla päällä 24 tuntia vuorokaudessa. Laitteiston hinta on verrattain edullinen. Keskuksen ohjaus voidaan toteuttaa myös internetin kautta hallittavaksi. Käyttökohteita on lukuisia, joten laitteistolla on mielestäni paljon potentiaalia.

Harmikseni joudun toteamaan, että juuri tämän laitteiston tämänhetkinen ongelma on ylikuumentuminen rakenteellisen suunnitteluvirheen takia. Uutta versiota ei toistaiseksi ole tullut vielä markkinoille. Pian sen jälkeen kun sain työn valmiiksi, Guruplug USB-portit lakkasivat toimimasta. Tämän totesin kun mittasin yleismittarilla jännitteet USB-porteista ja havaitsin, että ne olivat kadonneet. Asensin käyttöjärjestelmän uudestaan laitteeseen, mutta sekään ei auttanut. Odottelen laitteistosta uutta versiota.

#### 6.1.1 LinuxMCE mini ja Guruplug

Linkin <http://wiki.linuxmce.org/index.php/GuruPlug> mukaan LinuxMCE mini ja Guruplug saadaan sovitettua yhteen. Tämä versio on tarkoitettu vain kodin automaatiolaitteiden hallintaan ja siihen on tällä hetkellä vain Z-Wave tuki.

#### 6.1.2 OpenRemote ja Guruplug

Linkin <http://openremote.org/pages/viewpage.action?pageId=15532367> mukaan Guruplug ja OpenRemote saadaan sovitettua yhteen.

### 6.1.3 MisterHouse ja Guruplug

Linkin <http://misterhouse.wikispaces.com/MisterPlug> mukaan Guruplug ja MisterHouse saadaan sovitettua yhteen.

## 6.2 RGB-LED

Alun perin oli tarkoitus selvittää sopiiko RGB-LED tavallisen valaistuksen korvaajaksi. Ostamani sarja ei sellaisenaan siihen sovi. Valoteho ei ole tarpeeksi suuri ja värit eivät ole mielestäni tarpeeksi tasaiset. Kalliimpaa ja tasalaatuisempaa sarjaa en testannut, joten voi olla, että sellaisilla se olisi mahdollista. Sarja sopii kuitenkin erittäin hyvin automaattiseen yövalaisuun ja luomaan tunnelmaa sisustuksellisenä elementtinä. Yli 16 miljoonan värisävyt tekevät RGB-LED-sarjasta mukavan kodin tunnelmavalaisimen, jota voi ylpeänä esitellä vieraille. Valaisimella on paljon potentiaalisia asennuskohteita kodeissa ja julkisissa tiloissa.

## 6.3 Arduino-protokallusta

Arduino on loistava alusta aloittaa mikrokontrollereiden käyttö. Ohjelmia ja ohjeita on lukuisia ja itse ohjelmointikieli on selkeää. Arduino-laitteet ovat suhteellisen edullisia ja niihin on saatavilla erilaisia lisämoduuleja. En löytänyt Z-Wave-, KNX- ja X10- RGB-LED -ohjausjärjestelmää, joten päätin rakentaa sen itse. RGB-LED-valaistuksen ohjausjärjestelmä on helppo integroida HTML-käyttöliittymiä käyttäviin sovelluksiin yksinkertaisesti linkittämällä.

## 6.4 Tietoliikennestandardit

Tarkoituksena oli etsiä mahdollisimman kustannustehokkaita kodin automaatio-ohjauslaitteita ja niiden tukemia tietoliikennestandardeja toimiakseen mahdollisimman hyvin valitsimieni Linux kodin automaatio-ohjelmien kanssa. Zigbee-standardi jäi listalta pois koska siinä eri

laitevalmistajien laitteet eivät välttämättä toimi keskenään sekä Linux kodin automaatio-ohjelmien tuki oli huono.

### 6.4.1 X10

Ajatus siitä, että viestit kulkevat jo olemassa olevaa sähköverkkoa pitkin, on mielestäni järkevä. Tällä vältytään uusien kaapelien asentamiselta kohteissa, joissa rakennukseen ei ole vedetty kodin elektroniikkaa ohjaavia kaapeleita. Ensimmäiseksi ongelmaksi muodostuu kuitenkin 3-vaihejärjestelmä. Jos ohjausviestejä haluaa lähettää näiden kesken, on hankittava suhteellisen hinnakas laite. X10-viestit ovat alltiita häiriöille, ja tietoturva on myös kyseenalainen. Naapurustosta voidaan pahimmassa tapauksessa ohjata toisten asuntoja tahattomasti tai tahallisesti. Myös langaton standardi on tietoturvaltaan heikko. Tämän takia jättäisin itse valitsematta X10:tä kotiani ohjaamaan.

Vahvuudet:

- Ei vaadi suuria investointeja ja järjestelmää voi asteittain laajentaa
- Laitteiden tuottajia on suhteellisen paljon
- Laaja skaala eri tuotteita
- Suhteellisen edullisia laitteita
- Suhteellisen helpot asentaa ja käyttää
- Monet ohjelmat tukevat standardia

Heikkoudet:

- Viestit voivat kulkea naapurustoon ja toisinpäin ilman tätä estävää erillistä laitetta
- Vain yksi käsky kerrallaan
- Epävarmuus viestin kulusta perille, voi vaatia käskyn antamista useampaan kertaan
- Hidas, yhden käskyn toteuttaminen vie n.  $\frac{3}{4}$  sekuntia
- Viestit eivät kulje eri vaiheiden välillä, tähän pitää olla erillinen suhteellisen hinnakas laite
- Langattomassa lähetyksessä huono tietoturva, ei kryptausta

- Pienitehoiset laitteet (alle 50 W) eivät välttämättä toimi kaikkien X10-laitteiden kanssa
- Korkeatehoiset laitteet (liesi, imuri jne.) ollessa päällä heikentävät tai estävät viestien perillemeno

### 6.4.2 Z-Wave

Langattomista kodin automaatio-protokollista Z-Wave näyttää lupaavimmalta. Tietoliikenteen suojausmahdollisuus 128-bit AES-kryptauksella, matalaenerginen toiminta ja keskenään eri valmistajien tuotteiden yhteensopivuus tekee protokollasta mielestäni vahvan tekijän. Paristoilla toimivista laitteista voi olla montaa mieltä. Esim. jo paristoilla toimivan palohälyttimen vaihtaminen paristoilla toimivaan Z-Wave-palohälyttimeen tuo todennäköisesti lisäarvoa. Toisaalta mikäli paristoilla toimivia laitteita on asennettu paljon, voi se jo muodostaa lisää huolenpitoa ja muistamista, mikä taas voi laskea käyttömukavuutta. Toimilaitteiden tulisi käyttää mahdollisimman paljon muita virtalähteitä kuin paristoja mikäli mahdollista. AC-kytketyillä Z-Wave-laitteilla yhdistettynä järkevillä paristoilla toimivilla laitteilla, kuten palo- ja tulvahälyttimellä, on hyvä aloittaa kodin asteittainen automatisointi.

Vahvuudet:

- Ei vaadi suuria investointeja ja järjestelmää voi asteittain laajentaa
- Verkon ulottuvuus saadaan laajennettua välittäjäsilmoitusten avulla
- Paristoilla toimivat laitteet voidaan sijoittaa hankalisiin paikkoihin
- Nopea
- Ryhmäkäsyt mahdollisia
- Matalaenerginen
- Mahdollisuus tietoliikenteen kryptaukseen
- Eri valmistajien laitteet toimivat keskenään

Heikkoudet:

- Paristoilla toimivien laitteiden paristoja pitää vaihtaa ajoittain
- Tuotetarjonta ja skaala suppeampaa kuin X10-standardin laitteilla

### 6.4.3 KNX

KNX tarjoaa laajan skaalan eri viestintätapoja. Kirkkaimmaksi näistä nousee kierretty parikaapeli (TP) luotettavuutensa ja tietoturvasa ansiosta. Tämä valinta näkyy tietenkin loppukustannuksissa. Kaapelointi tulee maksamaan.

KNX (PL) sähköverkon lähetyksistä oli lähes mahdotonta löytää käyttökokemuksia internetin kautta. Olettaisin kuitenkin että tässä standardissa on samoja ongelmaelementtejä kuin X10:ssä, mutta ei ehkä vastaavassa mittakaavassa.

Langattoman standardin (RF) kryptauksen puute ihmetyttää. Joten sen käyttöä en suosittelisi.

Vahvuudet:

- Järjestelmää voi asteittain laajentaa
- Monta vaihtoehtoa tiedon siirrossa (TP, PL, RF, IP ja IR)
- TP:ssä hyvä tietoturva johtuen väyläkaapelista
- TP:ssä luotettava tiedonsiirto
- Nopea
- Ryhmäkäskyt mahdollisia
- Laaja valikoima tuotteita
- Monta tapaa ottaa laitteet käyttöön (A, E ja S –mode)
- Eri valmistajien laitteet toimivat keskenään

Heikkoudet:

- Langattomassa tiedonsiirrossa ei kryptausta
- TP vaatii omat kaapelit joka nostaa tiedonsiirron hintaa huomattavasti

### 6.4.4 Yhteenveto tutkituista tietoliikennestandardeista

Langattomaksi (RF) järjestelmäksi valitsisin Z-Waven koska tämä on pelkästään langaton standardi, nopea, turvallinen ja käyttää tiedon siirtoon



verrattain vähän energiaa. X10 ja KNX langattomien tietoturvan puute langattomissa järjestelmissä rajaa nämä pois.

Langalliseksi standardiksi kierretty parikaapeli (TP) toimisi todennäköisesti luotettavimmin. Korkeammat kustannukset saattavat rajata tämän standardin käyttöä.

Sähköverkon kautta lähetettävät viestit ovat alttiita häiriöille ja tietovuodoille. X10 on ollut pitkään markkinoilla ja siitä löytyy paljon käyttäjien tietoa verkosta kun taas KNX (PL) sähköverkon viestittimien käytöstä on varsin hankala löytää käyttäjäkokemuksia. Itse valitsisin kuitenkin KNX (PL) X10 sijaan mikäli toimilaitteiden hinnoissa ei ole merkittäviä kustannuseroja.

Todennäköisesti aloitan Z-Wave laitteilla koska Z-Wave on yhteensopiva LinuxMCE mini kodin automaatio-ohjelman kanssa. LinuxMCE minin saa asennettua Guruplugiin.

## **6.5 Linux- kodin automaatio-sovellukset**

Valittaessa järjestelmää kannattaa ensin tutustua sovellukseen, jota aikoo käyttää. Varsinkin tuetut laitteet kannattaa tarkistaa ennen niiden hankintaa. Ei tuetut laitteet eivät välttämättä toimi ja tämä voi tuoda pettymyksen.

Perehdyin sovelluksiin internetin kautta. Suoria käyttökokemuksia minulla ei ole koska en asentanut ohjelmia. LinuxMCE, Z-Wave ja uudistunut Guruplug voi myöhemmin mennä minulla henkilökohtaiseen testaukseen.

## 7 POHDINTA

Ihmiset suhtautuvat varauksellisesti kodin automatisointiin. Pelko hallitsemattomista tietoteknisistä tilanteista pitää kuluttajat varpaillaan. Mitä jos valokytkimen mekaaninen kosketus ei enää tottele? Miten käy kun viiden vuoden päästä jokin ohjauslaite menee rikki? Onko varalle laitteita ja voiko niitä itse vaihtaa? Huoltojen kustannukset saattavat kohota korkeaksi. Laitteet saattavat olla epävarmoja tai ohjelmat voivat virhetilanteissa toimia käyttäjän mielen vastaisesti. Nämä huolet ovat ymmärrettäviä ja aiheellisia; tekniikka voi pettää.

Kodin automatisoinnin voi aloittaa varovaisesti automaattisilla tunnelmavalaisimilla ja mahdollisesti liittämällä Z-Wave- palovaroittimen, - murtohälyttimen ja -etäohjauksen auton lämmitykselle. Ei kriittisille, mutta elämää helpottaville ja sisustuksellisesti näyttäville ratkaisuille todennäköisesti löytyy kysyntää. Tätä voisi kutsua puoliautomaattiseksi lähestymistavaksi.

Kaikki välineet kodin automatisoinnille on olemassa. Osaavalta henkilöltä kodin automatisoinnin toteuttaminen onnistunee verrattain kustannustehokkaasti, kun järjestelmässä käytetään hyväksi avoimien sovelluksien ohjausjärjestelmiä. Avoimet ohjelmat tulevat kuitenkin ilman mitään takuita. Kriittinen, jälkikäteen havaittu ongelma voi langeta täysin käyttäjän tai laitteiden asentajan (kolmannen osapuolen) ratkaistavaksi. Ohjelmavirheiden korjauksille voi muodostua pitkiä viiveitä, tai ne voivat jäädä kokonaan korjaamatta, mikäli palvelun tarjoaja ei näitä osaa käsitellä. Toiminnan ja turvallisuuden varmuus on kyseenalaista, jos esimerkiksi yritys joka ei tunne ohjelmistoja ja niiden syvällisiä toimintaperiaatteita tuottaa kodin automaatiopalveluita. Tällaisessa tapauksessa avoimien järjestelmien tarjoaja joutuisi tukeutumaan avoimien sovelluksien ryhmien vapaaehtoisein toimenpiteisiin. Pienellä budjetilla ja huonolla sovellusosaamisella palveluita

ei välttämättä kannata lähteä tarjoamaan varsinkaan kriittisiin kohteisiin kuten kodin ovien lukitusjärjestelmiin.

Seuraavana kehitysaskeleena RGB-LED-valaistukseen valitsisin Arduino-mikrokontrollerin ja siihen liitettävän Zigbee-lähettimen ja -vastaanottimen, jolloin ohjattavia RGB-LED-sarjoja voisi olla lukuisia eri paikoissa. Järjestelmä koostuisi vastaanottavassa päässä RGB-LEDeistä, muuntajasta, Arduino-Zigbee-moduulista ja Arduino-mikrokontrollerista. Tämä lisäisi muunteluvaihtoehtoja ja joustavuutta RGB-LED-valaistuksien ohjaukseen.

Aloitin työn lähes olemattomalla Linuxin, ohjelmoinnin ja kodin automaatio-sovelluksien ja -standardien tuntemuksella. Otin haasteen vastaan, koska tiedän, että Internetistä löytyy lähes kaikkeen tieto ja osaaminen. Työssä hyppäsin lähes tuntemattomaan kokemattomana opiskelijana. Tämä näkyi kulutetussa ajassa. Linuxin komentokehotteen tutustumiseen, Arduino- ja Python-ohjelmointikielen oppimiseen kului paljon aikaa. Myös ohjelmien ja standardien tutkiminen vei oman aikansa. Olen kuitenkin erittäin tyytyväinen kun pääsin tutustumaan tarkemmin Linuxin maailmaan. Kokonaisuudessaan työ oli haastava ja mielenkiintoinen.

## LÄHTEET

1. Philips Electronics. 2011. Kestävyys ja LED sekä LED ratkaisujen edut. Hakupäivä 9.5.2011  
<http://www.lighting.philips.fi/lightcommunity/trends/led/index.wpd>
2. X10 (industry standard). 2011. Hakupäivä 8.3.2011  
[http://en.wikipedia.org/wiki/X10\\_\(industry\\_standard\)](http://en.wikipedia.org/wiki/X10_(industry_standard))
3. Z-Wave FAQ. 2011. Hakupäivä 3.3.2011 <http://www.z-wave.com/modules/xoopsfaq/>
4. KNX, lisää standardista. 2011. Hakupäivä 12.3.2011 <http://www.knx.fi/>
5. Robomaa.com. 2011. Tuoteseloste. Hakupäivä 15.3.2011  
[http://www.robomaa.com/index.php?main\\_page=product\\_info&cPath=15\\_16&products\\_id=46](http://www.robomaa.com/index.php?main_page=product_info&cPath=15_16&products_id=46)
6. Arduino PWM. 2011. Hakupäivä 3.3.2011  
<http://www.arduino.cc/en/Tutorial/PWM>
7. LinuxMCE ohjelma. 2011. Hakupäivä 20.3.2011  
<http://en.wikipedia.org/wiki/LinuxMCE>
8. LinuxMCE. 2011. Hakupäivä 5.3.2011  
<http://linuxmce.com/index.php/gallery/screenshots#>
9. LinuxMCE orbiter. 2011. Hakupäivä 8.3.2011  
<http://wiki.linuxmce.org/index.php/Orbiter>
10. LinuxMCE orbiter. 2011. Hakupäivä 8.3.2011  
[http://wiki.linuxmce.org/index.php/Add\\_Web\\_Orbiter](http://wiki.linuxmce.org/index.php/Add_Web_Orbiter)

11. OpenRemote. 2011. Hakupäivä 14.3.2011  
<http://www.openremote.org/pages/viewpage.action?pagelId=2981955>
12. Lindfors J. 2010. OpenRemote Designer. Hakupäivä 14.3.2011  
<http://www.openremote.org/display/~juha/2010/07/04/OpenRemote+with+Apple+iPad>
13. MisterHouse kuvaus. 2011. Hakupäivä 16.3.2011  
<http://misterhouse.sourceforge.net/>
14. MisterHouse käyttöliittymä. 2011. Hakupäivä 16.3.2011  
<http://misterhouse.wikispaces.com/Tom>
15. Guruplug 2011. Hakupäivä 2.3.2011  
<http://www.globalscaletechnologies.com/t-guruplugdetails.aspx>

## LIITELUETTELO

- LIITE 1 VARI\_ASETUS.py lähdekoodi
- LIITE 2 ARDUINO\_RGB\_LED.pde lähdekoodi
- LIITE 3 webserver.py lähdekoodi
- LIITE 4 Asetustiedot
- LIITE 5 Kytkäkaavio

**VARI\_ASETUS.py lähdekoodi**

```
import gtk, sys, serial

#Yhteys mikrokontrolleriin, huom muist udev asetukset
ser = serial.Serial()
ser.setPort("/dev/arduino")
ser.baudrate = 57600
ser.open()
#GTK asetukset
w = gtk.Window()
c = gtk.ColorSelection()
w.add(c)
w.show_all()

#Variasetuksien syotto mikrokontrollerille
def callback(*args):
    color=c.get_current_color()
    ser.write("r" + chr(int(color.red/257)))
    ser.write("g" + chr(int(color.green/257)))
    ser.write("b" + chr(int(color.blue/257)))

c.connect('color-changed', callback)
w.connect('destroy', sys.exit)

gtk.main()

# Apuja on haettu alla olevasta linkista, yhteysasetukset joutui luomaan uudestaan
# http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1267104041
# Ohjeet: luo tiedosto VARI_ASETUS.py ja kopioi tämä koodi tiedostoon
# Liitä mikrokontrolleri USB-väylään
# Käynnistä tiedosto komennolla: python VARI_ASETUS.py
```

**ARDUINO\_RGB\_LED.pde lähdekoodi**

```
/*GPL Lisensoitu: Klaus Nurkkala*/
/*Muokkaa vain naita arvoja. ALKU-----*/

unsigned long viiveT = 15000;

/*Arvo millisekunneissa, 1 sekunti = arvo 1000. Koodin rakenne luotu niin etta viive jaetaan kahdella.
Esim: jos haluat 30 sekunnin viiveen sammutuksen alkamiselle, on laskutoimitus seuraavanlainen; 30000 / 2 = 15000, joten
muuttujalle viiveT asetetaan arvo 15000 */

int viive = 18; //Viive-aika himmennys tapahtumalle (isompi luku = pidempi viive)

/*Aseta PWM lahtoportit*/
int vihreaPin = 9;
int punainenPin = 10;
int sininenPin = 11;

/*Aseta liiketunnistimen tuloportti*/
int liikeT = 2;

/*Muokkaa vain naita arvoja. LOPPU-----*/

/*Asetetaan muuttujat laskurille*/
unsigned long aloitus_aika;
unsigned long nyk_aika;
unsigned long mennyt_aika;

/*Asetetaan muuttujat vareille*/
int RGBvari[3];
int punainen = 0;
int vihrea = 0;
int sininen = 0;

/*Asetetaan muuttujat varisummille*/
int variSumma;
int variSummaAlkup;

/*Esitellaan aliohjelmat*/
void haeVari();
void himmennaA();
void valaiseY();

/*Konfiguroidaan IO portit*/
void setup() {
  Serial.begin(57600);
  pinMode(punainenPin, OUTPUT);
```



```
pinMode(vihreaPin, OUTPUT);
pinMode(sininenPin, OUTPUT);
pinMode(liikeT, INPUT);
}

/*Asetetaan muuttujat laskurille*/
void loop() {

    //Haetaan varitieto sarjamuodossa
    haeVari();

    //Tallennetaan RGB varitieto toisiin muuttujiin myohempaan vertailuun
    punainen = RGBvari[0];
    vihrea = RGBvari[1];
    sininen = RGBvari[2];

    //Kirjoitetaan PWM tieto porteista ulos
    analogWrite(punainenPin, punainen);
    analogWrite(vihreaPin, vihrea);
    analogWrite(sininenPin, sininen);

    //Jos liiketunnistimen jannite 0V, aloitetaan laskuri
    if (digitalRead(liikeT) == LOW) {
        aloitus_aika = millis();
        do {
            nyk_aika = millis();
            mennyt_aika = nyk_aika - aloitus_aika;

            //Verrataan alkuperaisen varien summaa mahdolliseen uuteen syotettyyn varien summaan
            haeVari();
            variSumma = punainen + sininen + vihrea;
            variSummaAlkup = RGBvari[0] + RGBvari[1] + RGBvari[2];
            //Jos varitieto on muuttunut, lopetetaan while loop
            if (variSumma != variSummaAlkup) break;
        }

        //Jos ei varimuutoksia ja ei liiketunnistusta (0V), mennaan seuraavaan askeleeseen
        while(mennyt_aika < viiveT && digitalRead(liikeT) == LOW);

        //Jos liiketta ei tunnistettu vielakaan ja ei varimuutoksia, himmennetaan valot ja jos tunnistetaan liiketta tai varimuutos,
        hypataan seuraavaan askeleeseen
        while (digitalRead(liikeT) == LOW && variSumma == variSummaAlkup){
            haeVari();
            variSummaAlkup = RGBvari[0] + RGBvari[1] + RGBvari[2];
            himmennaA();
        }
    }
}
```

```

    }
}

//Jos liiketta tunnistettu tai varisumma poikkeaa, aloitetaan laskuri jonka mukana valaistus asetetaan portaattomasti
edelliselle asetustasolle
if (digitalRead(liikeT) == HIGH || variSumma != variSummaAlkup){
    aloitus_aika = millis();

    do {
        nyk_aika = millis();
        mennyt_aika = nyk_aika - aloitus_aika;
        valaiseY();
    }
    while(mennyt_aika < viiveT);
}

/*Aliohjelma valaistuksen portaattomalle kaynnistamiselle*/
void valaiseY(){

    haeVari();
    if (punainen < RGBvari[0]) {punainen ++;}
    if (vihrea < RGBvari[1]) {vihrea ++;}
    if (sininen < RGBvari[2]) {sininen ++;}
    if (punainen > RGBvari[0]) {punainen --;}
    if (vihrea > RGBvari[1]) {vihrea --;}
    if (sininen > RGBvari[2]) {sininen --;}
    analogWrite(punainenPin, punainen);
    analogWrite(vihreaPin, vihrea);
    analogWrite(sininenPin, sininen);
    delay(viive);
}

/*Aliohjelma valaistuksen portaattomalle sammutukselle */
void himmennA(){

    while (punainen > vihrea || sininen > vihrea){
        if (punainen > vihrea) {punainen --;}
        if (sininen > vihrea) {sininen --;}
        analogWrite(punainenPin, punainen);
        analogWrite(vihreaPin, vihrea);
        analogWrite(sininenPin, sininen);
        delay(viive);
    }
}

```

```
while (sininen > punainen || vihrea > punainen){
  if (sininen > punainen) {sininen --;}
  if (vihrea > punainen) {vihrea --;}
  analogWrite(punainenPin, punainen);
  analogWrite(vihreaPin, vihrea);
  analogWrite(sininenPin, sininen);
  delay(viive);
}

while (punainen > sininen || vihrea > sininen){
  if (punainen > sininen) {punainen --;}
  if (vihrea > sininen) {vihrea --;}
  analogWrite(punainenPin, punainen);
  analogWrite(vihreaPin, vihrea);
  analogWrite(sininenPin, sininen);
  delay(viive);
}

if (punainen > 0) {punainen --;}
if (vihrea > 0) {vihrea --;}
if (sininen > 0) {sininen --;}
analogWrite(punainenPin, punainen);
analogWrite(vihreaPin, vihrea);
analogWrite(sininenPin, sininen);
delay(viive);
}

/*Aliohjelma valaistustiedon hakemiselle tietokoneesta sarjamuodossa*/
void haeVari(){
  if(Serial.available() >= 2){

    switch( byte( Serial.read() ) ) {
      case 'r':
        RGBvari[0] = Serial.read();
        break;
      case 'g':
        RGBvari[1] = Serial.read();
        break;
      case 'b':
        RGBvari[2] = Serial.read();
        break;
    }
  }
}
```

**webservice.py lähdekoodi**

```

# GPL Licensed by Lari Hotari
import string, cgi, time, urlparse, sys, serial
from os import curdir, sep
from BaseHTTPServer import BaseHTTPRequestHandler, HTTPServer
from SimpleHTTPServer import SimpleHTTPRequestHandler

class ArduinoRgbSetter:

    def __init__(self):
        #Aukaistaan yhteys, huom laitenimi, muista udev asetukset
        self.ser = serial.Serial('/dev/arduino', 57600)
        self.ser.open()
        print "OPEN!"

    def passRgb(self, r, g, b):
        if (self.ser.isOpen()):
            #Lahetetaan rgb tieto sarjamuodossa kontrollerille
            print "Setting RGB", r, g, b
            self.ser.write("r" + chr(int(r)))
            self.ser.write("g" + chr(int(g)))
            self.ser.write("b" + chr(int(b)))
        else:
            print "serial isn't open."

class MyHandler(SimpleHTTPRequestHandler):
    def do_OPTIONS(self):
        #Haetaan url osoitteen kautta rgb tiedot
        o = urlparse.urlparse(self.path)
        if o.path == '/control':
            params = cgi.parse_qs(o.query)
            print "r", params.get('r')
            print "g", params.get('g')
            print "b", params.get('g')
            self.server.arduino.passRgb(params.get('r')[0],
            params.get('g')[0], params.get('b')[0])
            self.send_response(200)
        else:
            SimpleHTTPRequestHandler.do_OPTIONS(self)

class MyServer(HTTPServer):
    def __init__(self):
        HTTPServer.__init__(self, ('', 9080), MyHandler)
        self.arduino = ArduinoRgbSetter()

```

## LIITE 3/2

```
def main():  
    try:  
        server = MyServer()  
        print 'started httpserver...'  
        server.serve_forever()  
    except KeyboardInterrupt:  
        print '^C received, shutting down server'  
        server.socket.close()  
  
if __name__ == '__main__':  
    main()
```

## Asetustiedot

1) Luo tiedosto **99-usb-serial.rules** kansioon **/etc/udev/** ja lisää alla olevan tyyppinen asetus. **HUOM.** sinun yksittäiset asetukset poikkeavat todennäköisesti tästä joten joudut selvittämään omat asetustiedot:

```
#  
SUBSYSTEM=="tty", ATTRS{product}=="FT232R USB UART", ATTRS{idVendor}=="0403",  
ATTRS{idProduct}=="6001", ATTRS{serial}=="A700e0Lb", NAME="arduino"
```

```
#
```

Ohjeen lähde: <http://hintshop.ludvig.co.nz/show/persistent-names-usb-serial-devices/>

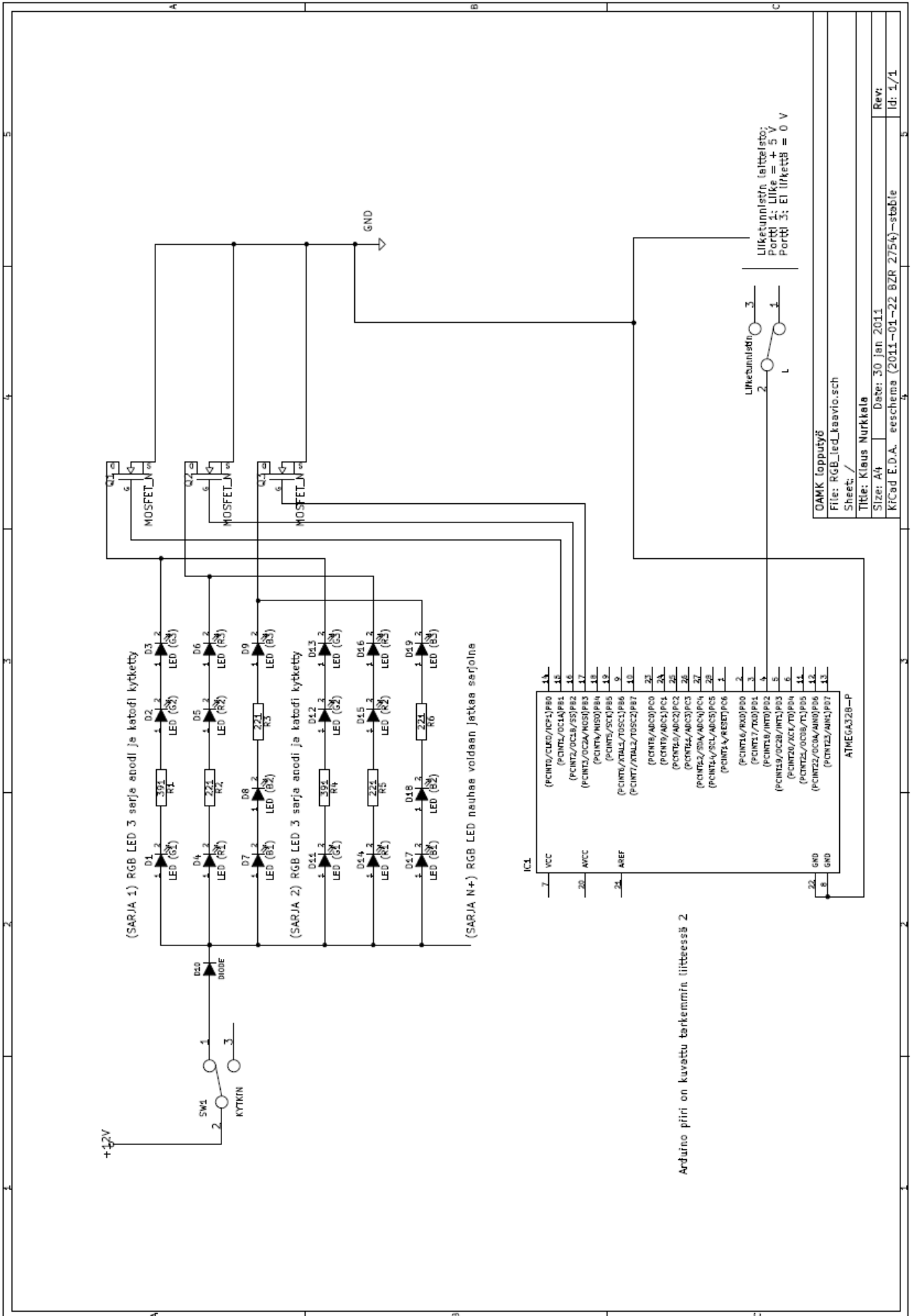
2) Muokkaa **layout.js** tiedoston **#colorpickerHolder** alla olevan mukaiseksi. **HUOM.** Aseta **IP** osoite palvelimesi osoitteeksi.

```
//  
$('#colorpickerHolder').ColorPicker({  
    flat: true,  
    onChange: function (hsb, hex, rgb) {  
        $.get("http://192.168.11.14:9080/control",{r: (rgb.r) ,g: (rgb.g),b: (rgb.b)});  
    }  
});  
//
```

Ohjelman lähde: <http://www.eyecon.ro/colorpicker/>

# KYTKENTÄKAAVIO

# LIITE 5



GAMK lopputyö	
File: RGB_led_kaavio.sch	
Sheet: /	
Title: Klausurikela	Date: 30 Jan 2011
Size: 44	Rev: 1/1
PCad E.D.A. eeschema (2011-01-22 BZR 275k)-stable	