

Pasi Väisänen

VÄRISÄVYVERTAILIJA

Insinööriö

Kajaanin ammattikorkeakoulu

Tekniikan ja liikenteen ala

Tietotekniikan koulutusohjelma

Syksy 2005



Osasto	Koulutusohjelma
Tekniikka	Tietotekniikka
Tekijä(t)	
Pasi Väisänen	
Työn nimi	
Värisävyvertailija	
Vaihtoehtoiset ammattiopinnot	Ohjaaja(t)
Konenäkö	Jukka Heino
Aika	Sivumäärä
Syyskuu 2005	24+12
Tiivistelmä	
<p>Insinööriyön tavoitteena oli suunnitella värisävyvertailija. Vertailijan tuli tunnistaa eri värisävyt toisistaan ja ilmaista niiden ero 7-segmentnäytöllä värin indeksinä. Laitteiston tuli olla halpa, helppokäyttöinen, ja sen tuli olla mahdollisimman tiiviissä paketissa. Yksi oleellinen tekijä oli, että sen tuli toimia 9 voltin nappiparistolla.</p> <p>Laitteisto koostuu kolmesta eri moduulista eli lähetin-vastaanotinyksiköstä, näytöstä ja prosessorista. Lähetin-vastaanotinyksikkö koostuu LEDeistä (RGB) ja valotransistoreista. Näyttönä toimii kolminumeroinen 7-segmentnäyttö. Värisävyjen tunnistus tapahtuu mittaamalla valotransistorin yli olevaa jännitettä, joka muuttuu mitattavan kohteen värisävyyn mukaan, kun sitä valaistaan erivärisillä LEDeillä. Prosessori muuntaa jännitteen A/D-muuntimessa digitaaliseen muotoon, jonka jälkeen se suorittaa mitatulle jännitteelle erinäisiä laskutoimituksia. Tämän jälkeen värin indeksiarvo tulostetaan 7-segmentnäytölle. Tuloksena on laite, jonka ns. ”rautapuoli” on valmis.</p>	
Luottamuksellinen	
Kyllä	
Ei X	
Hakusanat	
Konenäkö, värisävy, valotransistori	
Säilytyspaikka	
Kajaanin ammattikorkeakoulun kirjasto	



**Kajaanin
ammattikorkeakoulu**

Kajaani Polytechnic

**ABSTRACT
THESIS**

Faculty	Degree programme
Faculty of Engineering	Information Technology
Author(s)	
Pasi Väisänen	
Title	
A Colour Hue Comparisor	
Optional professional studies	Instructor(s) / Supervisor(s)
Machine Vision	Jukka Heino
Date	Total number of pages
Autumn 2005	24+12
Abstract	
<p>The goal of this Bachelor's thesis was to design a colour hue comparisor. The purpose of the comparisor was to identify different colour shades and to express the difference in a seven segment display as an index of a colour. The equipment was supposed to be cheap, easy to use and it was supposed to be in a most compact packet. One essential factor was that it would have to work with a nine volt battery.</p> <p>The equipment consists of three different modules, a transmitter-receiver unit, a display and a processor. The transmitter-receiver unit consists of LEDs (RGB) and phototransistors. The display is a three numbered seven segment display. The identification of the colour hues happens with measuring the voltage of the phototransistor. When the target is lightened up with different colours of LEDs, the voltage of the phototransistor will change with the colour hue of the target. The processor will convert the voltage into a digital form in an A/D converter which will execute different calculations for the measured voltage. After this the index rate of the colour will be transferred into the seven segment display.</p>	
Confidential	
Yes	
No <input checked="" type="checkbox"/>	
Keywords	
Machine vision, colour hue, phototransistor	
Deposited at	
Kajaani Polytechnic Library	

KÄYTETYT LYHENTEET

CIE	Comission Internationale de l'Eclairage
CMYK	Cyan, Magenta, Yellow, Key
RGB	Red, Green, Blue

SISÄLLYSLUETTELO

1	JOHDANTO	6
2	VÄRIEN TAUSTA	8
2.1	Värit	8
2.2	Laitekohtaiset värimallit	10
2.3	Laiteriippumattomat värimallit	11
2.4	Värien näkeminen.....	14
3	LAITTEEN VALMISTUS	16
3.1	Suunnittelu.....	16
3.2	Laitteiston toteutus	18
3.3	Ohjelmakoodi.....	20
4	TESTAUS	22
5	YHTEENVETO	23
	LÄHDELUETTELO	24
	LIITTEET	

1 JOHDANTO

Tämä insinöörityö on tehty Kajaanin ammattikorkeakoululle, jossa työn tilaajana on ollut laboratorioinsinööri Ismo Talus. Insinöörityön tavoitteena oli käyttää hyväksi koulussa opittuja taitoja. Itse työ sisälsi elektroniikan tuntemusta ja ohjelmointia eli työ perustui sulautettuihin järjestelmiin. Opinnäytetyönä oli kehittää laite, jolla on tarkoitus tarkastella kahta melkein samaa värisävyä olevaa kohdetta toisiinsa nähden ja ilmaista niiden ero värin indeksinä. Näin ollen laitteen nimi olisi värisävyvertailija.

Syyt, miksi insinöörityöni koskee värisävyjen erottamista toisistaan, tulevat aivan käytännöllisistä tarpeista. Esimerkiksi ihmissilmä on herkkä valon muutokselle, joka aistitaan värin muutoksena, vaikka itse perusväri ei muutu miksikään. Myös kohteen pinnanlaatu vaikuttaa värin havainnointiin. Kaiken tämän vuoksi tuli kehitellä laite, joka olisi tarkempi kuin ihmissilmä.

Värisävyero ilmaistaan punaisen, vihreän ja sinisen värin indeksinä. Indeksimuodostetaan jakamalla värisävyt punaisen värisävyllä, jolloin jäljelle jää vihreän ja sinisen indeksiluku. Jäljelle jääneille indeksiluvuille tehdään prosessorilla laskutoimituksia, jonka jälkeen ne tulostetaan 7-segmentnäytölle. Punaisen indeksi on aina yksi, joten sitä ei ole tarkoituksenmukaista tulostaa 7-segmentnäytölle.

Jotta värisävyvertailijan tuloksia voidaan tulkita oikein, on hyvä tietää miten ihmissilmä toimii. Silmän tärkeimmät komponentit ovat iiris, linssi ja verkkokalvo. Verkkokalvo on merkittävässä osassa värien näkemiseen. Verkkokalvolla on kahdentyyppisiä reseptorisoluja, sauvoja ja tappeja. Tappisolujen aallonpituusherkyys mahdollistaa värien näkemisen. Ihmisen tapaan hahmottaa värejä liittyy kolme piirrettä, värisävy, värin puhtaus ja kirkkaus.

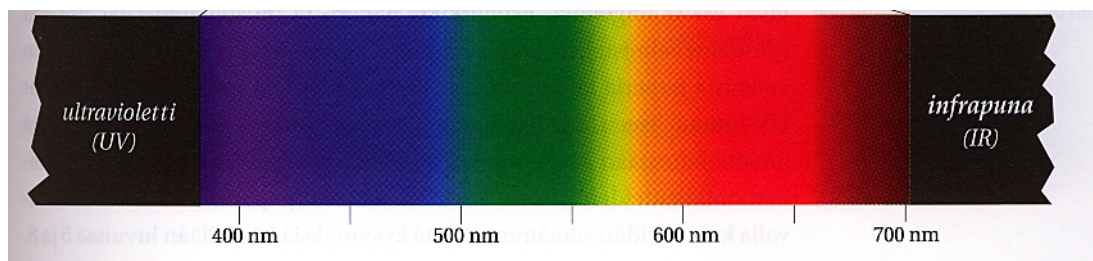
Värisävyvertailijan toimintaan oli hieman helpompi perehtyä suuntaavien opintojeni avulla. Suuntaavissa opinnoissa eli konenäössä oli käsitelty valon

aallonpituuksia ja värien sijoittumista sen alueelle, joten tämä helpotti ymmärtämään värien käyttäytymistä. Kuitenkin koska näitä asioita ei ollut käyty käytännössä laajemmin läpi, tuotti esimerkiksi LEDien valitseminen valaistus-tehon perusteella hieman ongelmia.

2 VÄRIEN TAUSTA

2.1 Värit

Mitä on väri ja mistä se tulee? Valo on aaltoliikkeenä etenevää sähkömagneettista säteilyä. Tuossa säteilyssä jokaista värisävyä vastaa oma aallonpituutensa, kuten kuvasta 1 voi havaita. Sellaista värisävyä, jota vastaavaa aallonpituutta ei ole valonlähteen säteilemässä valossa, emme voi nähdä. Näkyvää valoa lähinnä lyhytaaltoisempi säteily on ultraviolettisäteilyä ja pitkäaaltoisempi infrapunasäteilyä [1]. Spektri eli näkyvän valonaallon pituus, valaistuksesta riippuen, on 380 – 700 nm:n välillä. Värit sijoittuvat spektrille aina samaan järjestykseen, oli niitä nimetty sitten kolme tai useampi.



Kuva 1. Värien sijoittuminen spektrille [2]

Monet käyttämistämme valonlähteistä (kuten hehkulamput ja auringonvalo) tuottavat valoa tavalla, joka antaa meille kätevän termin valon värin kuvaamiseen, värilämpötilan. Jokainen tiheä kappale säteilee lämpöenergiaa. Atomit heijastavat energiaa, jota ne ovat imeneet itseensä jonkinlaisen palamisprosessin aikana. Matalilla lämpötiloilla tuo säteily on infrapuna-alueella ihmisilmän näköalueen ulkopuolella, jolloin sitä kutsutaan lämmöksi. Tutkiakseen tätä ilmiötä fyysikot kuvittelevat kappaleita, joista on poistettu kaikki valonlähteet, jolloin kaikki säteily on peräisin lämpöenergiasta. He kutsuvat näitä kappaleita mustiksi säteilijöiksi (blackbody radiators). Absoluuttinen musta kappale absorboi kaiken siihen osuvan säteilyn riippumatta säteilyn tulo- kulmasta ja aallonpituudesta. Seisottaessa pilkkopimeässä huoneessa, ollaan musta säteilijä, joka säteilee energiaa, jonka ainoastaan infrapunahavaintsin tai pöllö voi nähdä. [2.]

Heijastuminen ja läpäiseminen

Kappaleen pinnan täytyy olla vuorovaikutuksessa valon kanssa vaikuttaakseen valon väriin. Valo törmää kappaleeseen, sukeltaa jonkun matkaa pinnalla olevien atomien sisään ja ilmestyy sitten uudelleen esiin. Valon ollessa vuorovaikutuksessa noiden pinnan atomien kanssa kappale imee itseensä (absorboi) eräitä aallonpituuksia ja heijastaa toisia, joten heijastetun valon spektri-koostumus ei ole sama kuin tulevan valon. Sitä, kuinka paljon kappale heijastaa joitakin aallonpituuksia ja imee itseensä muita, kutsutaan spektriheijastussuhteeksi. Jos valonlähdettä muutetaan, ei kappaleen heijastussuhde muutu, vaikka näkyvä spektrienergia onkin erilainen. Heijastussuhde on siten kappaleen muuttumaton ominaisuus.

Läpäisevä (transmissive) kappale vaikuttaa aallonpituuksiin muuten samalla tavoin kuin äsken kuvattu heijastava kappale, paitsi että aineen pitää olla ainakin osittain läpinäkyvä, jotta valo voi edetä sen läpi. Myös se muuttaa valon aallonpituuskoostumusta imemällä itseensä joitakin aallonpituuksia ja päästämällä muut lävitseen. [2.]

Värin mittaus

Aina täytyy muistaa, ettei väriä voi mitata vaan mitattavana on valo. Tapahtuma on kolmen osatekijän välinen tapahtuma. Nämä osatekijät ovat valonlähde, kohde ja havaitsija. Laitteita joilla mittaamme ärsykettä, jonka havainnoijat lopuksi tulkitsevat värit, on kolmea päätyyppiä. Ne kaikki toimivat säteilemällä valoa, jonka spektrijakauma on tiedossa, minkä jälkeen pinnan heijastamaa tai välittämää valoa mitataan ilmaisimien avulla. Ilmaisimien on yksinkertaisesti fotonilaskuri. Se ei osaa määritellä laskemiensa fotonien aallonpituuksia, joten laitteen täytyy suodattaa ilmaisimeen menevää valoa.

Erät kolmen laitetyypin välillä, densitometrien, kolorimetrien ja spektrofotometrien, ovat niiden käyttämien suotimien lukumäärässä ja tyypissä sekä niiden anturien herkkyydessä.

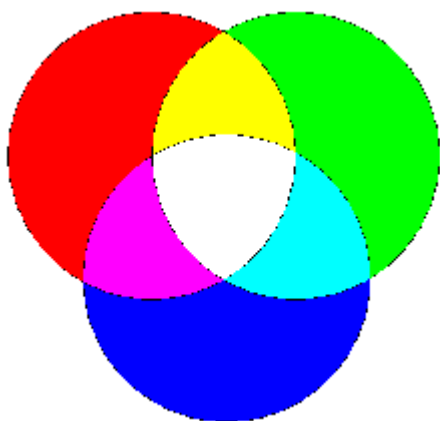
Densitometrit mittaavat tiheyttä, densiteettiä, eli kuinka paljon heijastavat pinnat imevät valoa tai läpinäkyvät pinnat päästävät sitä läpi.

Kolorimetrit mittaavat kolorimetrisiä arvoja, lukuja jotka mallintavat silmän tappisolujen reaktiota.

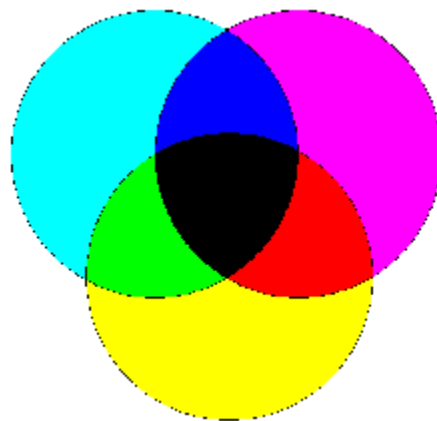
Spektrofotometrit mittaavat pinnan spektriominaisuuksia, toisin sanoen sitä, kuinka paljon valoa kullakin aallonpituudella pinta heijastaa tai välittää.

2.2 Laitekohtaiset värimallit

RGB- tai CMYK-malleja kutsutaan laitekohtaisiksi eli laiteriippuviksi värimalleiksi, jotka ovat esitettyinä kuvassa 2 ja 3.



Kuva 2. RGB-värijärjestelmä [5]



Kuva 3. CMY-värijärjestelmä [5]

Annetuista RGB- tai CMYK-arvoista syntyvä todellinen väri riippuu värin tuotavasta laitteesta. Yksinkertaisesti tämä tarkoittaa kahta asiaa:

- Sama RGB- tai CMYK-arvojen joukko tuottaa erilaisia värejä eri laitteissa (tai samalla laitteella eri paperin kanssa, jos kyse on tulostimesta tai painokoneesta)
- Tuottaaksemme saman värin eri laitteilla, täytyy muuttaa kullekin laitteelle lähettäviä RGB- tai CMYK-arvoja.

Näistä seuraa seuraavanlaisia ongelmia:

- Miten voimme tietää, mitä väriä RGB- tai CMYK-tiedoston lukujen on tarkoitus esittää? Toisin sanoen, mitä tarkoittaa ”255, 0, 0”? Se tarkoittaa punaista, mutta mitä punaista täsmälleen?
- Miten voimme tietää, mitä RGB- tai CMYK-arvoja laitteelle pitää välittää, jotta se tuottaisi halutun värin?

CMY vai CMYK? Teoriassa puhdas syaani imee itseensä 100 % punaisesta valosta aivan kuin puhdas magenta ja keltainen imevät kokonaan vihreän ja sinisen valon. Kun yksi tai useampi väriaineista ei ole sataprosenttisen puhdas, osa valosta heijastuu imeytymisen sijasta. Tämä on syy siihen, miksi monissa värijauhepohjaisissa laitteissa on vihertävä kolmivärinen musta, ja miksi painokoneen kolmivärinen musta on yleensä samean ruskea. Väriaineet eivät yksinkertaisesti ole täydellisiä. [2.]

Joten saadaksemme paremman mustan, joka imee itseensä mahdollisimman paljon valoa, käytämme mustaa mustetta. Toinen hyvä syy mustan musteen käyttöön painamisessa on se, että tekstin kaltaiset kokomustat kohteet on paljon helpompi painaa, kun ei tarvitse tasata syaania, magentaa ja keltaista versiota täydellisesti päällekkäin. Näistä syistä johtuen CMY:lle on lisätty K eli CMYK.

Lyhenne RGB tulee sanoista Red, Green ja Blue, jotka tarkoittavat värejä (punainen, vihreä ja sininen). CMYK tulee taas sanoista Cyan, Magenta, Yellow ja Key (Syaani, Magenta, Keltainen ja Key). Koska B -kirjain tarkoittaa sinistä, on mustalle pitänyt antaa jokin muu nimitys eli K.

2.3 Laiteriippumattomat värimallit

RGB ja CMYK vain kertovat laitteille, kuinka paljon väriainetta käytetään. Ne eivät kerro meille mitään laitteiden tuottamasta todellisesta väristä. CIE-mallit

kuvaavat täsmällistä väriä, jonka normaalilla värinäöllä varustettu ihminen näkisi hyvin tarkkaan määritellyissä katseluolosuhteissa. Mutta ne eivät sano mitään siitä, millä tavalla yksittäisen monitorin, skannerin tai tulostimen pitää tuottaa kyseinen väri.

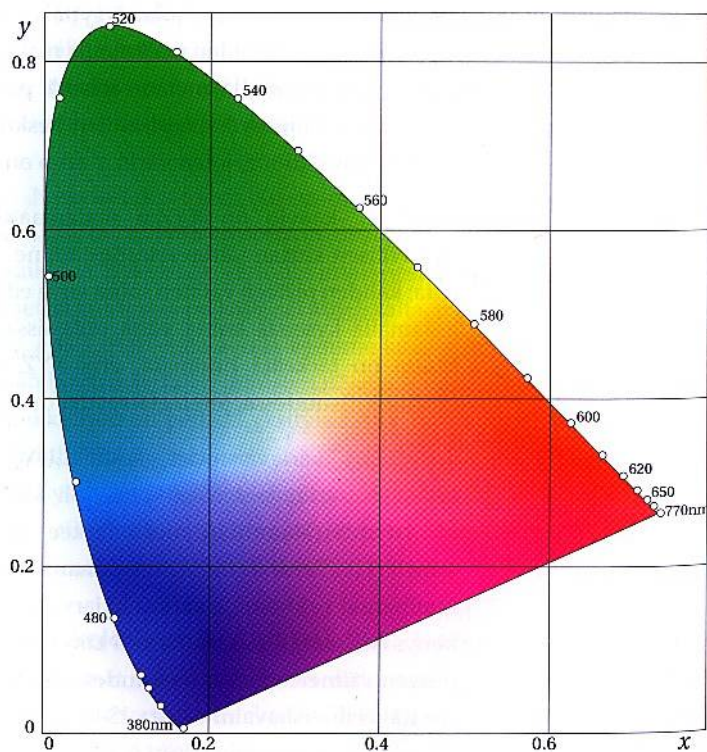
Onneksi on olemassa useita laiteriippumattomia numeerisia värimalleja. Laiteriippumattomat värimallit pyrkivät mallintamaan numeerisesti suoraan ihmisen värinäköä. Sen sijaan tietyt laitteet tarvitsevat lukuja värin tuottamiseksi. Vuonna 1931 CIE sai valmiiksi matemaattisen värimallin, jolla on kammottavan kuuloinen nimi CIE XYZ (1931). Tämä malli oli ainutlaatuinen, koska se yritti kuvata matemaattisesti värihavaintoa, jonka täsmällisesti määritetty ärsyke aiheuttaisi normaalilla värinäöllä varustetulle ihmiselle täsmällisesti määritetyssä katseluympäristössä.

Alkuperäisen mallin jälkeen CIE on tuottanut rykelmän värimalleja, joilla on yhtä hämäriä nimiä CIE L Ch, CIELUV, CIE xyY, CIELAB ja niin edelleen, jotka kaikki ovat CIE XYZ:n matemaattisia muunnoksia. Voidaksemme hallita väriä oikeassa elämässä meidän täytyy käyttää sekä laiteriippumattomia että laiteriippuvaisia värimalleja. [2.]

CIE-järjestelmä

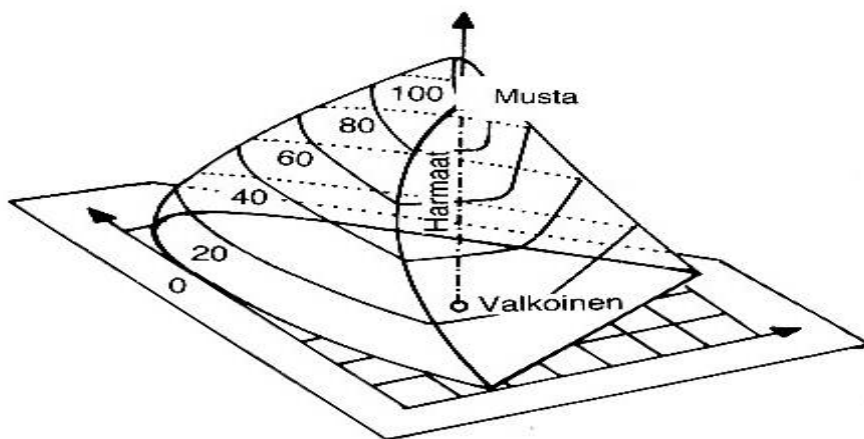
CIE-järjestelmässä voidaan jokainen ajateltavissa oleva väri ilmaista kahden koordinaatin ja valon heijastusprosentin avulla. Koordinaatit viittaavat väri-diagrammiin, jossa täysin puhtaat värit on sijoitettu venytetyn hevosenkengän muotoiselle kaarelle, kuten kuvasta 4 käy selville.

Puhtaat purppuravärit ovat saaneet paikkansa hevosenkengän kärkiä yhdistävällä purppuraviivalla. Siirryttäessä viivojen rajaamalla alueella kohti keskellä olevaa valkopistettä värien ärsykepuhtaus pienenee. Harmaa-asteikko ajatellaan sijoitetuksi kohtisuoraan kuvaustasoa vastaan siten, että valkoinen sijaitsee valkopisteessä, harmaat ylöspäin tummentuen sen yläpuolella ja musta ylimmässä huipussa.



Kuva 4. xy -kromaattisuuskaavio [2]

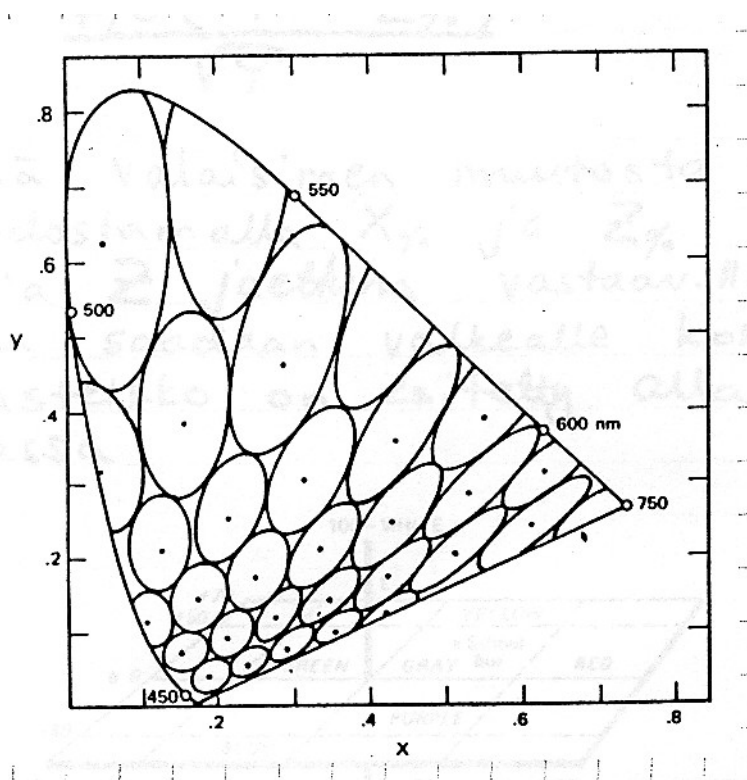
Kuva 5 selventää tätä valkean ja mustan sijoittumista hevosenkengälle. Valkoisen ja mustan välille mahtuu teoreettisesti rajaton määrä vaaleuseroja.



Kuva 5. CIE-värikappale [1]

Eräs heikkous, joka liittyy CIE xyY -asteikkoon, joka tunnetaan myös muodossa CIE Yxy , on siinä, että xy -tasossa ei ole tasavälinen. Tätä on havainnointu kuvassa 6, jossa on esitetty eri pisteiden ympärillä alueet, jotka vastaavat yhtä

suurta värieroa. Kuten kuvasta havaitaan, alueet ovat paljon suurempia vihreällä alueella kuin sinisellä alueella.



Kuva 6. Värialueiden suuruudet [3]

2.4 Värien näkeminen

On arvioitu, että ihminen pystyy erottamaan noin 160 väriä. Näiden lisäksi pystymme aistimaan erilaisia väri-intensiteetin (valoisuuden) asteita sekä erilaisia värikyllästeisyysasteita (värikylläisyys vähenee, jos esimerkiksi punaiseen sekoitetaan valkoista tai harmaata). Nämä tekijät voivat yhdistyä eri tavoin, joten erilaisia havaittavia värivivahteita on hyvin suuri määrä.

Tappisoluja on kummassakin silmässä yli viisi miljoonaa. Ne ovat erikoistuneet värien erottamiseen eivätkä toimi heikossa valaistuksessa.

Verkkokalvon tappisoluja on värisävyjen runsaudesta huolimatta ilmeisesti kuitenkin vain kolmenlaisia. Kussakin tappilajissa on yhtä valoon reagoivaa väriainetta. Yksi tappilaji reagoi parhaiten siniseen, yksi vihreään ja yksi punaiseen valoon. Tappisolujen kolmen valoon reagoivan väriaineen avulla

voidaan selittää värisokeuden lajit. Oikeampi nimi olisi tosin väriheikkous, sillä vika on harvoin täydellinen. Tavallisin värisokeuden muoto on punavihersokeus. Henkilöltä puuttuu joko punaherkkä tai viherherkkä tappiväriaine, eikä hän pysty erottamaan punaista ja vihreää toisistaan. [4.]

Lisäävät ja vähentävät päävärit

Ihmisen verkkokalvon trikromaattinen rakenne mahdollistaa additiiviset päävärit. Trikromaattisuus tarkoittaa, että meillä on kolmenlaisia tappisoluja värejä varten. Jos valitset kolme valonlähdetä, jotka jakavat näkyvän spektrin karkeasti ottaen kolmeen osaan ja joiden spektrijakaumat ovat osittain päällekkäin, kukin niistä lisää aallonpituuksia, jotka ärsyttävät yhtä tai useampaa silmän kolmesta reseptorista. Jaettaessa spektri karkeasti ottaen kolmeen osaan, saadaan kolme valonlähdetä, joita voisi kutsua punaiseksi, vihreäksi ja siniseksi. Kun lähdetään liikkeelle mustasta (ei aallonpituuksia), kukin kolmesta väristä lisää aallonpituuksia, kunnes päädymme valkoiseen (kaikkia aallonpituuksia tasapuolisesti). Tästä tulee nimitys additiivinen eli lisäävä.

Myös subtraktiivisten päävärien (syaanin, magentan ja keltaisen) perusta on trikromaattisuudessa. Sen sijaan, että lisääisivät aallonpituuksia mustaan, ne vähentävät aallonpituuksia valkoisesta valonlähteestä. Toisin sanoen termi ”syaani muste” tarkoittaa ”pitkien aallonpituuksien vähentäjää” tai yksinkertaisesti ”punaisen vähentäjää”. Se vähentää pitkiä (punaisia) aallonpituuksia valkoisesta valosta (kuten paperista heijastuneesta valosta). Samaan tapaan magenta muste vähentää keskipituisia aallonpituuksia eli vihreitä. Keltainen vähentää lyhyitä aallonpituuksia eli sinistä. [2.]

3 LAITTEEN VALMISTUS

3.1 Suunnittelu

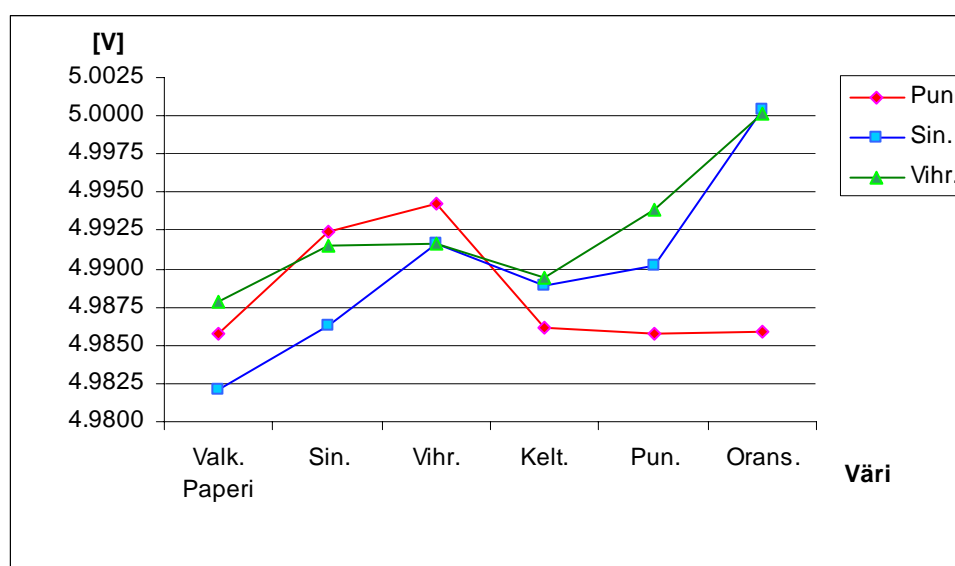
Laitteen suunnittelu alkoi miettimällä laitteen toimintoja ja tärkeimpiä osia, kuten lähetin-vastaanotinta ja näyttöä. Lähetin-vastaanottimen tuli olla mahdollisimman pienessä tilassa. Itse vastaanottimen eli valotransistorin tuli käsittää näkyvän valonaallon pituusalue. Kokonaiskuvan tultua selväksi alkoi tarvittavien komponenttien valinta. Valmistus aloitettiin tekemällä ns. koe-yksikkö. Mahdollisimman yksinkertainen koe-yksikkö koostui pelkästään lähetin-vastaanottimesta ja sellaisista osista, jotka olisivat saatavilla heti. Tällä kokeilulla tuli selvittää millaisilla valaisimilla tuleva laitteisto toimisi. Lähtöajatuksena oli, että laitteiston valaisin eli lähetinyksikkö koostuisi erivärisistä LEDeistä. Jos ei toimisi, siirryttäisiin tavallisiin hehkulanka-valaisimiin.

Koe-yksikön valmistuttua alkoi LEDien valaistustehon tutkiminen. Valaistustehon tutkiminen aloitettiin asettamalla yksi LEDi per valotransistori. Ensimmäiset kokeilut eivät vaikuttaneet lupaavilta. LEDin valaistusteho ei ollut riittävä, jotta valotransistorin jännitteen muutoksia voitiin pitää tarpeeksi suurina. Tämän jälkeen lisättiin LEDien määrä kahteen per valotransistori. Vieläkin vaikutti pahasti siltä, ettei jännitteen muutos ollut tarvittavan suuri.

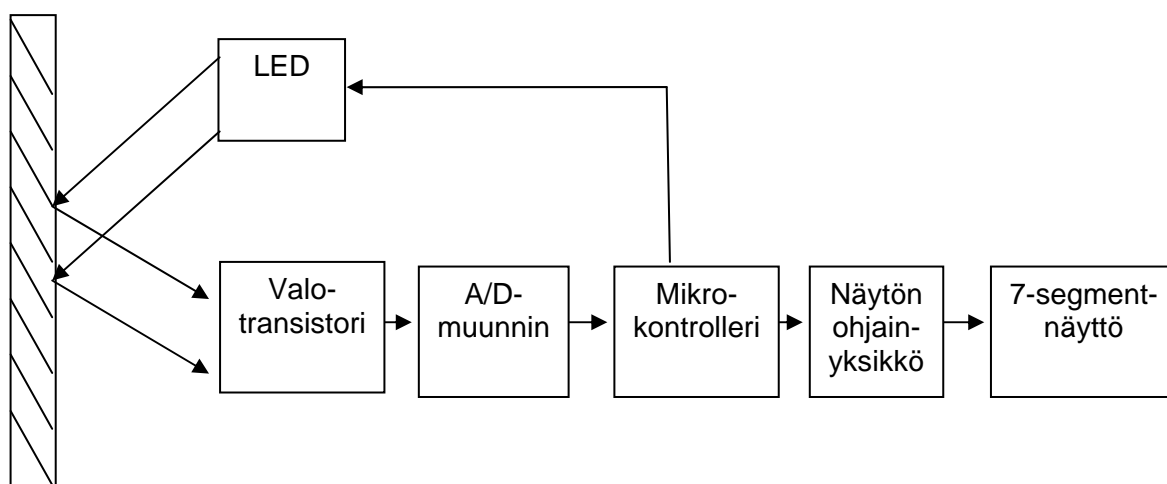
Tätä hieman pohdittuani päätin vaihtaa erimalliset LEDit. Uudet LEDit olivat valaistusteholtaan suurempia, sekä niiden valaistusalue oli suurempi. Nyt vaikutti siltä, että valotransistorin jännitteen muutos olisi tarpeeksi suuri, jotta voitaisiin tunnistaa eri värisävyt toisistaan. Mitattavina testiväreinä käytettiin erivärisiä paperiliuskoja. Valotransistorin ja LEDien käyttöjännitteenä oli 5 voltia ja LEDien määrä oli kaksi LEDiä per valotransistori. Kuvasta 7 voidaan nähdä, miten eri värit vaikuttivat valotransistorin yli olevaan jännitteeseen.

Kokeilu osoitti, että värierot voidaan erottaa toisistaan näillä komponenteilla.

Kokeiluiden jälkeen tuli miettiä, miten tätä kokonaisuutta voitaisiin ohjata. Pääkomponentin eli mikrokontrollerin tuli pitää sisältää A/D-muuntimen. Siinä myös tulisi olla tarpeeksi monta ohjelmoitavaa kanavaa ohjaamaan muita laitteistoon kuuluvia osia, esimerkiksi LEDejä. Myös kontrolleriin kuuluvat muistit olisivat tärkeitä. Mikrokontrollerin tuli kuulua 8051-prosessoriperheeseen, koska valmiina oleva ohjelmointilaitteisto tuki 8051-perheeseen kuuluvia piirejä. Lopputulos oli, että päädyttäisiin Atmellin T89C51CC01 -mikrokontrolleriin.



Kuva 7. Valotransistorin yli oleva jännite



Kuva 8. Toiminnallinen lohkokaavio

Koeyksikkö oli koottu koekytentäalustalle, joten kun tarvittavat osat oli tilattu, oli vuorossa piirilevyn suunnittelu ja sen jrsintä. Piirilevyn suunnittelu tapahtui

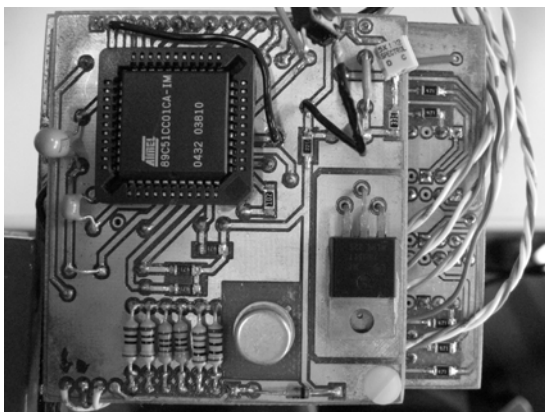
Pads – ohjelmistolla. Tämä kaikki oli uutta ja tuntematonta, joten itse ohjelmaan tutustuminen vei aika ison osan aikaa itse työstä. Piirilevyt tuli suunnitella kaksipuoleiseksi, jotta kaikki tarvittavat osat saataisiin mahtumaan mahdollisimman pieneen tilaan. Laitteen toiminnallinen lohkokaavio on esitettyä kuvassa 8.

3.2 Laitteiston toteutus

Laitteisto koostuu kolmesta moduulista eli lähetin-vastaanotinyksiköstä, näyttö yksiköstä ja prosessorikortista. Laite sisältää kaksi kaksipuolista piirilevyä. Toinen piirilevyistä sisältää mikrokontrollerin ja sen oheiskomponentit, ja toinen piirilevy sisältää 7-segmentnäytön ja sen ohjainpiirit. Kytkenäkaaviot ovat liitteinä A ja B.

Piirilevyllä, jolla mikrokontrolleri sijaitsee, on myös joukko muita komponentteja joista on seuraavassa lyhyt kuvaus. Käyttöjännitteenä laitteistolla ja itse mikrokontrollerilla tuli olla 5 voltia. Jotta jännite saataisiin halutuksi 9 voltin paristosta, tarvittaisiin siihen soveltuva regulaattori. Valotransistoreille piti säätää jännite 3 volttiin, koska mikrokontrollerin analoginen sisääntulo kestäisi vain tämän 3 voltia. Säätö tapahtui diodin ja vastusten avulla. LEDien ohjausta varten rakennettiin transistorikytkin. Kytkintä ohjataan mikrokontrollerin kautta ohjelmallisesti. Jotta itse mikrokontrolleri toimisi, tuli sille asettaa ulkoinen kide. Kiteen taajuus on 12MHz. Mikrokontrolleriin asennettava ohjelmakoodi sijaitsi flash muistissa, jonka valinta tapahtui asettamalla mikrokontrollerin EA -linja (external access) käyttöjännitteeseen. Kuvassa 9 on nähtävissä mikrokontrolleri.

Piirilevy jolla 7-segmentnäyttö yksikkö sijaitsee, sisältää joitakin komponentteja joista on seuraavassa lyhyt kuvaus. Käyttöjännite on sama 5 voltia. Näyttö yksikkö koostuu kolmesta 7-segmentnäytöstä, joka on esitetty kuvassa 10. Jokaisella näytöllä on oma BCD-dekooderi. Jotta näytölle voidaan tulostaa jotakin, tarvitsee se myös teho-ohjaimen. Myös näitä teho-ohjaimia on yksi per näyttö. 7-segmentnäyttö koostuu seitsemästä elementistä, joilla jokaisella on oma etuvastuksensa.



Kuva 9. Mikrokontrolleri

Lähetin-vastaanotinyksikkö sisältää ainoastaan LEDit ja valotransistorit. Tämä yksikkö ei ole piirilevyllä, vaan se on muovista jyrskyssä ”kotelossa”. Piirilevyjen jyrkimisen jälkeen oli vuorossa osien juottaminen kiinni. Kun osat olivat paikallaan, tuli juotokset tarkastaa mahdollisten kylmäjuotosten varalta. Tarkastus tapahtui yleismittarilla. Piirilevyjen ollessa valmiita ja tarkastettuja, alkoi ohjelmakoodin kehittäminen. Käytetyt komponentit ovat listattu liitteeseen D-1 ja D-2.



Kuva 10. 7-segmentnäyttö yksikkö

3.3 Ohjelmakoodi

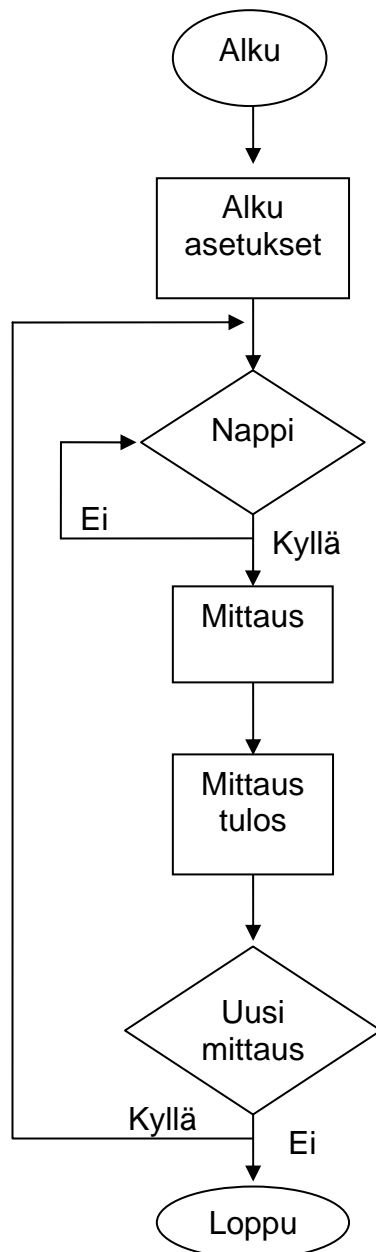
Ohjelmointikielenä toimi C-kieli ja itse koodaaminen tapahtui IAR -ohjelmalla. Tämä ohjelma ei ollut paras mahdollinen laitteistokoodaukseen, koska sen virheenkorjaus ei ole millään tavalla järkevä tai käytännöllinen.

Ohjelman lataus mikrokontrollerille tapahtui erillisellä laitteella (Dataman-48). Tämä ei ollut järkevin mahdollinen tapa ladata koodi mikrokontrollerille. Tällä tavalla piiri tuli aina irrottaa ja viedä ohjelmointi laitteelle. Tämä tuli rasittamaan piirilevyyn kiinnitettyä kantaa, joka lopulta irtosi juotoksistaan. Kaikki tämä oli myös aikaa vievää touhua.

”Se mitä virheitä rautapuolella tehdään, voidaan korjata koodaamalla”, pätee myös tässä työssä. Pahin virhe tuli kytkettäessä 7-segmentin ohjainpiirejä. Tässä tapauksessa ohjainpiirin vedot menivät ristiin. Tämä ristiin meneminen aiheutti tulostukseen epämääräisiä merkkejä. Onneksi ristiin kytkennän aiheuttamat virheet voitiin korjata ohjelmakoodilla.

Pääpiirteissään ohjelmakoodi sisälsi mikrokontrollerin porttien alustamisen, sekä A/D-muuntimen ohjain koodin. Lyhykäisyydessään ohjelman kulku oli kuvan 11 mukainen, joka esittää ohjelmallista vuokaaviota. Koodi on liitteenä C.

Mittauksen kulku alkaisi napin painalluksella, jonka jälkeen mikrokontrolleri antaisi käskyn sytyttää LEDit. Kun LEDit ovat olleet päällä jonkin aikaa, mikrokontrolleri lukisi valotransistorin yli olevan jännitteen arvoja joitakin kymmeniä kertoja, jonka jälkeen se muuttaisi ne digitaaliseen muotoon. Näistä arvoista tehtäisiin keskiarvostaminen ja jolle tehtäisiin värin indeksien muodostaminen. Indeksien muodostaminen tapahtuu jakamalla sinisen arvo punaisella sekä vihreän arvo punaisella. Näin ollen saataisiin kaksi lukua, jotka olisivat värin indeksi arvoja. Nämä arvot tulostettaisiin 7-segmentnäytölle. Näillä vihreän ja sinisen indeksi arvoilla ilmaistaisiin värisävy.



Kuva 11. Ohjelmallinen vuokaavio

Mittauksen suoritettua, näyttöön jää vilkkumaan sinisen ja vihreän indeksi. Jos laitteesta katkaistaan virrat, edellisen mittauksen tulokset pyyhkiytyvät pois. Myös start napin painalluksesta laitteisto suorittaa uuden mittauksen ja esittää uudet tulokset.

4 TESTAUS

Varsinainen testaus suoritettiin koeyksiköllä. Tällä tavoin pystyttiin toteamaan LEDien ja valotransistorien toimivuus tulevassa kytkennässä. Testeillä haettiin optimaalista valaistus tehoa jokaiselle värilliselle LEDi parille. Valaistustehon määrittelemisen oli siinä määrin tärkeää, että kaikkien valotransistorien jännite muutokset saataisiin mahdollisimman lähelle toisiaan. Tästä esimerkkinä on kuva 7. Varsinaista lopputestausta ei ollut mahdollista suorittaa ohjelmakoodin keskeneräisyyden vuoksi.

5 YHTEENVETO

Insinööriyön tavoitteena oli tutkia ja rakentaa värisävyeroja tunnistava laite. Laite tuli toimia yhdeksän voltin paristolla ja sen tuli olla mahdollisimman pienessä paketissa. Työn suorittaminen alkoi tutkimalla olisiko laitteen lähetinyksikkö mahdollista rakentaa LEDeillä, sekä tutkimalla sopivaa vastaanotinta.

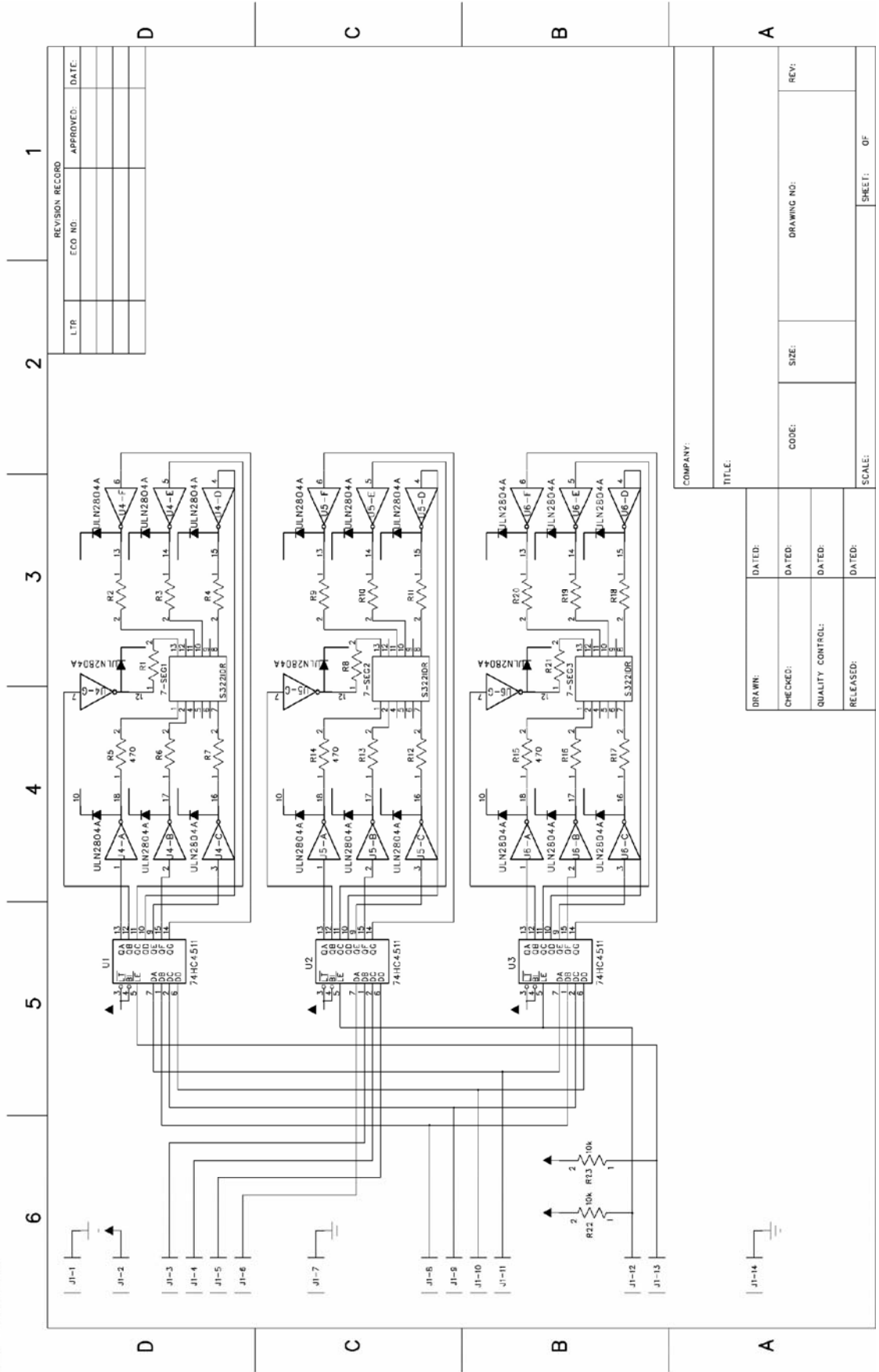
Värisävyvertailija on ainakin käytettyjen osien puolesta toimiva ratkaisu värisävyjen erottamiseksi toisistaan, kuten koekytkenän testaukset sen osoittivat. Laitteen lopputestaus jäi puuttumaan ohjelmakoodin toimimattomuuden vuoksi.

Parannettavaa työhön jäi jonkin verran. Esimerkiksi mittaustuloksen muistiin tallentaminen ja tulosten sieltä hakemisessa, olisi mahdollista käyttää eri muistialuetta laajemmin ja monipuolisemmin. Erityisen huomionarvoiseksi tulee sarjaliikenteen puuttuminen. Sarjaliikenne mahdollistaisi nopeamman ja helpomman ohjelmointitavan mikrokontrollerille. Myös koodin toiminnan seuranta ja sen virheen korjaus tulisi helpommaksi sarjaliikenteen avulla.

LÄHDELUETTELO

- 1 Bruce Fraser, Chris Murphy, Fred Bunting, Värinhallinta, Helsinki: Edita Prima, 2004. 524 s. ISBN 951-826-714-6.
- 2 Niensted W, Hänninen O, Arstila A, Björkqvist S. Ihmisen fysiologia ja anatomia, 7.painos. Porvoo: WSOY, 1991. 635 s. ISBN 951-0-16801-7.
- 3 Pentti Romppainen, Aaltoanturit. Kajaanin ammattikorkeakoulu: opetusmoniste 2002.
- 4 Seppo Rihlama. Värioppi, 6. uusittu painos. Tampere. Tammerpaino Oy, 1997. 175 s. ISBN 951-682-413-7.
- 5 <http://www.csc.fi/gopas/kuvat/liite/kuva30.html#REF32788> (13.9.2005)

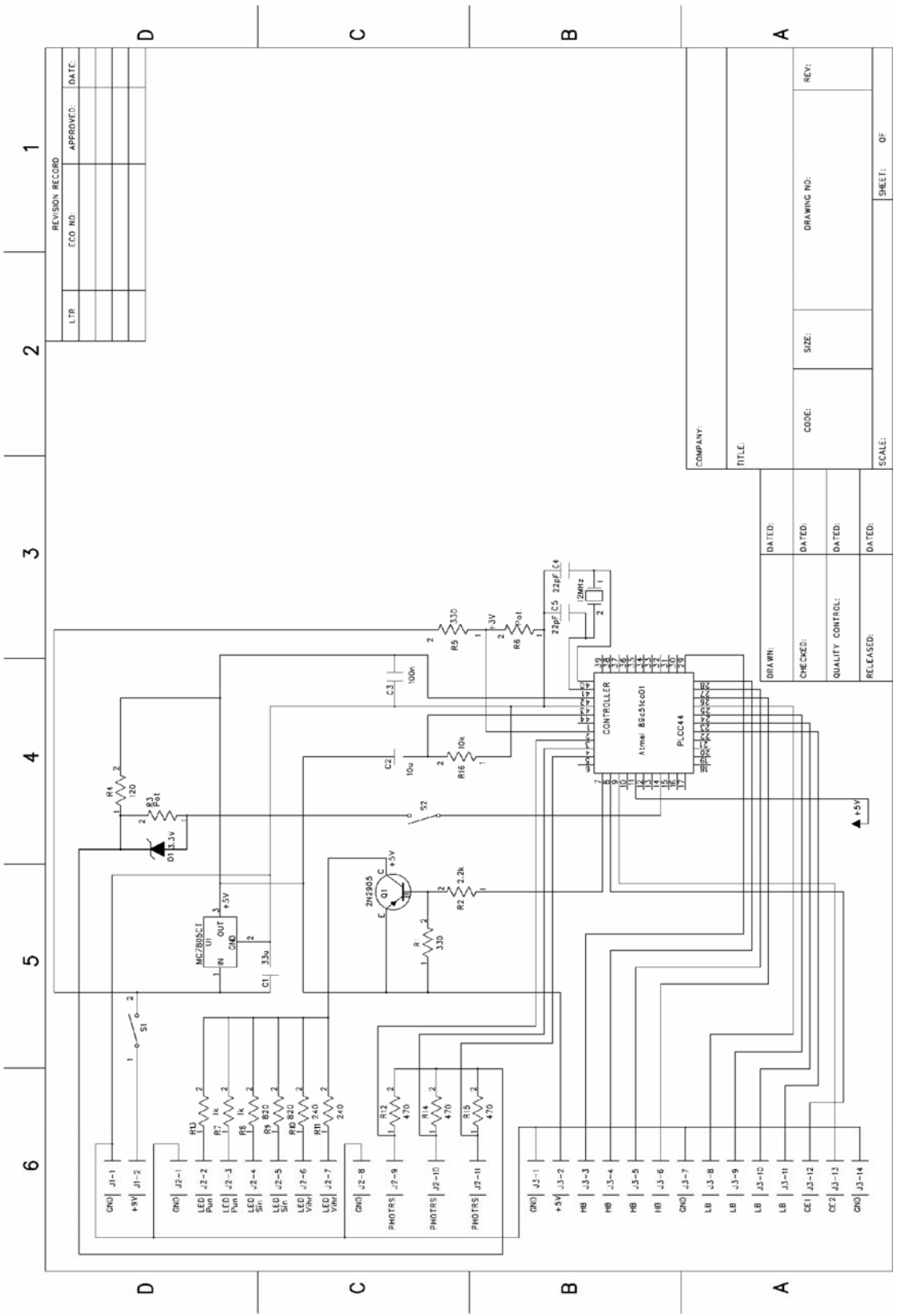
Myrskylampi 1 - Fin 004 004 1317 08 2005



REVISION RECORD			
LTR	ECO NO.	APPROVED:	DATE:

COMPANY:		TITLE:	
DRAWN:	DATED:	CODE:	SIZE:
CHECKED:	DATED:	DRAWING NO:	
QUALITY CONTROL:	DATED:	REV:	
RELEASED:	DATED:	SCALE:	SHEET: OF

PROJEKTI 17 - TÄMMEKÖNTELEFONIN 2005



REVISION RECORD			
LTR	ECO NO.	APPROVED:	DATE:

COMPANY:		TITLE:	
CODE:	SIZE:	DRAWING NO.:	REV.:
SCALE:	SHEET:	OF:	

DATE:	DATE:	DATE:	DATE:
CHECKED:	QUALITY CONTROL:	RELEASED:	

```

#include<io89c51cc01.h>

unsigned int vanha, uusi_arvo;
unsigned int data_A = 9; /*Sadat*/
unsigned int data_B = 9; /*Kympit*/
unsigned int data_C = 9; /*Ykköset*/
unsigned int value_converted = 0x0000; /* converted value */
unsigned int value_AN0 = 0x0000; /* converted AN0 value */
unsigned int value_AN1 = 0x0000; /* converted AN1 value */
unsigned int value_AN2 = 0x0000; /* converted AN2 value */
unsigned int GREEN_data;
unsigned int BLUE_data;

double RED; /* Punaisen indeksi (käytännössä 1) */
double GREEN; /* Vihreän indeksi */
double BLUE; /* Sinisen indeksi */

bit end_of_conversion = 0; /* software flag */
bit ulkoinen_kest = 0;

void LED_ON();
void Mittaus();
void LED_OFF();

void Laske_vari();
void delay(int s);
void Tulosta_data();
void Printtaa_s();
void Printtaa_k();
void Printtaa_y();
void Printtaa_yk();
interrupt void adc_int (void);

int tens;
int ones;

/*
 * FUNCTION_PURPOSE:this function setup Adc with channel 0, 1 and 2 and start
 * 10bits conversion.
 * FUNCTION_INPUTS:void
 * FUNCTION_OUTPUTS:void
 */

void main(void)
{

P1 = 0x07; /* Asetetaan portti P1:n linjat 0,1,2 Inputeiksi
           * ja muut linjat Outputteiksi */
P2 = 0x00; /* Asetetaan portti P2 Output tilaan */
P1_5 = 1; /* Asetetaan CE1 linja kirjoitus valmiiksi */

```

```

P1_6 = 1; /* Asetetaan CE2 linja kirjoitus valmiiksi */
P1_4 = 1; /* LED:ien ohjauslinja High tilaan (LED:n sytytys/sammutus) */

INT0 = 1; /* Asetetaan INT0 linja Input tilaan */
EX0 = 1; /* Sallitaan ulkoinen käynnistys (Nappi)*/
ADCF = 0x07; /* configure channel P1.0(AN0),P1.1(AN1)and P1.2(AN2) */

/* init prescaler for adc clock
 * Fadc = Fperiph/(2*(32-PRS)), PRS -> ADCLK[4:0] */
ADCLK = 0xF6; /* Fosc = 16 MHz, Fadsc = 153.8khz ( 0x06)*/
ADCON = 0x20; /* Enable the ADC */

EA = 1; /* enable interrupts */
EADC = 1; /* enable ADC interrupt */

while(1)
{
if (ulkoinen_kesk == 1)
{
EA = 0;
delay (20);
EA = 1;
ulkoinen_kesk = 0;
LED_ON();
delay(300);
Mittaus();
LED_OFF();
Laske_vari();
} /*if end*/
} /*while end*/
} /*main end*/
/*****
Aliohjelma viiveelle.
*****/
void delay(int s)
{
int i,j;

for(i=0;i<s;i++)
for(j=0;j<s;j++);
}
/*****
Aliohjelma LED:ien sytyttämiseksi.
*****/
void LED_ON()
{
P1_4 = 0; /* Sytytetään LED:t */
delay (100);
}

```

```

/*****
Aliohjelma valittujen linjojen munnokselle.
*****/
void Mittaus()
{
  /* end_of_conversion=0; */
  ADCON &= ~0x07;      /* Clear the channel field ADCON[2:0] */
  ADCON |= 0x00;      /* Select channel 0 */
  ADCON &= ~0x40;     /* standard mode */
  ADCON |= 0x08;      /* Start conversion */

  while(!end_of_conversion); /* wait end of conversion */
  end_of_conversion = 0;    /* clear software flag */
  value_AN0 = value_converted; /* save RED:s converted value */

  /*test 0*/
  data_A = ( value_AN0/100);
  data_B = ( value_AN0%100)/10;
  data_C = ( value_AN0%10);
  Tulosta_data();
  delay(200);

  ADCON &= ~0x07;      /* Clear the channel field ADCON[2:0] */
  ADCON |= 0x01;      /* Select channel 1 */
  ADCON &= ~0x40;     /* standard mode */
  ADCON |= 0x08;      /* Start conversion */

  while(!end_of_conversion); /* wait end of conversion */
  end_of_conversion = 0;    /* clear software flag */
  value_AN1 = value_converted; /* save GREEN:s converted value */

  /*test 1*/
  data_A = ( value_AN1/100);
  data_B = ( value_AN1%100)/10;
  data_C = ( value_AN1%10);
  Tulosta_data();
  delay(200);

  ADCON &= ~0x07;      /* Clear the channel field ADCON[2:0] */
  ADCON |= 0x02;      /* Select channel 2 */
  ADCON &= ~0x40;     /* standard mode */
  ADCON |= 0x08;      /* Start conversion */

  while(!end_of_conversion); /* wait end of conversion */
  end_of_conversion = 0;    /* clear software flag */
  value_AN2 = value_converted; /* save BLUE:s converted value */

  /*test 2*/
  data_A = ( value_AN2/100);

```

```

data_B = ( value_AN2%100)/10;
data_C = ( value_AN2%10);
Tulosta_data();
delay(200);
}
/*****

Aliohjelma LED:ien sammuttamiselle.
*****/
void LED_OFF()
{
P1_4 = 1;
delay (100);
}

/*****

*****/
void Tulosta_data()
{
Printtaa_s(); /* tulostaa puretun arvon 7-segmentille */
Printtaa_k(); /* tulostaa puretun arvon 7-segmentille */
Printtaa_y(); /* tulostaa puretun arvon 7-segmentille */
Printtaa_yk();
delay(200);
}
/*****

Suoritetaan numeron debukkaus, jonka
jälkeen tulostetaan "sadat" 7-segmentille.
*****/
void Printtaa_s()
{
P1_5 = 1; /* CE1 ei salli lukitusta*/
P1_6 = 0; /* CE2 sallii lukituksen*/
delay (100);

switch (data_A)
{
case 0:
P2=0x00;
break;
case 1:
P2=0x80;
break;
case 2:
P2=0x10;
break;
case 3:
P2=0x90;
break;
case 4:

```

```

        P2=0x20;
        break;
    case 5:
        P2=0xA0;
        break;
    case 6:
        P2=0x30;
        break;
    case 7:
        P2=0xB0;
        break;
    case 8:
        P2=0x40;
        break;
    case 9:
        P2=0xC0;
        break;
    }
    delay(100);
    P1_5 = 1;
    P1_6 = 1;
    delay (100);
}
/*****
Suoritetaan numeron debukkaus, jonka
jälkeen tulostetaan kymppit tens:iin.
*****/
void Printtaa_k()
{
    switch (data_B)
    {
        case 0:
            tens = 0x00;
            break;
        case 1:
            tens = 0x08;
            break;
        case 2:
            tens = 0x01;
            break;
        case 3:
            tens = 0x09;
            break;
        case 4:
            tens = 0x02;
            break;
        case 5:
            tens = 0x0A;
            break;
        case 6:

```

```

        tens = 0x03;
        break;
    case 7:
        tens = 0x0B;
        break;
    case 8:
        tens = 0x04;
        break;
    case 9:
        tens = 0x0C;
        break;
    }
}
/*****
    Suoritetaan numeron debukkaus, jonka
    jälkeen tulostetaan ykköset ones:iin.
    *****/
void Printtaa_y()
{
    switch (data_C)
    {
        case 0:
            ones = 0x00;
            break;
        case 1:
            ones = 0x80;
            break;
        case 2:
            ones = 0x10;
            break;
        case 3:
            ones = 0x90;
            break;
        case 4:
            ones = 0x20;
            break;
        case 5:
            ones = 0xA0;
            break;
        case 6:
            ones = 0x30;
            break;
        case 7:
            ones = 0xB0;
            break;
        case 8:
            ones = 0x40;
            break;
        case 9:
            ones = 0xC0;

```



```

        break;
    }
}
/*****
    Lasketaan yhteen ykköset ja kymmenet, jonka
    jälkeen tulostetaan ne 7-segmentille.
    *****/
void Printtaa_yk()
{
    P1_5 = 0; /* CE1 sallii lukituksen */
    P1_6 = 1; /* CE2 ei salli lukitusta */
    delay(100);

    P2 = tens + ones;

    delay(100);
    P1_5 = 1;
    P1_6 = 1;
    delay(100);
}
/*****
    Lasketaan vihreän ja sinisen indeksit
    *****/
void Laske_vari()
{
    /*RED = (double)value_AN0/(double)value_AN0;*/
    GREEN = (double)value_AN1/(double)value_AN0;
    BLUE = (double)value_AN2/(double)value_AN0;

    /*laske ja muokkaa vihreä/punainen*/

    GREEN_data = GREEN * 10000;

    if( GREEN_data >= 10000)
        GREEN_data = GREEN_data - 10000;
    else
        GREEN_data = GREEN_data - 9000;

    data_A = (GREEN_data/100);
    data_B = (GREEN_data%100)/10;
    data_C = (GREEN_data%10);
    Tulosta_data();
    delay(200);
    /* vihreä/punainen valmis */

    data_A = 2;
    data_B = 0;
    data_C = 0;
    Tulosta_data();

```

```

delay(200);

/*laske ja muokkaa sininen/punainen*/
BLUE_data = BLUE * 10000;

if( BLUE_data >= 10000)
    BLUE_data = BLUE_data - 10000;
else
    BLUE_data = BLUE_data - 9000;

data_A = (BLUE_data/100);
data_B = (BLUE_data%100)/10;
data_C = (BLUE_data%10);
Tulosta_data();
delay(200);
/* sininen/punainen valmis */

data_A = 3;
Tulosta_data();
delay(200);
}
/*
 * FUNCTION_PURPOSE:Adc interrupt, save ADDH and ADDL into an unsigned int
 * FUNCTION_INPUTS:void
 * FUNCTION_OUTPUTS:void
 */
interrupt void adc_int (void)
{
    ADCON &= ~0x10;          /* Clear the End of conversion flag */
    value_converted = ADDH<<2; /* save 8 msb bits */
    value_converted |= (ADDL & 0x03); /* save 2 lsb bits */
    end_of_conversion = 1;    /* set flag */
}
interrupt void extern0_int (void)
{
    ulkoinen_kesk = 1;
}

```

Bill Of Materials for Näyttö.sch on Fri Sep 23 10:31:51 2005				

Reference	Part Name	Manufacturer	Description	QTY

U1-3	74HC4511	TEXAS	BCD TO 7 SEGMENT LATCH/ INSTRUMENTS DECODER/ DRIVER	3
J1	CON-SIP-14P		GENERIC 14 PIN SIP HEADER .100 CENTERS	1
R1-21	RES1206,470	IPC SM-782	SURFACE MOUNT RESISTOR 0.062 STD. X 0.126 INCHES, 1/8W	21
R22-23	RES-1W,10k		RES BODY:250 CENTERS:800	2
7-SEG1-3	S322IDR			3
U4-6	ULN2804A	MOTOROLA	OCTAL PERIPHERAL DRIVER ARRAY	3

Bill Of Materials for Prossu.sch on Fri Sep 23 10:26:56 2005				

Reference	Part Name	Manufacturer	Description	QTY

Q1	2N2905	MOTOROLA	GENERAL PURPOSE PNP SILICON TRANSISTOR	1
C2	CAP-CC05,10u, ???	KEMET	RADIAL CERAMIC CAPACITOR, MIL-SPEC SIZE CK05	1
C4-5	CAP-CC05,22pF, ???	KEMET	RADIAL CERAMIC CAPACITOR, MIL-SPEC SIZE CK05	2
C1	CAP-CC05,33u, ???	KEMET	RADIAL CERAMIC CAPACITOR, MIL-SPEC SIZE CK05	1
C3	CAP-CK05,100n, ???	KEMET	RADIAL CERAMIC CAPACITOR, MIL-SPEC SIZE CK05	1
J2	CON-SIP-11P		GENERIC 11 PIN SIP HEADER .100 CENTERS	1
J3	CON-SIP-14P		GENERIC 14 PIN SIP HEADER .100 CENTERS	1
J1	CON-SIP-2P		GENERIC 2 PIN SIP HEADER .100 CENTERS	1
U1	MC7805CT REGULATOR;	MOTOROLA	POSITIVE VOLTAGE FIXED +5.0 VOLTS	1
	CONTROLLER PLCC44	Atmel 89c51cc01		1
R4	RES1206,120	IPC SM-782	SURFACE MOUNT RESISTOR 0.062 X 0.126 INCHES, 1/8W STD.	1
R2	RES1206,2.2k	IPC SM-782	SURFACE MOUNT RESISTOR 0.062 X 0.126 INCHES, 1/8W STD.	1
R1 R5	RES1206,330	IPC SM-782	SURFACE MOUNT RESISTOR 0.062 X 0.126 INCHES, 1/8W STD.	2
R12	RES1206,470	IPC SM-782	SURFACE MOUNT RESISTOR 0.062 X 0.126 INCHES, 1/8W STD.	3
R14-15				
R3	RES1206,Pot	IPC SM-782	SURFACE MOUNT RESISTOR 0.062 X 0.126 INCHES, 1/8W STD.	1
R6	RES1206,Pot.	IPC SM-782	SURFACE MOUNT RESISTOR 0.062 X 0.126 INCHES, 1/8W STD.	1
R7 R13	RES-1/8W,1k		RES BODY:060 CENTERS:400	2
R10-11	RES-1/8W,240		RES BODY:060 CENTERS:400	2
R8-9	RES-1/8W,820		RES BODY:060 CENTERS:400	2
R16	RES-1W,10k		RES BODY:250 CENTERS:800	1
S2	SWITCH-ST		WIRE CONNECTION	1
S1	SWITCH-ST,???		WIRE CONNECTION	1
Y1	XTAL1,12MHz		CRYSTAL	1
D1	ZENER-DIODE, 3.3V		GENERIC ZENER-DIODE	1