

Saku Rantamäki

SULAUTETTUIJEN JÄRJESTELMIEN KÄYTTÖ  
PERINTEISISSÄ LAUTAPELEISSÄ

Tietotekniikan koulutusohjelma

Tietoliikennetekniikka

2009



Satakunnan ammattikorkeakoulu  
Satakunta University of Applied Sciences

## SULAUTETTUIJEN JÄRJESTELMIEN KÄYTTÖ PERINTEISISSÄ LAUTAPELEISSÄ

Rantamäki, Saku  
Satakunnan ammattikorkeakoulu  
Tietotekniikan koulutusohjelma  
Heinäkuu 2009  
Ekholm, Ari  
Sivumäärä: 39

Asiasanat: sulautetut järjestelmät, lautapelit, RFID

---

Tämän opinnäytetyön aiheena oli tutkia modernin elektroniikan ja sulautettujen järjestelmien käyttöä perinteisissä lautapeleissä. Tämä opinnäytetyö on tehty osana ELLI-projektia, joka toteutettiin Satakunnan ammattikorkeakoulussa vuosina 2006 ja 2007 ja jonka rahoituksesta vastasivat Euroopan Unioni ja Nelostuote Oy.

Tutkimuksen käynnisti Nelostuote Oy ja tutkimuksen pääasiallisena tarkoituksena oli kehittää uuden tyyppinen lautapeli, jossa hyödynnettäisiin RFID-tekniikkaa pelinappuloiden paikallistamiseksi ja joka ohjaisi DVD-soitinta infrapuna-yleiskaukosäätimen avulla.

Teoreettisessa osassa perehdytään erilaisiin RFID-tekniikoihin, sulautettuihin järjestelmiin ja myös muutamiin muihin tekniikoihin, joita voitaisiin soveltaa lautapelien yhteydessä, vaikkakin pääasiallisesti tarkastellaan näistä kahta ensimmäistä.

Kokeellisessa osassa rakennettiin ja ohjelmoitiin sulautetun lautapelin prototyyppi, käyttäen hyväksi tekniikoita, joita teoreettisessa osassa käsiteltiin. Prototyyppi koostui kartonkisesta pelilaudasta, pelinappuloista, joihin sisällytettiin yksinkertainen RFID-tunniste, tavanomaisesta infrapuna-yleiskaukosäätimestä ja Alteran Nios II Development Kit -kehitysalustasta. Prototyyppi suunniteltiin ja rakennettiin konseptin todentamiseksi, ei niinkään valmistuskelpoiseksi valmiiksi laitteeksi.

Sulautettujen järjestelmien käyttö sulautettuna perinteiseen pelilautaan tuo mukanaan monta haastetta. Tällaisen laitteen on toimittava paristoilla, niiden on oltava pieniä ja halpoja valmistaa, sekä myöskin luotettavia, jotta pelielämys ei kärsisi. Myöskin näihin asioihin paneuduttiin työssä.

## USE OF EMBEDDED SYSTEMS IN TRADITIONAL BOARD GAMES

Rantamäki, Saku

Satakunnan ammattikorkeakoulu, Satakunta University of Applied Sciences

Degree Programme in Information Technology

July 2009

Ekholm, Ari

Number of pages: 39

Key words: embedded systems, board games, RFID

---

The purpose of this thesis was to explore some of the possible uses of modern electronics and embedded systems integrated in traditional board games. This thesis is one of the products of the project ELLI, which was executed at Satakunta University of Applied Sciences in 2006 and 2007 and funded by the European Union and Nelostuote Oy.

This study was commissioned by Nelostuote Oy and its primary aim was to develop a new type of board game that utilizes Radio Frequency Identification technology to locate the pieces on the board and that communicates with DVD players with a universal IR remote control.

The topics discussed in the theoretical part of the thesis include different types of RFID implementations and embedded systems and also some other technologies that may be used with board games, although the primary focus of this study is on the first two mentioned.

In the empirical part of the thesis, a prototype was built and programmed, implementing the technologies described in the theoretical part. The prototype was composed of a cardboard game board, plastic game pieces that integrated simple RFID chips, an ordinary universal IR remote control and Altera Nios II Development Kit. The prototype was designed and built as a proof of concept, rather than a device that is ready for production.

Implementing an embedded system in traditional board games brings many challenges. They need to run on battery power, be small in size and inexpensive to produce, and also be reliable and accurate so that the gaming experience would not suffer. These matters were also taken into consideration in this study.

# SISÄLLYS

JOHDANTO.....	5
INTERAKTIIVISUUDEN LISÄÄMINEN LAUTAPELEIHIN .....	7
Käyttäjän suora syöte kytkimien ja resistiivisten virtapiirien avulla .....	7
RFID (Radio Frequency Identification) pelinappuloiden paikannuksessa .....	8
Ulkoisten laitteiden ohjaus ja käyttö osana peliympäristöä.....	10
Kuva ja ääni AV-liitännän kautta.....	12
Elektronisen paperin käyttö pelilaudan pohjana .....	13
PROTOTYYPIN TOTEUTUS.....	14
Lähtökohta ja vaatimukset .....	14
Alustava suunnittelu.....	18
Haaste: hopeapastalla paperille painetut kelat .....	21
Rajoitteiden poistaminen: FPGA korvaa mikro-ohjaimen.....	22
Yhteenvedo toteutettavan prototyypin rakenteesta.....	24
Analogisen osion suunnittelu.....	25
Pelinappuloiden tunnisteet.....	26
Referenssjännitteiden tuottaminen.....	27
Mittasignaalin suodatus ja vahvistus .....	28
Mittauskelan valinta multiplekserillä.....	29
Mittauskelojen jännitteen mittaaminen.....	30
Kaukosäätimen ohjaus kahdella multiplekserillä.....	32
Digitaalisen osion suunnittelu.....	34
Mittausignaalin tuottaminen.....	35
AD-muuntimen ohjaus.....	36
Pelilaudan valmistelu ja liittäminen sekä prototyypin viimeistely .....	36
Valmiin prototyypin testaus .....	37
PROTOTYYPIN JATKOKEHITYS JA JOHTOPÄÄTÖKSET .....	38
LÄHTEET .....	39
LIITTEET	

# 1 JOHDANTO

Sulautettujen järjestelmien suosio on kasvanut valtavasti viimeisten vuosikymmenien aikana ja elektroniikan pudonnut hintataso mahdollistaakin nykyään älykkyyden lisäämisen miltei mihin tahansa kulutuslaitteeseen aina partakoneesta jääkaappiin. Lähiaikoina suurin kilpailu sulautettujen järjestelmien markkinoilla onkin ollut uusien ja markkina-arvoa omaavien sovelluskohteiden etsimisessä.

Tämä opinnäytetyö on raportti eräästä sovellusaluetta tutkineesta projektista. Projekti tutki mahdollisuutta sulauttaa elektroniikkaa ja ennen kaikkea RFID-tekniikkaa perinteiseen lautapeliin. Projektin alkuvaiheessa (vuonna 2007) ei löytynyt tietoa yhdestäkään valmiista, tälle sovellusalueelle tehdystä tuotteesta, joka kyseistä tekniikkaa hyödyntäisi. Toisaalta elektroniikan ja sen tuotantomenetelmien kehitys on vasta 2000-luvulla pystynyt vastaamaan haasteisiin, joita tällaisen lautapelin valmistus tuo mukanaan. Tällaisia haasteita tuovat mm. vaatimukset pienelle virrankulutukselle ja koolle sekä hyvälle eri johdinmateriaalien liitostekniikoille.

Elektroniikan sulauttaminen pelilautaan on kuitenkin ristiriitaista. Perinteiset pahviset pelilaudat ovat olleet pitkään videopelien rinnalla ja kun näiden kahden ominaisuuksia ja ajatusmaailmaa lähdettiin yhdistämään, oli toisinaan vaikea päättää mihin raja vedettäisiin. Tässä projektissa oli kuitenkin lähtökohtana elektroniikan toiminnallisuuden piilottaminen niin, että pelaajien ei tarvitse kiinnittää huomiota sen toimintaan. Elektroniikan tarkoitus oli vain tuoda pelaamiselle lisäarvoa, samalla kuitenkin säilyttäen lautapelien oma arvomaailma ja pelaajien yhdessä tekemisen tunnelma.

Nelostuote Oy, työn tilaaja asetti projektille kaksi tavoitetta: Elektroniikan pelilautaan sulauttamisen mahdollisuuksien tutkiminen ja interaktiivisen pelilaudan prototyypin rakentaminen. Prototyypin rakennettaessa ei ollut tarkoitus käyttää viimeisintä ja kalleinta saatavilla olevaa teknologiaa, vaan tavoitteena oli pikemminkin yksinkertainen ja toimiva, helposti sarjavalmistukseen vietävä laite. Sarjavalmistusta

ajatellen pyrittiin käyttämään sellaisia menetelmiä, että Nelostuote Oy pystyisi tekemään suurimman osan tuotannon eri vaiheista Porin toimipisteessään.

Projekti suoritettiin Satakunnan ammattikorkeakoululla, tekniikka Porin toimipisteessä, koulun sisäisenä projektina, mutta samalla kuitenkin tiiviissä yhteistyössä Nelostuote Oy:n kanssa. Projektin laajuudesta johtuen se toteutettiin ryhmässä, jossa jokaisella oli omat vastualueensa. Suunnittelija Leila Tasku vastasi projektin vetämisestä ja yhteistyöstä työn tilaajan kanssa. DI Ari Ekholm sekä TkL Tauno Perkiö avustivat projektissa koko prototyypin rakennusvaiheen ajan. Rahoituksesta vastasivat Nelostuote Oy ja EU.

Tämä opinnäytetyö on jaettu kahteen osaan, joista ensimmäinen käsittelee sovel-lusalueen yleistä tutkimusta sekä tulevaisuuden tekniikoiden kartoittamista ja toinen varsinaisen prototyypin rakentamista.

## 2 INTERAKTIIVISUUDEN LISÄÄMINEN LAUTAPELEIHIN

Suurin ja oikeastaan ainoa etu, joka sulautetulla pelilaudalla on perinteiseen verrattuna on sen interaktiivisuus: Pelaaja voi omalla toiminnallaan antaa elektroniikalle pelilaudan kautta syötteitä ja elektroniikka pystyy vastaavasti näiden syötteiden ja erilaisten elektronisten tai elektromekaanisten ratkaisujen kautta tarjoamaan pelaajalle sisältöä. Tästä syystä projektin aikainen tutkimustyö keskittyikin nimenomaan interaktiivisuuden lisäämiseen.

Seuraavissa kohdissa kuvataan erilaisia tapoja joilla käyttäjä ja älykäs pelilauta voivat keskustella, sekä on tutkittu näiden tekniikoiden hyviä ja huonoja puolia ja tuotu hieman esiin myös sarjatuotantoon vaikuttavia näkökohtia. Tässä opinnäytetyössä tutkitut tekniikat rajautuvat suuresti työn tilaajan toiveiden mukaisesti ja käsittely on myös melkoisen pintapuolista, koska suurin osa resursseista oli koko projektin aikana sidottu prototyypin rakentamiseen.

### 2.1 Käyttäjän suora syöte kytkimien ja resistiivisten virtapiirien avulla

Kytkimet ovat varmastikin yksinkertaisin väline elektroniikan kanssa vuorovaikuttamiseen. Niiden avulla käyttäjä voi suoraan ohjata elektroniikan toimintaa ennalta määrätyillä tavoilla. Kytkimien käyttö sulautetussa pelilaudassa tuo kuitenkin mukanaan tuotannollisia ja pelaamisen laatuun vaikuttavia ongelmia.

Tuotannolliseksi ongelmaksi voidaan lukea pelkkään pahviseen pelilautaan verrattuna kalliimpi tuotantoprosessi, koska mekaanisten kytkimien käyttö vaatii usein irrallisten komponenttien liittämisen pelilautaan. Jopa käytettäessä yksinkertaisia kalvokytkimiä, jouduttaisiin tekemään pahvisia pelilautoja valmistavaan tuotantolinjaan muutoksia. Lisäksi kytkimien käyttö vaikeuttaa osaltaan lautojen laadunvalvontaa, koska kytkimiä täytyisi pystyä testaamaan tuotantovaiheessa.

Interaktiivisen lautapelin käyttö saattaisi myös olla hämäävää, jos siinä olisi paljon erilaisia kytkimiä, kuten painonappeja. Silloin varsinainen lautapelimäisyys katoaisi ja pelaajat joutuisivat kiinnittämään koko ajan huomiota nappien painamiseen.

Tässä projektissa valmistetun prototyypin osalta ei haluttu lisätä varsinaiseen pelilautaan yhtään mekaanista kytkintä. Sen sijaan pelilaudan keskelle oli tarkoitus asentaa muovinen kotelo, johon tulisi mahdollisesti tulevaisuudessa myös laitteen vaativat paristot. Tämän kotelon pintaan oli tarkoitus asentaa painonappeja olennaisten toimintojen ohjaukseen: Herätteet uuden pelin aloittamiseen ja virtuaalisen nopan heittämiseen haluttiin toiminnallisista syistä painonappien taakse.

Käyttäjän suorasta vuorovaikutuksesta voidaan tosin päästä eroon käyttämällä pelilaudassa esimerkiksi magneettisia kytkimiä ja asentamalla pelinappuloihin magneetit tai vaihtoehtoisesti voidaan pelilautaan sisällyttää johtimia joita pelinappulat yhdistävät laudalla liikkeessään. Näissä ratkaisuissa on kuitenkin molemmissa omat ongelmansa. Kummassakaan ei pelilauta saa mitään tietoa siitä, mikä nappula missäkin kohdassa on, ellei käyttäjä tätä tietoa omalla toiminnallaan pelilaudalle välitä. Lisäksi molemmat ratkaisut vaatisivat myös muutoksia tuotantomenetelmissä.

## 2.2 RFID (Radio Frequency Identification) pelinappuloiden paikannuksessa

RFID:llä tarkoitetaan radiotaajuista etätunnistusta, jossa erilliseltä tunnisteelta pystytään lukemaan tunnistetiedot ilman, että tunnisteen ja lukulaitteen välillä olisi galvaanista yhteyttä. Tällaisia tunnisteita kutsutaan myös suomalaisittain nimellä saattomuisti ja niiden sovelluskohteita löytyy nykypäivänä lähes rajattomasti kaikilta teollisuuden ja arkielämän alueilta, tosin niiden käyttö lautapeleissä on harvinaista ja muutamia vuosia sitten olematonta. Nykyään ne kuitenkin mahdollistaisivat – tietysti tekniikasta riippuen – tehokkaan tavan myös pelinappuloiden tunnistamiseen pelilaudalla, eikä RFID-tekniikan soveltaminen tule välttämättä edes tuotantolinjassa kalliiksi. Usein RFID:n yhteydessä tunnisteilla tarkoitetaan nimenomaan digitaalisia tunnisteita joita on kolmen tyyppisiä: aktiivisia, passiivisia ja puoli-passiivisia.



Aktiivisissa tunnisteissa on sisäänrakennettu virtalähde, jota tunniste käyttää hyväksi lähettäessään tietoja ja myös mahdollisesti sisäänrakennetun muistin ylläpitämiseen tai muun toiminnallisuuden, kuten vaikka lämpötilan mittaamisen mahdollistamiseksi. Tämän tyyppisillä tunnisteilla saadaan pisin mahdollinen lukuetaisyys, eli etaisyys lukulaitteen ja tunnisteiden antennien välillä, joka saattaa parhaimmillaan ylittää jopa sata metriä.

Aktiiviset tunnisteet voivat olla nykyään, virtalähde mukaan lukien, pienimmillään kolikon kokoluokkaa, joten niiden käyttö lautapeleissäkin voisi olla mahdollista mm. pelinappuloiden sisään integroimalla. Aktiivinen tunniste ei kuitenkaan lautapelien yhteydessä tuo mitään etua passiiviseen tai puoli-passiiviseen tunnisteeseen verrattuna, sillä lukuetaisyydet eivät ole pitkiä ja aktiivisten tunnisteiden käyttöikä rajoittuu virtalähteen ikään, joka tosin saattaa olla useita vuosia tunnisteesta riippuen. Lisäksi aktiivisia tunnisteita voidaan käyttää tehokkaasti radiopaikannuksessa, mutta tällä hetkellä markkinoilla olevien (vuonna 2009) kohtuuhintaisten paikannustekniikoiden tarkkuus ei riitä lautapelien sovelluksiin.

Passiivinen RFID-tunniste ei sisällä virtalähdettä, vaan se kytkeytyy induktiivisesti tai kapasitiivisesti lukulaitteeseen joka tarjoaa sille käyttösähkön. Tällainen tunniste keskustelelee lukijalaitteen kanssa muuttamalla antennipiirinsä sähköisiä ominaisuuksia niin, että muutos havaitaan lukijassa, kun taas aktiiviset tunnisteet lähettävät lukijalle viestejä radiotaajuuksia käyttämällä. Passiivinen tunniste sisältää käytännössä aina vain kyseisen tunnisteiden tunnistamiseen vaaditun elektroniikan ja tunnistetietoja on lähes poikkeuksetta mahdoton jälkikäteen muuttaa. Näiden tunnisteiden etu onkin niiden yksinkertaisuudessa ja edullisuudessa, mitkä tekevätkin niistä hyvän vaihtoehdon lautapeleihin ja sarjatuotantoon.

Passiiviset tunnisteet ovat pieniä ja niiden lukuetaisyys on välillä 10 mm – 5 m. Pelilaudalta tällaiset tunnisteet on helppo paikallistaa niin, että pelilaudalle tehdään useita tunnisteita joiden avulla tunnisteet kytkeytyvät lukijalaitteeseen joko induktiivisesti tai kapasitiivisesti. Jos käytetään induktiivisia tunnisteita, voidaan pelilaudan grafiikan alle piilottaa esimerkiksi piirilevy jossa on jokaisen tunnisteiden

kohdalla kela. Tällaisen sovelluksen rakentamisessa säästettäisiin myös kustannuksia, koska voitaisiin käyttää vain yhtä lukijaa, joka kytkettäisiin elektronisesti yksitel- len jokaiseen kelaan.

Puoli-passiivinen tunnistus on yhdistelmä aktiivisen ja passiivisen tunnistuksen raken- teesta. Se sisältää virtalähteen, mutta virtalähdettä käytetään vain sisäisen toiminnan, kuten muistin, tarpeisiin. Energian lukijalaitteen kanssa kommunikointiin ottaa puoli- passiivinen tunnistus samalla tavalla lukijalaitteelta, kuin passiivinenkin tunnistus. Kuitenkin puoli-passiivisen tunnistuksen suurin mahdollinen lukuetaisyys on passiiv- ista parempi, koska puoli-passiivinen tunnistus ei tarvitse ulkoista energiaa omaan sisäiseen toimintaansa. Tunnistusten käytössä pelilautasovelluksessa on käyttökohteita tunnistusten sisäiselle muistille tai muulle mahdolliselle toimin- nallisuudelle kuitenkin hyvin rajallisesti, varsinkin jos otetaan huomioon puoli- passiivistenkin tunnistusten virtalähteen rajoitettu ikä. /1, 2/

Eräs mahdollinen tapa toteuttaa pelinappuloiden paikallistaminen on yksinkertais- tettu versio passiivisista tunnistuksista. Jos on tarve erotella vain muutamia pelinappu- loita toisistaan, voidaan passiivinen RFID-tunnistus korvata yksinkertaisella resonans- sipiirillä. Tällöin lukijalaitteella voidaan lähettää sini-muotoista signaalia mittake- lalle, johon resonanssipiiri on induktiivisesti kytkeytynyt, ja mittakelan impedans- sista voidaan tunnistaa, mikä pelinappula sen lähellä milläkin hetkellä on. Tätä tek- niikkaa sovellettiin tässä opinnäytetyössä rakennetun prototyypin yhteydessä. Tek- niikka oli yksinkertainen toteuttaa ja tarjosi kaikki tarvittavat ominaisuudet muuta- mien pelinappuloiden tunnistamiseen pelilaudalta.

### 2.3 Ulkoisten laitteiden ohjaus ja käyttö osana peliympäristöä

Pelilaudan älykkyys ei tuo juurikaan mitään lisäarvoa pelaajille, jos pelilauta ottaa vain vastaan syötteitä, mutta ei tuota mitään pelaajille. Jos pelilauta tietää koko ajan pelin tilanteen, käyttäen hyväksi kytkimiltä tai muilta tunnistusmenetelmiltä tulevia tietoja, voi se yhtä hyvin myös ohjata pelin kulkua ja tarjota sisältöä multimedian muodossa, ja tämä varmasti tuo suurimman hyödyn sulautettujen järjestelmien

käytöstä. Sen sijaan, että lautapelin elektroniikka tarjoaisi suoraan sisältöä pelaajille, on varmasti paljon kustannustehokkaampaa käyttää jo miltei joka taloudessa ole-massa olevaa tekniikkaa hyväksi. Tästä syntyi idea kaukosäätimen integroimiseksi ja DVD-soittimen ohjaamiseksi. Nelostuotte Oy:llä oli jo valmiiksi vahva osaaminen DVD-pelien valmistamisesta ja koska DVD oli jo valmiiksi tuttu media, sen käyttö sisällön tarjoamiseen pelaajille oli luonnollinen valinta.

Ulkoisten laitteiden, kuten tässä tapauksessa DVD-soittimen infrapunalla ohjaaminen ei ole ongelmaton ja tässäkin yhteydessä pitää ottaa huomioon ainakin seuraavat asiat: Kommunikointiin käytetyn siirtotien häiriöt, laitteistojen yhteensopivuus ja yhteyden luonti niiden välille sekä virhetilanteista toipuminen.

Infrapuna on herkkä häiriöille ja esteille. Tämä tulisi ottaa huomioon, varsinkin jos sulautettu järjestelmä rakennetaan kiinteäksi osaksi pelilautaa. Mikäli näin tehdään, on pelaajien aina otettava huomioon pelilaudan sijoitus ja taattava esteetön optinen yhteys pelilaudan infrapuna-lähettimeltä DVD-soittimen vastaanottimelle, mikä voi rajoittaa peliympäristöä ja latistaa pelitunnelmaa.

Laitteistojen yhteensopivuudella on todella paljon merkitystä ja siksi ainakin kauppalisen älykkään pelilaudan tulisi tukea miltei kaikkia mahdollisia DVD-soittimia. Toinen ongelma yhteensopivuuteen liittyen saattaa tulla eri laitteiden, kuten DVD-soittimen ja digiboksin välille, jos nämä käyttävät joillekin toiminnoilleen samaa infrapunakoodia. Tätä ei kuitenkaan voitaisi lukea suoranaisesti pelilaudan ongelmaksi ja sen merkitys muutenkin on pienempi. Sen sijaan suurempi ongelma on asennuksen helpoksi tekeminen. Jotkut yleiskaukosäätimet osaavat nauhoittaa signaaleja toisesta kaukosäätimestä, mutta tämä tekniikka ei ole kovinkaan yksinkertainen toteuttaa. Toisaalta ei ole työmäärällisesti tai kustannuksellisesti kannattavaa alkaa tekemään kaikki DVD-soittimet kattavaa koodikirjastoa, varsinkaan jos tarkoituksena on tehdä jotakin muuta kuin DVD-yleiskaukosäädin. Siksi tämän työn toisessa osassa kuvatus prototyypin rakennusvaiheessa käytettiin valmista yleiskaukosäädintä, joka sisälsi suurimman osan markkinoilla olevien soitinten ohjauskoodeista.

Virhetilanteista on pystyttävä toipumaan. Tämä toiminnallisuus on välttämätöntä, mikäli DVD-soittimessa esiintyy häiriöitä tai yhteys pelilaudan ja soittimen välillä on katkennut juuri ohjaukoodin lähettämisen aikana. Varsinaisen pelilaudan häiriöihin ja väärin toimimiseen ei voida vaikuttaa, mutta jos pelilaudan elektroniikka tietää tarkkaan pelin tilanteen, ei ole mitenkään mahdotonta ohjata DVD-soitinta takaisin ajankohtaiseen tilanteeseen. Tämä ei kuitenkaan ole mahdollista mikäli pelin loogikka on monimutkainen ja osa siitä on toteutettu DVD:n sisäisenä ohjelmointina.

Ongelmistaan huolimatta tällainen on helppo ja edullinen ratkaisu toteuttaa.

#### 2.4 Kuva ja ääni AV-liitännän kautta

Jos älykäs pelilauta pystyisi suoraan välittämään kuva- ja äänisignaalit esimerkiksi televisiolle, ei välissä tarvittaisi ollenkaan kaukosäädintä tai DVD-soitinta ja samalla pystyttäisiin vaikuttamaan pelin sisältöön paljon tehokkaammin. Ongelmatonta tällaisen AV-liitännän (AV, Audio, Video) rakentaminen ei kuitenkaan ole. Multimediantallennus vie paljon tilaa ja koska optisen tallennusvälineen käyttö osana lautapeliä ei ole välttämättä kovin kustannustehokas ratkaisu, jää vaihtoehdoksi lähinnä Flashmuistin käyttö. Jos peliin on tarkoitus liittää paljon sisältöä ja mahdollisesti myöhemmässä vaiheessa tehdä samalla periaatteellaa uusi peli käyttäen taas eri sisältöä, lienee järkevintä, varsinkin, kun pienen kapasiteetin muistipiirit ovat paljon edullisempia, kuin suuren kapasiteetin.

Eräs mahdollinen vaihtoehto olisi käyttää standardin mukaisia USB-muistitikkuja, joille pelaajat voisivat siirtää peliin kuuluvan sisällön tietokoneen avulla. Sisältö voitaisiin toimittaa esimerkiksi CD-levyllä pelin mukana. Tällöin itse pelilauta pystyttäisiin myymään edullisemmin ja USB-muistitikun hankinta jäisi pelaajan vastuulle ja hän voisi käyttää sitä myös muihin tarkoituksiin.

Mikäli pelilautaan integroitaisiin AV-liitäntä, tarkoittaisi se myös sitä, että pelilaudan elektroniikan vastuulle jäisi kuva- ja äänimateriaalin mahdollinen pakkauksen tai

koodauksen purkaminen, mikä puolestaan lisäisi suuresti elektroniikan virrankulutusta.

Mahdollisestikin paras ratkaisu tällaiseen ratkaisuun olisi pelilaudan toiminnallisuuden jakaminen kahteen osaan: Varsinainen pelilauta toimisi tyhjänä laitteena ja viestittäisi televisioon kytketylle erilliselle laitteelle vain pelilaudan nappuloiden tilatietoja. Televisioon kytketty laite saisi virtansa sähköverkosta, lataisi pelin kulkuun liittyvän logiikan ja multimediamateriaalin USB-muistitikulta ja ohjaisi tätä mediaa televisiolle pelin edistyessä. Tämä yksikkö ja pelilauta keskustelisivat radioaajuuksien avulla, jotka ovat huomattavasti luotettavampi siirtotie, kuin mitä on infrapuna. Vielä parempi olisi, jos kommunikointiin voitaisiin käyttää jotakin valmista ja laajennettavaa protokollaa, kuten Bluetoothia. Bluetoothin etuna olisi mm. se, että elektronisen paperin hinnan pudotessa, televisioon liitetystä keskusyksiköltä voitaisiin ladata langattomasti koko pelilaudan ulkoasu suoraan pelilaudalle.

Näillä ratkaisuilla voitaisiin alkuinvestoinnin hinta pitää kohtuullisena, käyttö yksinkertaisena ja uusien pelien hankkiminen helppona ja edullisena.

## 2.5 Elektronisen paperin käyttö pelilaudan pohjana

Elektroninen paperi ei ole kovinkaan uusi keksintö, sillä sen ensimmäinen sovellus tehtiin jo 1970-luvulla. Pitkästä iästään huolimatta ei elektroninen paperi, tai lyhyemmin e-paperi, ole saanut kovinkaan paljoa julkisuutta, vaikka sitä on käytetty lehtien ja kirjojen jakeluun, sekä näyttömateriaalina mm. kelloissa ja puhelimissa. Elektroninen paperi on jo tällä hetkellä edullisempaa valmistaa, kuin ovat LCD-tekniikkaa käyttävät tietokoneen näytöt. Internetissä liittyy huhuja, että e-paperia hyödyntäviä kannettavia tietokoneitakin saattaa olla lähiaikoina tulossa, joten sen hinta tulevaisuudessa saattaa mahdollistaa sen käytön myös lautapeleissä. /3, 4/

Elektronisen paperin ominaisuuksia piti tämän projektin yhteydessä tutkia tarkemmin, mutta ajanpuutteen ja prototyypin rakentamiseen kuluneiden resurssien vuoksi tästä suunnitelmasta jouduttiin kuitenkin luopumaan.

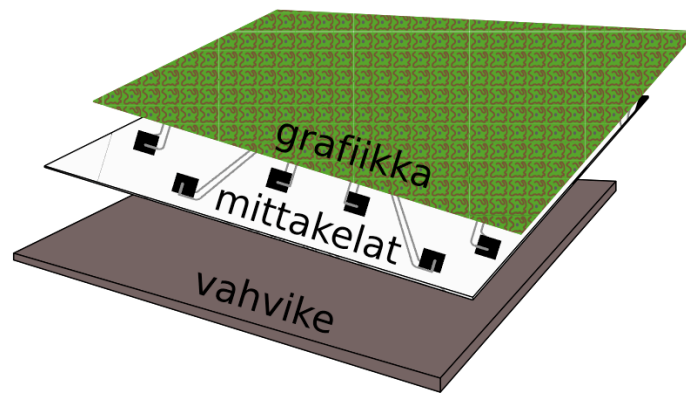
### 3 PROTOTYYPIN TOTEUTUS

#### 3.1 Lähtökohta ja vaatimukset

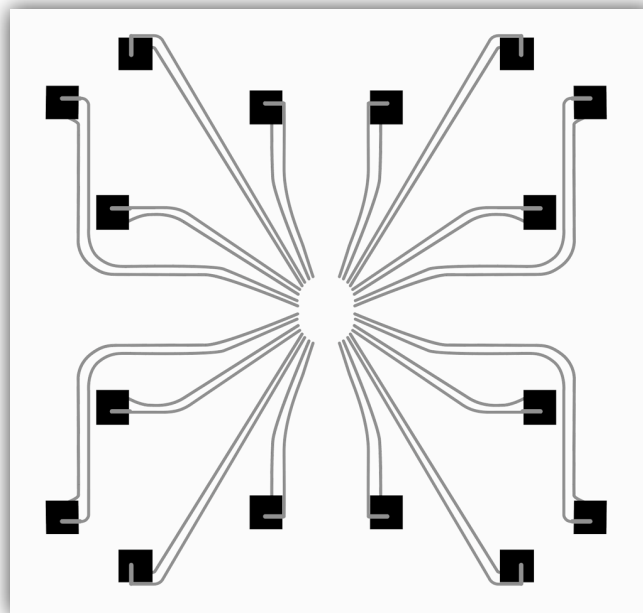
Prototyypin rakentamisen lähtökohtana oli Nelostuote Oy:n aiempi tutkimus. Heidän tutkimusryhmänsä oli selvittänyt pääasiassa erilaisten RFID-pohjaisten tunnistustekniikoiden käyttömahdollisuuksia, sekä erilaisia johdinpintojen valmistusprosesseja. Satakunnan ammattikorkeakoululla prototyypin suunnittelu aloitettiin pohjautuen tähän aiempaan tutkimukseen. Tavoitteena oli käyttää prototyypin valmistuksessa mahdollisimman paljon sellaisia ratkaisuita, joita Nelostuote Oy voisi myöhemmin käyttää Porin tuotantolinjallaan myös lopullisten, kaupallisten tuotteiden valmistuksessa.

Prototyypin toimintaperiaate ja vaatimukset olivat selvät alusta lähtien: Pelilaudan tulisi tunnistaa pelinappulat langattomasti, sekä ohjata DVD-soitinta näyttämään pelaajille pelin sisältöön liittyvää videomateriaalia. DVD-soittimen ohjaukseen päätettiin käyttää valmista, kaupasta ostettua yleiskaukosäädintä. Näin prototyyppiä pystyttäisiin esittelyvaiheessa käyttämään useamman, kuin yhden erilaisen soittimen kanssa. Toisaalta näin säästettiin myös aikaa joka olisi mennyt infrapunalähttimen rakentamiseen ja DVD-soitinten ohjauskoodien selvittämiseen.

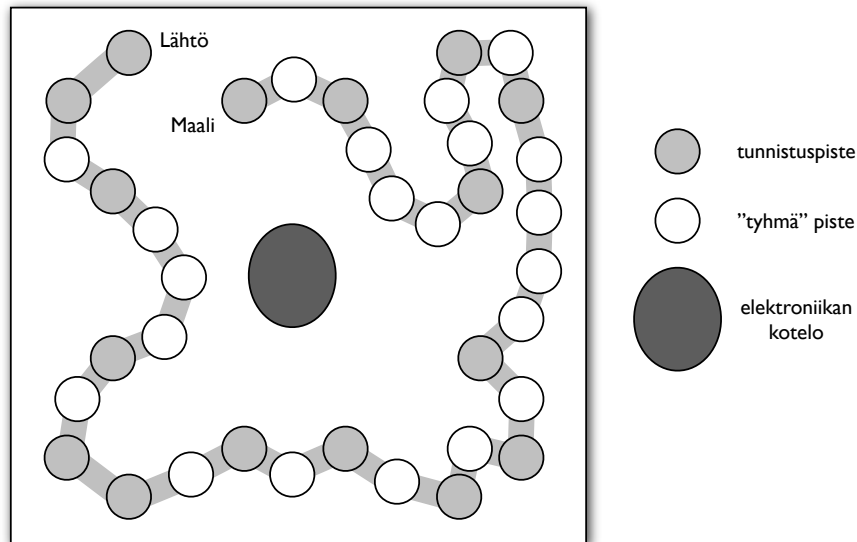
Työn tilaajan toivomuksesta, pelilaudan toteutuksessa käytettiin seripainossa paperille painettua johdintekniikkaa. Pelilauta koostui kolmesta yhteen liimatusta kerroksesta (Kuva 1). Alimmaisena oli paksusta kartongista tehty vahvike. Tämän päälle liimattiin johdinkuviot – eli tässä tapauksessa pelinappuloiden tunnistamisesta vastaavat mittauskelat – sisältä paperiarkki. Johtimet painettiin seripainossa paperiarkille tätä ennen (Kuva 2). Päälimmäiseksi liimattiin vielä yksi paperiarkki, joka sisälsi peliin liittyvän grafiikan (Kuva 3). Mittauskelojen tekeminen vaati kolme erillistä painokertaa: Ensimmäisellä kerralla painettiin johtimet, toisella kerralla eristemateriaali mittauskelojen päälle ja viimeisellä kerralla mittauskelat yhdistettiin kokonaisuksi lyhyellä johtimella (Kuva 4).



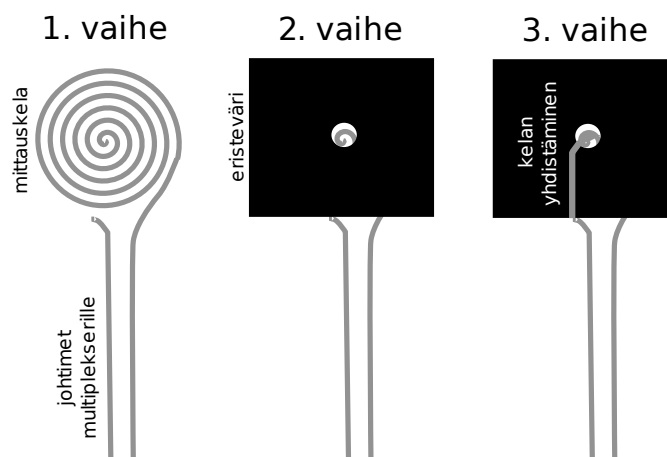
*Kuva 1: Pelilaudan rakenne*



*Kuva 2: Hopeapastalla painettujen mittakelojen asettelu*



Kuva 3: Pelaajalle näkyvä osa yksinkertaistettuna



Kuva 4: Johdinkelojen valmistus paperille seripainotekniikalla.

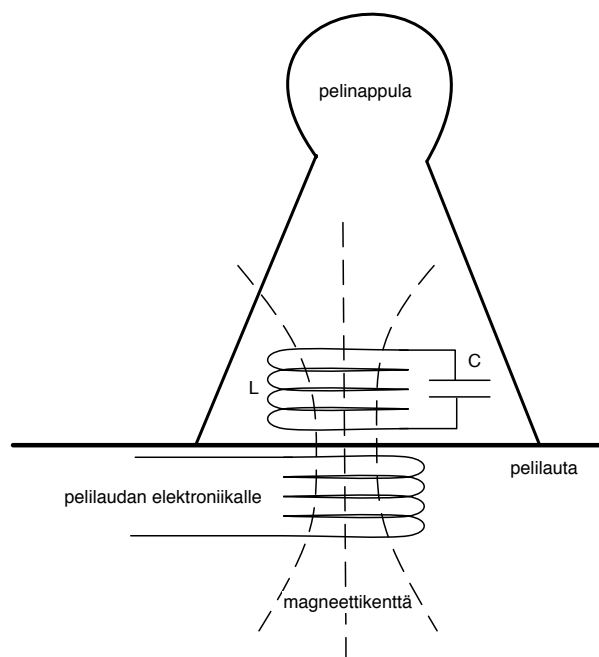
Pelilaudan älykkyydestä vastaava elektroniikka oli alunperin tarkoitus koteloida prototyypissä muovikoteloon pelilaudan keskelle (Kuva 3), mutta tiukka aikataulu ei mahdollistanut kaikkien toimintojen integroimista yhdelle piirilevyille. Siten elektroniikkaa ei saatu riittävän pieneen tilaan. Näin päädyttiin ratkaisuun, että esittelytilanteessa pelilautaa nostettiin alustasta ja erillisillä piirilevyillä oleva elektroniikka piilotettiin pelilaudan alle.

Vaikka prototyypin toteutusvaiheessa pohdittiinkin digitaalisen RFID-tekniikan käyttöä yhtenä vaihtoehtona, otettiin lopulta kuitenkin käyttöön yksinkertaisempi



ratkaisu, jossa pelinappuloiden tunnistus tapahtui niiden pohjaan asennettujen, yksinkertaisten LC-sarjakytkeytyjen resonanssipiirien avulla. Näin myös pelilautaan integroidun elektroniikan suunnittelua saatiin helpotettua ja oli helpompi pysyä kiinni tiukassa aikataulussa. Lisäksi Nelostuote Oy:n aiempi tutkimus tuki juuri tämänkaltaista ratkaisua.

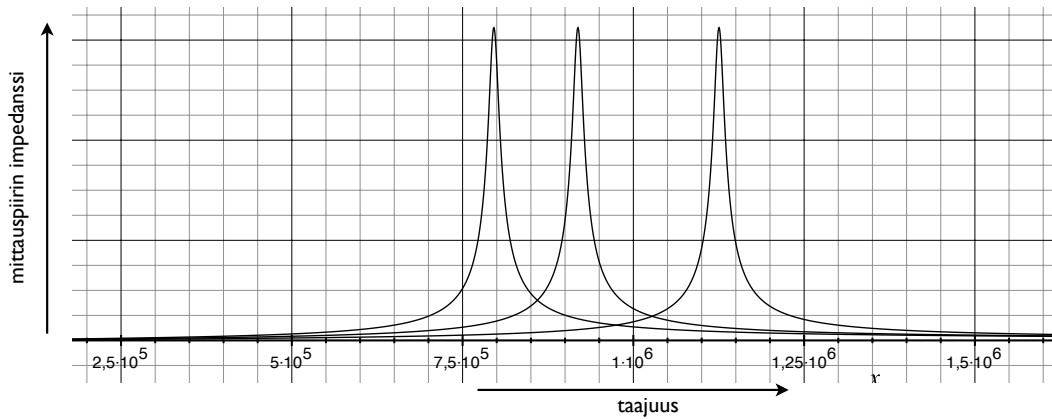
Pelinappuloihin sisällytetyn LC-kytkennän kela toimi paitsi osana resonanssipiiriä, mutta lisäksi se kytkeytyi induktiivisesti pelilaudalle sijoitettuihin keloihin – aina niiden päällä ollessaan (Kuva 5).



*Kuva 5: Pelinappulan rakenne*

Jokaisen pelinappulan LC-piirin resonanssitaajuus oli erilainen (Kuva 6). Tämä saatiin aikaan siten, että kaikissa pelinappuloissa käytettiin samanlaista, piirilevyllä painettua kela, mutta kapasitanssia vaihdettiin. Pelinappuloiden tunnistaminen tapahtui eri nappuloille ominaisen resonanssitaajuuden avulla. Pelilautaan integroitu elektroniikka kytkeytyi yhteen pelilaudalla olevaan mittauskelaan kerrallaan ja lähetti tälle tiettyjä, eritaajuisia sini-muotoisia signaaleita. Jos kyseisen mittauskelan päällä oli pelinappula, jonka resonanssitaajuus vastasi lähetetyn signaalin taajuutta, oli tämä mahdollista havaita mittauskelan piirin impedanssin kasvuna. Näin signaalin taajuutta ja kela vaihtamalla, sekä jännitettä mittaamalla, saatiin luotua jokaiselle

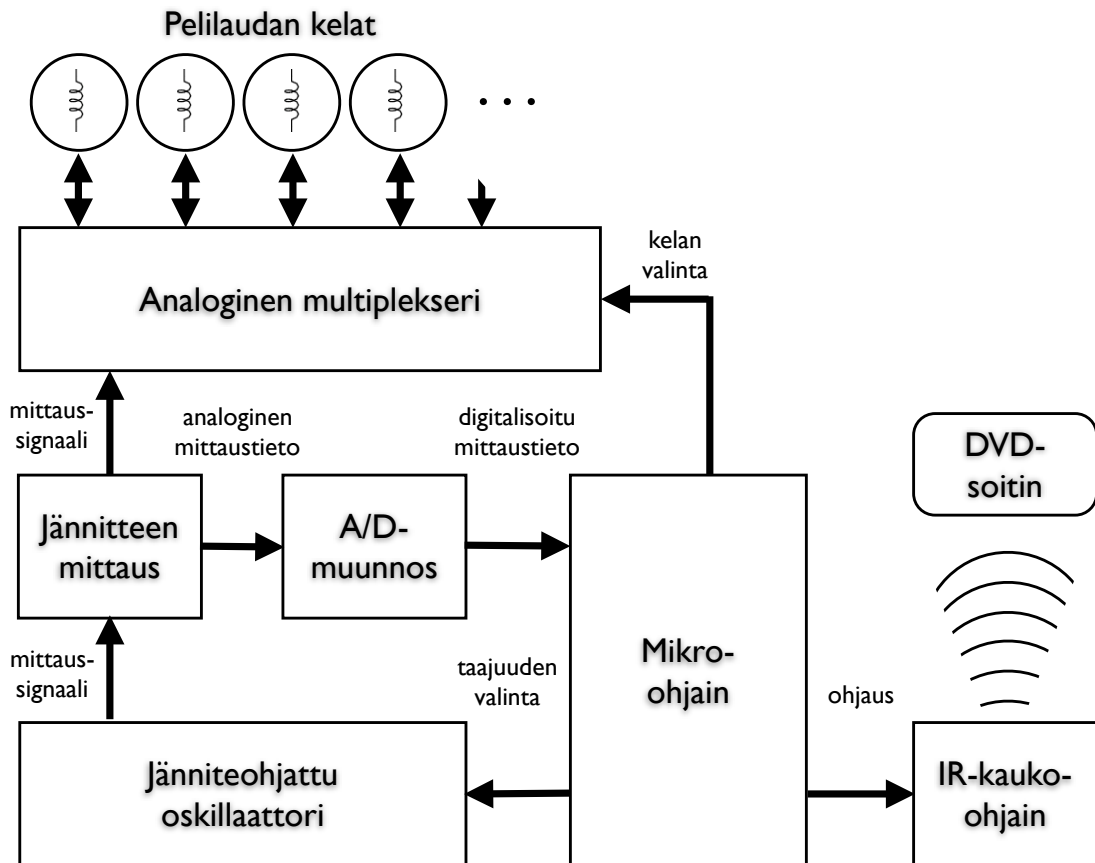
pelilautaan integroidulle mittauskelalle taajuusvastetaulukko. Tällaisesta taulukosta voitiin ohjelmallisesti päätellä, mikä pelinappula kulloinkin mittakelan päällä oli.



*Kuva 6: Kapasitanssin muuttamisen vaikutus resonanssitaajuuteen*

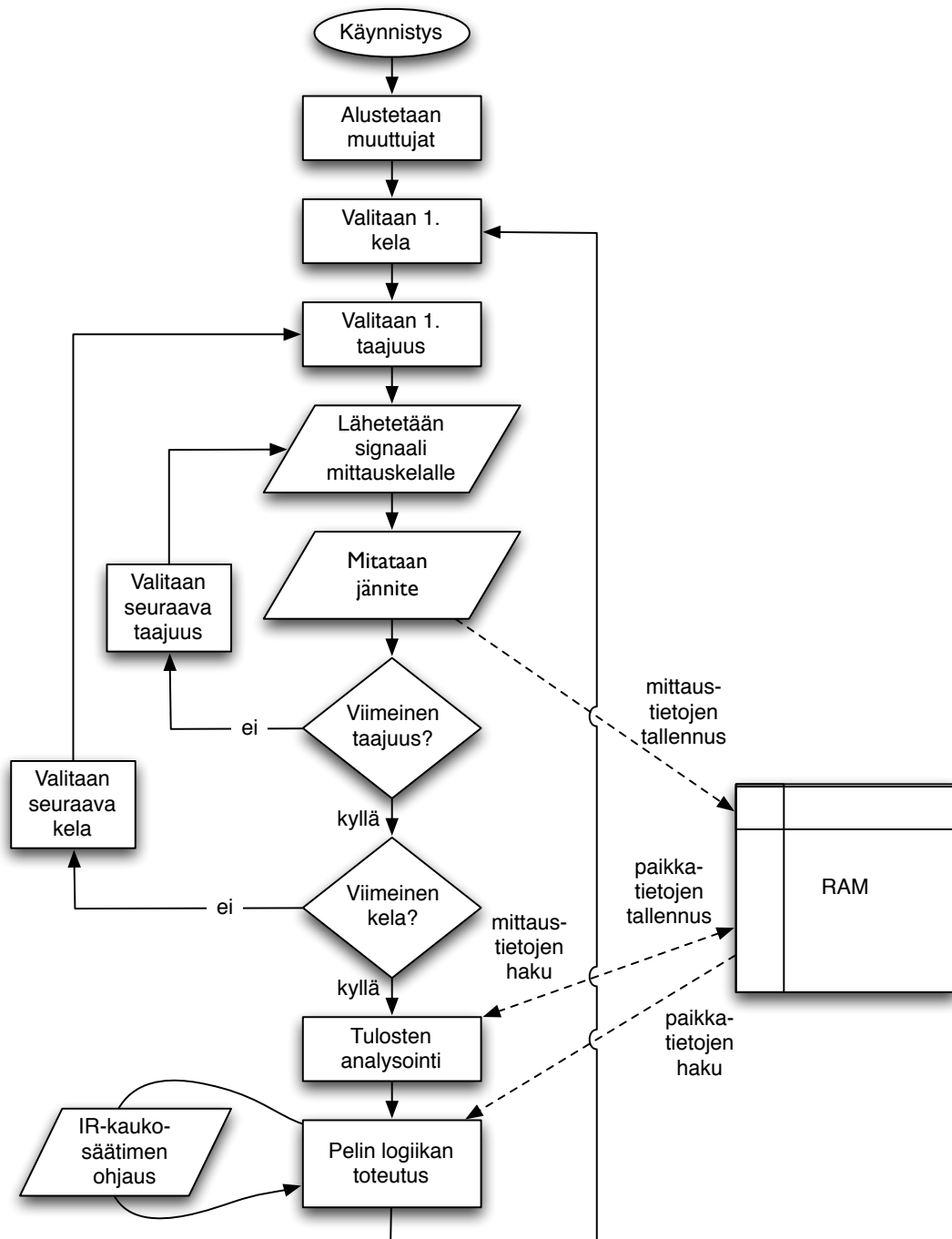
### 3.2 Alustava suunnittelu

Prototyypin toteutuksen suunnittelu aloitettiin periaatteellisella tasolla ja ennen kaikkea tekniikoiden rajoitteiden selvittämällä. Elektroniikan suunnittelua helpotettiin kokonaisuuden pilkkomisella pienempiin ja toisistaan näennäisesti riippumattomiin osiin. Elektroniikka jaettiin erilaisten toimintojen ja tarkoituksen mukaan toiminnallisiin lohkoihin ja näin saatiinkin ensimmäinen kokonaisuutta kuvaava lohkoavaio (Kuva 7).



Kuva 7: Pelilaudan elektronikan lohko-kaavio

Mikrokontrolleriin tehtävän ohjelman osalta laadittiin lisäksi suuntaa antava vuo-kaavio, jotta saataisiin hieman näkökulmaa tulevan ohjelman mahdollisesta toimin-nasta ja ennen kaikkea sen raudalle asettamista vaatimuksista.



Kuva 8: Mikro-ohjaimen toiminnan esitys vuokaaviona

Vuokaavio (Kuva 8) kuvaa mikro-ohjaimen sisäistä toimintaa ja antaa kuvan sen toimintaperiaatteesta. Lopullisen prototyypin ohjelma toteutettiin tämän kaavion periaatteen mukaisesti, mutta tässä opinnäytetyössä ei kuitenkaan käsitellä ohjelman toimintaa tämän syvällisemmin. Sulautetun ohjelman toteutuksesta vastasi lehtori Ari Ekholm.

Vaikka ratkaisumalli olisi varmastikin ollut toimiva, poikkesi valmiin prototyypin rakenne jonkin verran tästä alkuperäisestä suunnitelmasta. Suurin muutos oli kuitenkin mikro-ohjaimen korvaaminen FPGA-piirillä.

### 3.3 Haaste: hopeapastalla paperille painetut kelat

Hopeapasta ja kelojen valmistaminen painotekniikan avulla toivat varmastikin suurimmat rajoitteet prototyypin suunniteltaessa ja rakennettaessa. Jo pelkästään valmistusprosessi seripainolla oli tarkkaa, virheherkkää ja kallista. Pasta pysyi kuivuttuaan kasassa siihen sisällytetyn liima-aineen avulla. Tämä liima-aine ei ollut kuitenkaan kovinkaan kimmoisaa ja painetut johtimet olivat herkkiä vaurioille, varsinkin ennen kuin vahvike oli liimattu johdinkartongin taakse.

Hopea on itsessään hyvä johde, mutta pastaseokseen kuuluvat muut aineet – kuten epoksipohjainen liima – heikensivät painettujen johtimien johtavuutta huomattavasti. Huono johtavuus toi mukanaan ristiriidan: Mitä suurempi induktanssi mittauskeloihin oltaisiin saatu, sitä tarkempaa pelinappuloiden tunnistaminen olisi ollut. Suuren induktanssin saamiseksi oltaisiin tarvittu suuri kierrosmäärä mittauskelaan, mutta samalla tämä olisi lisännyt mittauskelan resistanssia. Resistanssi kun olisi kasvanut johtimien pituuden kasvaessa ja poikkipinta-alan pienentyessä ja täten pelinappuloiden tunnistaminen olisi vaikeutunut. /5/

Resistanssin kasvaminen mittauspiirissä huononsi kytkennän Q-arvoa ja teki näin erilaisten pelinappuloiden tunnistamisen vaikeaksi. Toinen sähköinen ongelma oli, että pitkät, noin 40 cm mittaiset, rinnakkain painetut mittauskelojen johtimet toivat kytkentään hajakapasitanssia. Hajakapasitanssin olemassaolosta seurasi, että pelilaudalla itsellään oli myös oma resonanssitaajuutensa. Tämä piti ottaa huomioon, kun suunniteltiin käytettäviä mittaussignaaleiden taajuuksia.

Nimenomaan huonon Q-arvon vuoksi, hylättiin lopullisesti idea digitaalisen RFID-tekniikan käytöstä. Painettujen kelojen kautta ei oltaisi pystytty siirtämään tarpeeksi

energiaa passiivisille RFID-tunnisteille, joita pidettiin hintansa ja yksinkertaisuutensa vuoksi ainoana vaihtoehtona edellä kuvatulle LC-kytkennälle.

Hopeapastalla painetut kelat tuottivat vielä yhden ongelman, joka oli eri mittakelojen välinen toleranssi, erityisesti resistanssin osalta. Lähempänä pelilaudan keskipistettä saattoi mittakelapiirin vastus olla noin  $50\Omega$ , kun vastaava arvo kauimmaisilla piireillä saattoi olla yli  $100\Omega$ . Prototyypillä suunniteltaessa, päätettiin tämä ongelma kiertää niin, että eri keloilta mitattuihin arvoihin tehtiin ohjelmallinen korjaus.

### 3.4 Rajoitteiden poistaminen: FPGA korvaa mikro-ohjaimen

Merkittävin yksittäinen päätös elektroniikan suunnittelussa oli mikro-ohjaimen valinta. Sitä valittaessa olivat tärkeässä osassa mm. seuraavat seikat:

- **Nopeus:** Mikro-ohjaimen tulisi pystyä skannaamaan kaikki pelilaudalla olevat mittauskelat, käyttäen kaikkia taajuuksia ja skannaus pitäisi tehdä useita kertoja sekunissa.
- **Luotettavuus:** Pienikin virhe mikro-ohjaimen tai sen ohjelman toiminnassa voi pilata hetkessä pitkäänkin kestäneen pelin. Luotettavuutta pyrittiin lisäämään käyttämällä hyvin tuettuja laitteita ja yksinkertaisia ratkaisuita.
- **Laajennettavuus:** Prototyypille nähtiin hyvät jatkokehitysmahdollisuudet ja niitä ei haluttu rajoittaa huonoista valinnoista johtuvilla ratkaisuille. Tässä yhteydessä liitännäväylojen valikoima tuli tärkeäksi tekijäksi.
- **Integroidut toiminnot:** Jos mikro-ohjain olisi sisältänyt AD-muuntimen, ei sitä olisi tarvittu erillisenä komponenttina piirilevyllä. Mahdollisuutta käyttää sisäänrakennettuja DA-muuntimia mittaussignaalien tuottamiseksi myös mietittiin, mutta rakennusvaiheessa (kesällä 2007), ei tarpeeksi nopeita DA-muuntimia ollut vielä tarpeeksi edullisesti saatavilla.
- **Hyvät kehitystyökalut ja simulaattorit** olisivat nopeuttaneet ohjelman kehitysprosessia ja täten prototyypin valmistumista. Lisäksi toivottiin, että omien työkalujen rakentaminen mm. pelilogiikan suunnittelemiseen olisi ollut helppoa. Näin pelin logiikan muuttaminen olisi onnistunut ilman, että olisi täytyntä käsitellä suoraan elektroniikkaa ohjaavaa ohjelmakoodia.

- **Pieni virrankulutus:** Koska tarkoituksena oli rakentaa paristoilla toimiva sulautettu järjestelmä, yritettiin virrankulutus pitää myöskin pienenä. Tätä ei kuitenkaan nähty ratkaisevana tekijänä, koska suunnitteilla oli kuitenkin prototyyppi ja aikataulussa ei ollut muutenkaan tilaa suunnitelman hienosäätöön.

Mikro-ohjaimen riittämättömyyttä määriteltyyn tarkoitukseensa ruvettiin kuitenkin epäilemään, kun pelilaudan aiheuttamat todelliset rajoitukset tulivat selville. Suurimpana ongelmana nähtiin eri taajuuksien mittaussignaalien tuottaminen. Koska mikro-ohjainten DA-muuntimet ja samoin digitaaliset lähdöt eivät olleet riittävän nopeita, alettiin miettimään FPGA-piirien käyttöä. Samalla todettiin, että FPGA-piirit mahdollistavat paremman laajennettavuuden ja mahdollisuuden useisiin erilaisiin toteutustapoihin. Lisäksi useilla FPGA-valmistajilla, kuten Alteralla ja Xilinx:llä on toimivat, monipuoliset ja ilmaiset kehitystyökalut omille tuotteilleen.

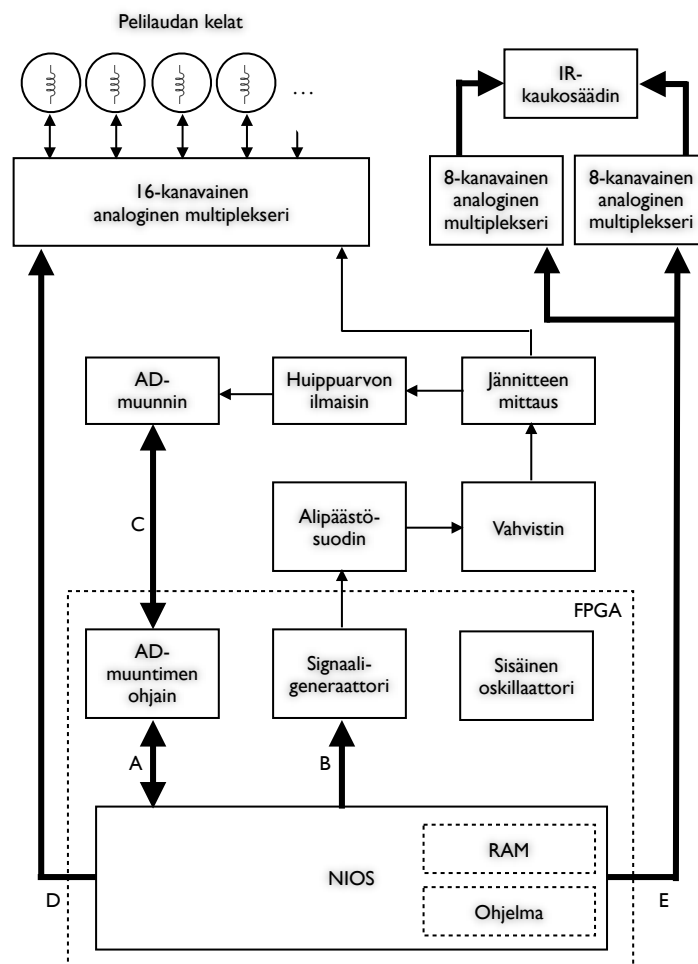
Kun päädyttiin käyttämään FPGA-piiriä, päätettiin silti käyttää yleiskäyttöistä, mielellään avoimen lähdekoodin lisenssillä julkaistua prosessoria piirille sulautettuna. Näin päädyttiin ratkaisuun, että prototyypissä käytettäisiin sivustolta [www.opencores.org](http://www.opencores.org) löytynyttä T51-sarjan kanssa yhteensopivaa, avointa prosessorin suunnitelmaa FPGA:lle sulautettuna. T51-sarjan prosessoreille löytyi jo tuolloin (keväällä 2009) paljon käyttökelpoisia kehitystyökaluja ja simulaattoreita. Ajatuksena oli, että jotakin avointa simulaattoria laajentamalla oltaisiin saatu rakennettua koko pelilautaa matkiva versio. Tällaisella simulaattorilla oltaisiin uuden pelin sisältöä ja logiikkaa suunniteltaessa pystytty käytännössä pelaamaan koko peli läpi ilman, että yhtään ohjelmaa olisi tarvinnut siirtää varsinaiselle sulautetulle pelilaudalle.

Tästä suunnitelmasta kuitenkin luovuttiin, koska sen toteuttamiseen vaadittu aika ei olisi mitenkään mahtunut projektin aikatauluun. Valmis prototyyppi sisälsi FPGA-piirin, mutta prosessorina toimi Alteran kehittämä sulautettu Nios-prosessori. Nios ei myöskään vaadi erillistä rautaa, vaan sekin sulautetaan FPGA-piirille. Hyvänä puoleena sen käytössä on, että se on hyvin tuettu Alteran kehitysympäristöissä ja erilaisilla FPGA-piireillä, sekä lisäksi sen ohjelmointi tapahtuu yksinkertaisesti C-kielillä.

Nios on myös helposti laajennettavissa: Siinä missä perinteisestä rauta-pohjaisesta mikro-ohjaimesta loppuu IO-väylät kesken, voi Nios:iin liittää kehitysympäristössä uuden IO-väylän käytännössä muutamalla hiiren klikkauksella. Kehitysympäristö liittää Nios:n ohjelmalliseen rautaa uuden IO-väylän ja samalla tekee kyseistä väylää koskevat määrytykset avoimena olevan ohjelmaprojektin kirjastotiedostoihin. Näin ohjelman kehittäjän ei tarvitse käyttää koskaan kyseiselle suunnitelmalle määriteltyjä IO-väylöjen osoitteita ja samalla C-kielellä kirjoitetut ohjelmat säilyttävät siirrettävyytensä ja selkeytensä.

Nios-prosessorin käyttöönoton myötä kuitenkin luovuttiin mahdollisuudesta omien simulointityökalujen luomiseen, ainakin tämän projektin osalta.

### 3.5 Yhteenveto toteutettavan prototyypin rakenteesta



Kuva 9: Prototyypin lopullinen lohkokaavio. Digitaaliset väylät on merkitty kirjaimin.



Kuvassa 9 on kuvattu valmiin prototyypin rakenne yhden lohkokaaavion muodossa. Digitaalisen ja analogisen osan rajapinta yritettiin pitää mahdollisimman yksinkertaisena. Tämä johtui siitä, että ajan säästämiseksi suunnittelu jaettiin heti alussa kahteen osaan: analogisen osan suunnittelu ja digitaalisen osan suunnittelu. Molempien osien suunnittelu aloitettiin samaan aikaan ja oli ensiarvoisen tärkeää, että osat saatiin sovitettua lopulta yhteen.

Ainoa sarjavyöly valmiissa prototyypissä oli FPGA:n ja AD-muuntimen välillä oleva SPI-väylä (Kuva 9, väylä C). Kaikki muut väylät (väylät A, B, D, E) olivat rinnakkaisia. On hyvä huomata, että Nios-proessori itsessään tukee suoraan SPI-väylää, mutta tässä ratkaisussa päädyttiin ratkaisuun, jossa Nios keskusteli AD-muuntimen kanssa erillisen ohjaimen kautta. Päätös T51-proessorin hylkäämisestä ja Nios:n käyttöönotosta tuli vasta projektin viime metreillä ja vaikka siirros oli melko vaivaton, oli Nios:n SPI-väylä todella huonosti dokumentoitu ja haasteellinen ottaa käyttöön. T51:llä toteutetusta ratkaisusta saatiin kuitenkin helposti kopioitua VHDL-kielellä kuvattu ohjainpiiri, joka kirjoitettiin juuri AD-muuntimen kanssa keskustelemista varten. Ohjain liittyy Nios-prosessoriin rinnakkaisväylien kautta. /6/

Toinen FPGA:n sisälle sulautettu, Nios:sta irrallinen osa on järjestelmäkellon jakaja. Jakajan tarkoitus on tuottaa analogiselle elektroniikalle pelinappuloiden resonanssitaajuuksia. Nios ohjaa tätä jakajaa yhden 8-bittisen rinnakkaisväylän kautta, asettaen haluamansa jakajan väylälle. Jakaja käynnistyy automaattisesti havaittuaan väylällä nollasta poikkeavan arvon.

### 3.6 Analogisen osion suunnittelu

Analogisen elektroniikan käyttö prototyyppiä rakennettaessa pyrittiin pitämään mahdollisimman vähäisenä, koska mitä enemmän toimintoja FPGA-piiriin pystyttäisiin sisällyttämään, sitä joustavampi prototyyppi olisi suunnitella. Analogisen osan tehtäviksi jäivät FPGA:lta tulevan mittasignaalin suodatus, vahvistus, ohjaaminen oikealle kelalle, sekä tietysti mittaustuloksen tuominen AD-muuntimelle. Tällaisella

ratkaisulla laitteesta saatiin niin skaalautuva, että saatiin jopa säilytettyä mahdollisuus passiivisten RFID-tunnisteiden käyttöön. Ainoa este niiden käyttöön valmiissa prototyypissä oli enää pelilaudan mittauskelojen huono laatu. Muutoin käyttöönotto olisi tarvinnut vain FPGA:n uudelleenohjelmoinnin.

### 3.6.1 Pelinappuloiden tunnistet

Pelinappuloiden tunnistus tapahtui niiden pohjaan liimattujen, piirilevyllä valmistettujen LC-sarjakytkentöjen avulla. Sarjakytkennän kela oli valmistettu suoraan monikerroksiselle piirilevyllä ja levyn yläpuolelle juotettiin eri kapasitanssin omaavia kondensaattoreita oikean resonanssitaajuuden saamiseksi. Valmiiden kelojen induktanssi selvitettiin yksinkertaisesti mittaamalla.

RLC-sarjakytkennän resonanssitaajuus saatiin laskemalla yhtälöstä 1.

$$f_0 = \frac{1}{2\pi\sqrt{LC}} \quad (1)$$

Valmiissa prototyypissä valittiin kolmelle käytetylle pelinappulalle resonanssitaajuudet väliltä 500 kHz ... 2000 kHz. Pienempiä taajuuksia testattaessa huomattiin, että pelinappuloiden tunnistaminen oli käytännössä mahdotonta, sillä mittakelan piirin jännitteen muutokset olivat liian pieniä. Toisaalta taas suurempia taajuuksia käytettäessä tulivat vastaan elektroniikan rajoitteet, ennen kaikkea mittaussignaalia vahvistaneen operaatiovahvistimen osalta – taajuuden kasvaessa yli 2 MHz:n, käytetyn operaatiovahvistimen vahvistuskyky heikkeni oleellisesti. Resonanssitaajuuksien välille täytyi myös jättää vapaa 200 kHz kaista, pelinappuloiden virityspiirien kais-tanleveydestä johtuen.

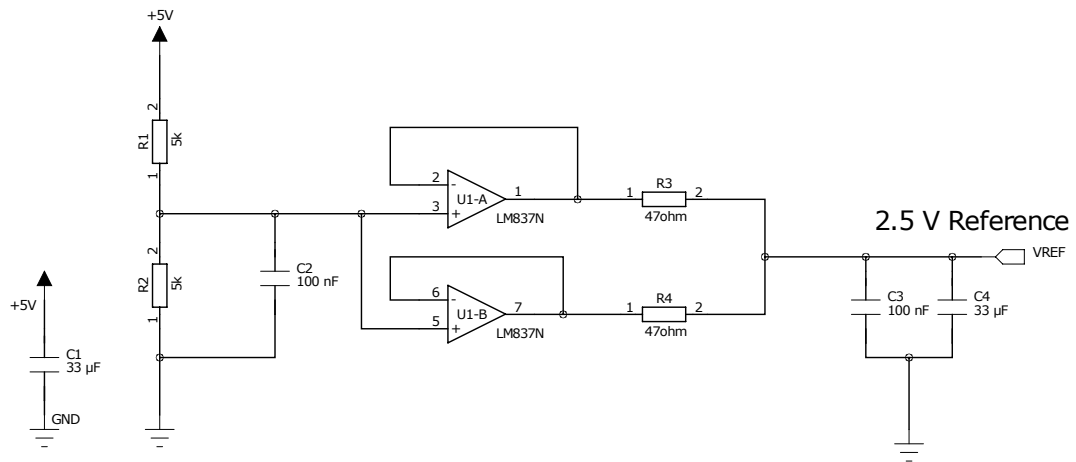
### 3.6.2 Referenssjännitteiden tuottaminen

Prototyypistä haluttiin tehdä mahdollisimman helposti tuotantoon siirrettävä ja eräs tärkeä seikka tämän kannalta oli mahdollisuus paristojen käyttöön. Siksi elektroniikan käyttöjännitteeksi valittiin 5 voltia. Vaikka päätettiin käyttää yksipuoleista jännitelähdettä (0V ... +5V), oli prototyypin elektroniikan kuitenkin tarkoitus tuottaa, käsitellä ja mitata vaihtojännitesignaaleita. Jotta mittaussignaalit eivät olisi päässeet käyttöjännitteeseen nähden negatiivisiksi, nostettiin niiden tasajännitekomponentti nolasta voltista +2,5 volttiin.

Sen jälkeen kun mittasignaalin käsittely päätettiin tehdä operaatiovahvistimilla, oli seuraavana haastena löytää tarkoitukseen sopiva vahvistin. Etsittävälle komponentille oli kaksi vaatimusta: Sen vahvistuksen piti olla riittävän hyvä prototyypissä käytetyillä taajuuksilla (suurin käytetty taajuus oli 2 MHz) ja lisäksi sen piti toimia normaalia pienemmällä käyttöjännitteellä, viidellä voltilla. Näin päädyttiin käyttämään paikallisesta elektroniikkaliikkeestä löytynyttä LM837N-operaatiovahvistina, joka oli edullinen ja ominaisuuksiltaan riittävä.

Kyseisen operaatiovahvistimen kaistanleveysvahvistus (GBW, Gain BandWidth) oli minimissään 15 MHz. Tämä tarkoittaa, että tulevaa signaalia voidaan vahvistaa kertoimella, joka on 15 MHz jaettuna vahvistettavan signaalin taajuudella. Näin ollen 2 MHz taajuisista tulosignaalia voidaan vahvistaa 7,5 kertaisesti. LM837N sisältää neljä erillistä operaatiovahvistina. Yhtä tarvittiin mittakeloille lähetettävän signaalin vahvistukseen ja yhtä mittapiirin jännitteen mittaukseen. Jäljelle jääneitä kahta vahvistinta päätettiin käyttää vertailujännitteen tuottamisessa. Samalla välttyttiin ylimääräisten komponenttien käytöltä. /7/

Referenssjännite tuotettiin kahdella rinnakkaisella jännitteenseuraajakytkenellä (Kuva 10). Lähtöjännite aseteltiin käyttöjännitteen puoleen väliin vastuksista (R1, R2) tehdyllä jännitejakajalla.

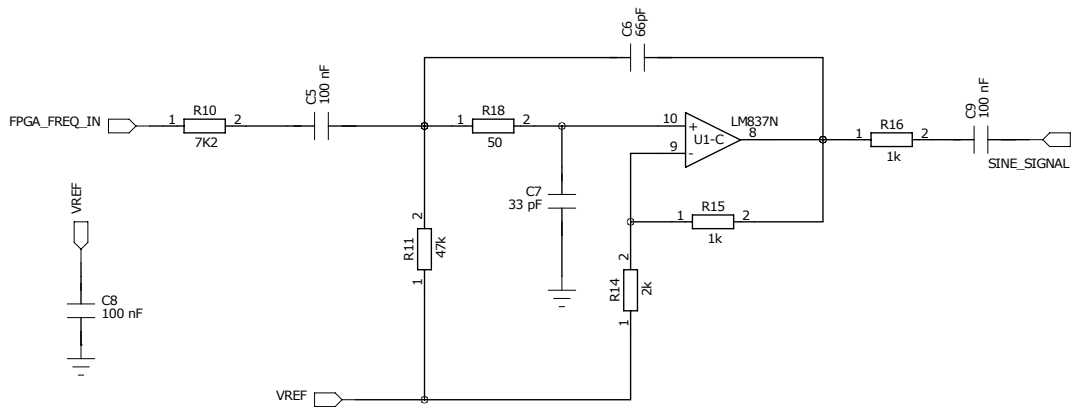


Kuva 10: Vertailujännitteen tuottamiseen käytetty kytkentä

### 3.6.3 Mittasignaalin suodatus ja vahvistus

Varsinainen mittasignaali tuotettiin FPGA-piirillä, mutta sitä ei voitu suoraan ohjata mittakeloille seuraavista syistä: FPGA-piiriä ei haluttu kuormittaa sen vikaantumisen välttämiseksi ja piiri ei olisi pystynyt tuottamaan niin paljon virtaa, mitä pelinappuloiden tunnistamiseen oltaisiin vaadittu. Toinen ongelma oli se, että mittakeloille ei haluttu ohjata suoraan FPGA:lta tulevaa kanttiaaltoa, vaan sen suodattaminen siniaaloksi nähtiin välttämättömäksi – jo pelkästään mittaustarkkuuden takaamiseksi. Näin vältyttiin esimerkiksi tilanteelta, että jonkin pelinappulan resonanssitaajuutta vastaavan kanttiaallon jokin harmoninen olisi osunut samalle taajuudelle, mitä toisen pelinappulan resonanssitaajuus.

Komponenttien säästämiseksi päätettiin mittasignaalin suodatus ja vahvistus toteuttaa yhdellä kytkennällä, käyttäen operaatiovahvistimella toteutettua aktiivista alipäästösuodinta (Kuva 11).

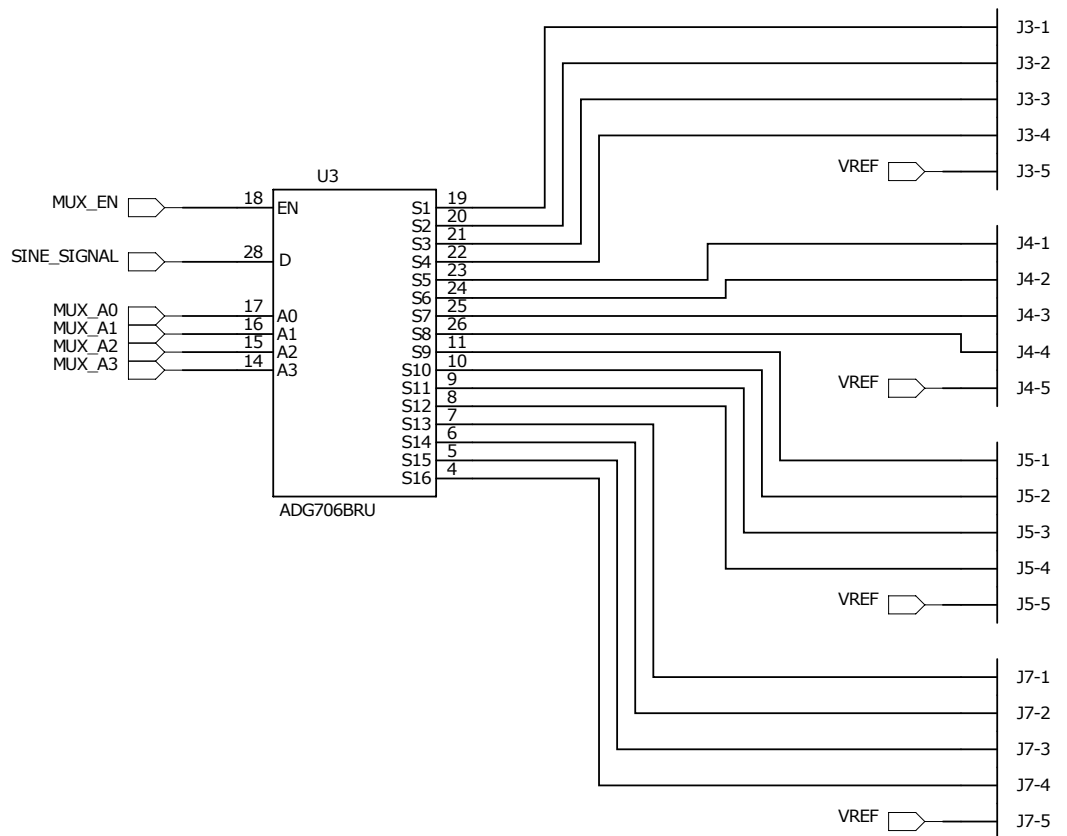


Kuva 11: Mittasignaalin suodatukseen ja vahvistukseen käytetty kytkentä

Kondensaattorilla C5 irroitetaan FPGA-piiriltä tuleva signaali maatasostaan ja nostetaan vertailujännitteeseen (VREF) vastuksen R11 avulla. R18 ja C7 toteuttavat yksinkertaisen alipäästösuotimen, jonka kautta signaali ohjataan vahvistimelle. Operaatiovahvistin U1-C toimii ei-invertoivana vahvistimena. Vahvistuksen suuruuden määrää vastuksilla R14 ja R15 toteutettu jännitejako. Vahvistimella on myös positiivinen takaisinkytkentä, kondensaattorin C6 kautta. Sen tarkoitus on lisätä vahvistusta suuremmilla taajuuksilla ja näin kompensoida alipäästösuotimen vaimentamia, mutta kuitenkin tunnistamisessa käytettyjä taajuuksia.

#### 3.6.4 Mittauskelan valinta multiplekserillä

Mittauskelojen kytkentä päätettiin toteuttaa analogisen multiplekserin avulla. Eri-laisia multipleksereitä ja analogisia kytkinpiirejä testattiin, mutta lopulta päädyttiin ADG706BRU-multiplekseriin. Kyseinen multiplekseri on 16-kanavainen ja sillä on paljon pienempi sisäinen vastus kytketyssä kanavassa, mitä monilla muilla analogisilla multipleksereillä. Sisäinen vastus oli merkittävä tekijä multiplekseriä valittaessa, koska mittakelapiirin jännitettä mittaava kytkentä näki multiplekserin sisäisen vastuksen osana mitattavaa kytkentää ja suuri sisäinen vastus vaikeutti mittaamista. Lisäksi kyseisen tyyppin valintaa puolsi myös sen toimivuus analogiselle multiplekserille alhaisella, viiden voltin käyttöjännitteellä. /8/



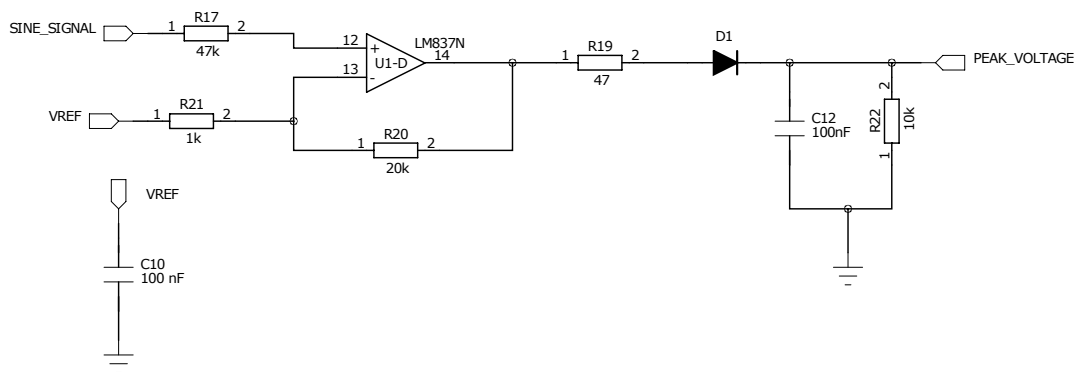
Kuva 12: Mittauskelojen kytkentä multiplekseriin

Kuvassa 12 on kuvattu mittakelojen valintaan käytetyn analogisen multiplekserin kytkentä. Pelilaudalla olevat mittakelat kytkettiin suoraan liittimiin J3, J4, J5 ja J7. FPGA kytkettiin suoraan multipleksereiden nastoihin 14, 15, 16, 17 ja 18. Nastaan 28 tuotiin suodatettu ja vahvistettu mittasignaali.

### 3.6.5 Mittauskelojen jännitteen mittaaminen

Kuten edellä on kuvattu, mittauspiirin impedanssi kasvaa tietyllä taajuudella, jos sen päällä on oikean resonanssitaajuuden omaava pelinappula. Koska pelilaudan kela-  
piirin impedanssia ei voida kuitenkaan suoraan mitata, täytyy mittaaminen tehdä jonkin muun suureen avulla. Tässä tapauksessa apuna käytettiin jännitettä. Kuvan 11 kytkennässä on operaatiovahvistimen U1-C lähde sarjassa yhden kilo-ohmin vastus ja sen jälkeen sadan mikrofaradin kondensaattori, jonka tarkoitus on irrottaa lähtösignaali nollatasostaan. Kondensaattori kytkeytyy analogisen multiplekserin

nastaa 28 ja tätä kautta myös mittakeloihin. Samalla se kytkeytyy myös kuvan 13 huippuarvon ilmaisimeen.



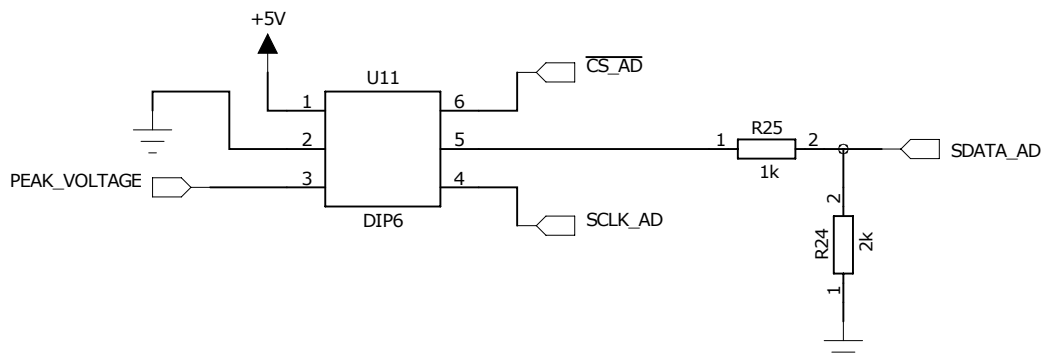
*Kuva 13: Mittaustuloksen vahvistin ja huippuarvon ilmaisim*

Vaikka pelilaudalla olevissa mittakeloissa ja analogisessa multiplekserissä olikin resistanssia, saatiin näiden sarjaankytkennän yhteivastus noin sadan ohmin luokkaan., kun taas pelinappuloiden mittakelan impedanssiin aiheuttamat muutokset olivat vain kymmenen ohmin luokkaa. Nämä arvot eivät kuitenkaan olleet suuria verrattuna vastuksen R16 (Kuva 11) resistanssiin. Toisin sanoen, R16 määräsi mittapiirin kautta kulkevan virran. Koska virta mittapiirissä oli vakio, sen yli vaikuttava jännite (Kuva 12, jännite välillä SINE\_SIGNAL <-> VREF) oli suoraan verrannollinen sen impedanssiin, yhtälön 2 mukaisesti.

$$U=Z \cdot I \quad (2)$$

Mittapiirin jännite oli kuitenkin vaihtovirtaa, jota on vaikea tulkita digitaalisesti. Näin ollen se muutettiin helpommin AD-muuntimella luettavaan muotoon, eli tasajännitteeksi. Muunnos tehtiin huippuarvon ilmaisimella (Kuva 13, diodi D1 ja kondensaattori C12), mitä ennen täytyi tulevaa signaalia kuitenkin vahvistaa. Vahvistukseen käytettiin operaatiovahvistinkytkentää (Kuva 13, vasen puoli). Näin jännitteen vaihtelut mittapiirisissä saatiin paremmin esiin ja samalla vältyttiin sen ylimääräiseltä kuormittamiselta.

Jännite mitattiin kuvan 14 mukaisella kytkennällä. AD-muunnin (U11) kytkeytyy FPGA-piiriin SPI-väylänsä avulla.



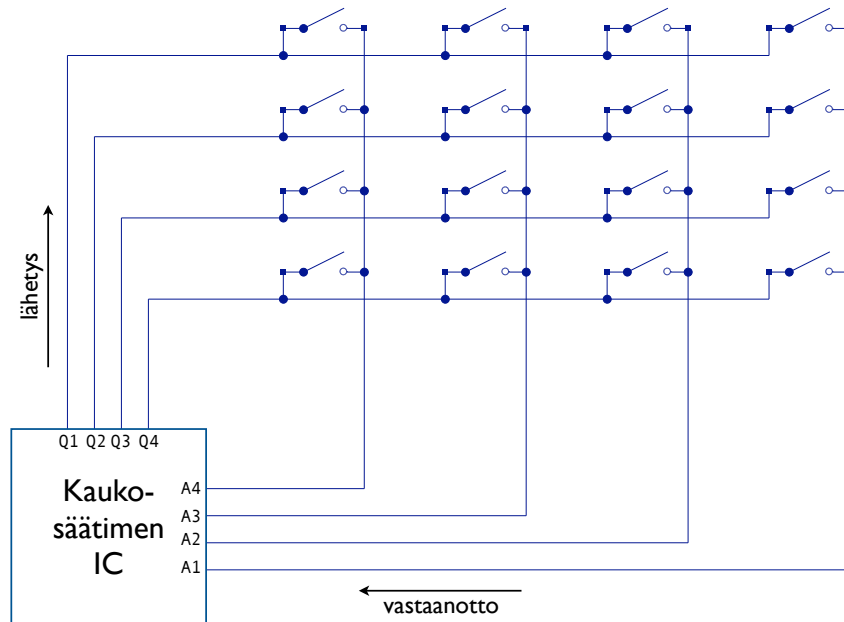
Kuva 14: AD-muuntimen kytkentä

### 3.6.6 Kaukosäätimen ohjaus kahdella multiplekserillä

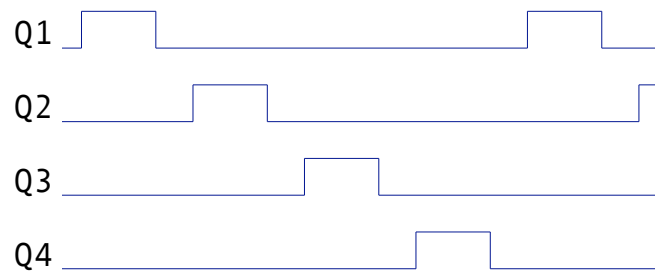
Kolmannen osapuolen rakentaman yleiskaukosäätimen käyttäminen vaikutti prototyypin suunnittelun alkuvaiheessa hyvältä idealta. Valmista laitetta käyttämällä säätettäisiin paljon suunnitteluun ja rakentamiseen menevää aikaa ja lisäksi valmiiden yleiskaukosäädinten koodikirjastot ovat usein hyvin kattavia. Tämä nähtiin eduksi, mikäli prototyyppiä haluttaisiin myöhemmin testata ja esitellä eri paikoissa ja eri DVD-soittimia apuna käyttäen.

Ongelmatonta valmiin kaukosäätimen liittämisen ei kuitenkaan ollut, kuten tultiin suunnitteluvaiheessa huomaamaan. Kun sopivan tuntuinen yleiskaukosäädin oli ensin ostettu kaupasta, purettiin se osiin ja perehdyttiin sen toimintaan. Tällöin huomattiin, että ostetussa kaukosäätimessä oli koodikirjasto sekä matriisinäppäimistön lukemiseen ja infrapunasiinaalin tuottamiseen käytetty elektroniikka integroitu yhdelle IC-piirille. Piirin ulkopuolisten komponenttien tarkoitus oli käytännössä IR-LED:in ohjaus. Koska piirillä ei ollut minkäänlaista ohjaukseen soveltuvaa väylää, jäi matriisinäppäimistön (Kuva 15) matkiminen ainoaksi tavaksi ohjata kaukosäätimen toimintaa.



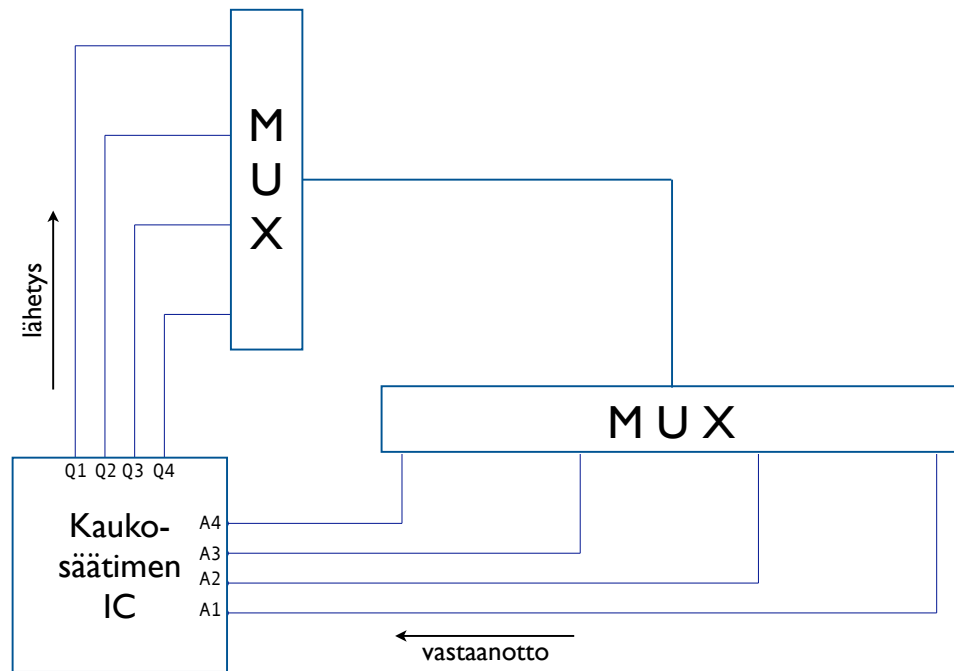


Kuva 15: Matriisinäppäimistön toimintaperiaate



Kuva 16: Luku-signaalien ajoitus

Ongelmaksi liittämässä muodostui matriisinäppäimistön lukutapa. Näppäimistöä luetaan läpi niin, että näppäimistön ohjain kytkee jännitteen yhteen riviin kerrallaan (Kuva 16) ja lukee tämän jälkeen tuloista, oliko jokin kytkin suljettuna. Näin ollen, jos näppäimistön ohjainta oltaisiin niin sanotusti huijattu, matkimalla suoraan näppäimistön toimintaa, olisi signaalien ajastaminen ollut ensiarvoisen tärkeää. Hetken pohdiskelun jälkeen kaukosäätimen ohjaamiseen kuitenkin keksittiin hyvin tarkoitukseen sopiva ratkaisu, ja mikä parasta ei sen toteuttamiseen vaadittu kuin kaksi analogista multiplekseriä (Kuva 17).



Kuva 17: Matriisinäppäimistön korvaaminen analogisilla multipleksereillä

Ratkaisussa fyysinen näppäimistö korvattiin analogisilla multipleksereillä. Multipleksereillä pystyttiin kytkemään tietty rivi sähköisesti tiettyyn sarakkeeseen ja ajoitus menetti täten merkityksensä. Multipleksereiden ohjaussignaalit tuotiin suoraan FPGA-piiriltä, joka näki multipleksereiden kytkennän yhtenä sarjaväylänä.

Tässäkin ratkaisussa tosin oli omat puutteensa, sillä kaukosäätimen saamiseksi ohjelmointitilaan, piti painaa kahta painiketta samanaikaisesti. Multipleksერიkytkentä pystyi matkimaan vain yhden painikkeen painamista kerrallaan.

### 3.7 Digitaalisen osion suunnittelu

Koululla oli varastossa Alteran Nios II -kehitysalusta, joka oli varustettu Alteran valmistamalla, Cyclone II -tyypin FPGA-piirillä. Kehitysalusta oli reilusti ylimitoitettu projektin tarpeisiin, mutta se päätettiin ottaa käyttöön sen mukana tulleen Nios-kehitysympäristön lisenssin vuoksi.

Kehitysalustassa oli useampia yleiskäyttöisiä (GPIO, General Purpose IO) liitäntäriemuja. Näitä päätettiin käyttää muuhun elektroniikkaan kytkeytymiseen ja niistä valit-

tiin parhaiten tarkoitukseen sopivat, eli sellaiset, joista saatiin suoraan käyttöjännite ja riittävä määrä signaaleita muille piirilevyille.

Kun liitäntöjen käyttö oli päätetty, täytyi seuraavaksi valita, mikä FPGA:n ja muun elektroniikan välinen signaali kytkettäisiin mihinkin liittimen nasaan. Tämän jälkeen työtä jatkettiin Alteran Quartus- ja Nios-kehitysympäristöillä. /10/

Ensimmäiseksi Quartus-ohjelmalla aloitettiin uusi projekti, valittiin kytkentäkaavanäkymä ja määriteltiin tulo- ja lähtösignaalit vastaamaan suunniteltuja. Seuraavaksi suunnitelmaan lisättiin Nios-proessori, jolle luotiin vaaditut IO-väylät. Prosessin jälkeen Quartus lisäsi kytkentäkaavaan Nios-proessorin uutena komponenttina. Sen kytkentä FPGA:n sisäisiin komponentteihin ja ulkoisiin IO-väylöihin täytyi tehdä kuitenkin käsipelillä. FPGA:n sisälle integroitiin Nios:n lisäksi kaksi muuta komponenttia. Nämä olivat kellosignaalin jakajapiiri ja AD-muuntimen ohjain. Molemmat kirjoitettiin VHDL-kuvauskielellä, Quartus-ympäristössä. Kun VHDL-kuvaus oli valmis, Quartus osasi laatia kuvauksen tuloja ja lähtöjä vastaavan komponentin kytkentäkaavaan. Tämän jälkeen komponenttien liittäminen tapahtui graafisen työkalun avulla.

### 3.7.1 Mittaussignaalin tuottaminen

Varsinainen mittaussignaali prototyypissä tuotetaan digitaalisesti, FPGA-piirin järjestelmäkellon taajuutta (66 MHz) jakamalla. Jakamisen suorittaa VHDL-kielellä kuvattu lohko, joka ottaa tuloinaan jakajan (Nios:lta, 8 bittiä) ja järjestelmän kellosignaalin. Näiden avulla se tuottaa lähtöönsä halutun taajuuden signaalin. Mikäli jakajan arvo on 0, on lähtö korkea-impedanssisessa tilassa.

Tämän lohkon täydellinen VHDL-kielinen kuvaus on liitteenä.

### 3.7.2 AD-muuntimen ohjaus

Kuten aikaisemmin tässä raportissa kerrottiin, tukee Nios suoraan SPI-väylän käyttöä, mutta koska väylän käyttö ei ollut ohjelmallisesti kovinkaan yksinkertaista, päädyttiin käyttämään aiemmin suunniteltua VHDL-lohkoa. Kyseinen VHDL-lohko tehtiin samaa prototyyppiä varten, T51-prosessorin kanssa toimivaksi. Ohjain tarjosi prosessorille rinnakkaisen dataväylän (8 bittiä) josta mittausarvo luettiin ja kaksisuuntaisen, 3 bittiä leveän ohjausväylän (2 tuloa, yksi lähtö).

Ohjainta hallittiin seuraavasti: Ensimmäisellä signaalilla ohjainta käskettiin aloittamaan muunnos. Kun muunnos oli valmis, toi ohjain AD-muuntimen arvon dataväylälle ja nosti toisen, merkkisignaalin ylös. Kun prosessori oli lukenut datan, se kuittasi tämän kolmannella signaalilla, jolloin ohjain palasi takaisin odotustilaan.

Tämän lohkon täydellinen VHDL-kielinen kuvaus on liitteenä.

### 3.8 Pelilaudan valmistelu ja liittäminen sekä prototyypin viimeistely

Projektin loppumetreillä oli edessä vielä eräs työläs haaste, nimittäin pelilaudan kasaaminen ja liittäminen elektroniikkaan. Sen jälkeen kun pelilaudan pintagrafiikka oli saatu valmiiksi ja painettua huomattiin, että sille ei löytynyt vastaavaa mittakela-arkkia. Toisin sanoen painetut kelat olivat eri paikassa kuin missä niiden olisi kuulunut olla pelilaudan grafiikan mukaan. Ongelma saatiin kuitenkin korjattua leikkaamalla ja liimaamalla painettuja keloja eri arkeilta ja tarkan kohdistamisen jälkeen oli pelilauta kasassa.

Mittakelojen liittäminen elektroniikkaan olikin sen jälkeen hieman monimutkaisempaa. Juottaminen ei onnistunut, eikä löytynyt mitään hyvää liimaa, jonka johtavuus olisi ollut tarpeeksi hyvä. Johtavat liimat taas eivät pitäneet riittävän hyvin. Lopulta liitokset saatiin tehtyä niin, että kuorittiin liitettävää kaapelia noin kolmen senttimetrin matkalta, painettiin kuorittu osa vasten painettua johdinpintaa ja liimattiin se kuumaliimalla muutamasta pisteestä kiinni. Kontaktia parannettiin hopeapastalla,

levittämällä sitä runsaasti avoimen johtimen päälle. Kun pasta oli kuivunut, peitettiin avoin liitos vielä kuumaliimalla, liitoksen suojaamiseksi. Näin pelilauta saatiin johdotettua.

Kun pelilauta oli valmis, liitettiin kaikki osat paikalleen (kaukosäädin, FPGA-kehitysalusta, analogisen elektroniikan sisältävä piirilevy ja itse pelilauta) ja päästiin aloittamaan varsinainen testausvaihe.

### 3.9 Valmiin prototyypin testaus

Kaikeksi hämmästykseksi, elektroniikka toimi kuten pitikin. Ainoana poikkeuksena oli, että kaksi mittakelaa kuudestatoista olivat hajonneet kasausvaiheessa, mutta tässä vaiheessa vikaa oli käytännössä mahdoton korjata. Näin varsinainen testausvaihe päästiin kuitenkin aloittamaan.

Testausvaihe kesti vain muutaman arkipäivän – tällöin tuli vastaan projektin deadline. Suurimmat ongelmat saatiin kuitenkin korjattua pois Nios:n ohjelmistosta ja pelilaudalla pystyi näiden kahden päivän jälkeen pelaamaan.

Lyhyen testijakson jälkeen prototyypille laadittiin Nelostuote Oy:n toimesta kattavat kasausohjeet ja valmis laite pakattiin ja luovutettiin heille lainaan.

#### 4 PROTOTYYPIN JATKOKEHITYS JA JOHTOPÄÄTÖKSET

Sulautettujen järjestelmien käyttömahdollisuuksia on valtavasti ja voisikin sanoa, että niiden käytöllä on vain mielikuvitus rajana. Projektissa, jonka tuloksesta tämä raporttikin kertoo, todistaa että sulautetuilla järjestelmillä saattaa olla osansa myös lautapeliä tulevaisuudessa ja tämän raportin alussa käytiinkin läpi joitakin jatkokehitysmahdollisuuksia.

Suurimmaksi ongelmaksi elektroniikan sulauttamiselle pelilautaan tuli tämän projektin tutkimuksen osalta kallis hinta, sekä elektroniikan rajoitteet ja suunnittelun mukanaan tuomat haasteet. Elektronisia lautapelejä lienee mahdoton myydä kuluttajille suotuisaan hintaan vielä nykyteknologialla.

Projekti toi mukanaan paljon uusia haasteita; sellaisia joita ei oppitunneilla istuessaan edes osannut kuvitellakaan. Projektissa työskentely myös opetti valtavasti uusia asioita, jos ei niinkään elektroniikasta, niin ainakin projektien vetämisestä ja niissä työskentelystä. Samalla tuli huomattua, kuinka tärkeää tehokas töiden jakaminen, raportointi ja dokumentointi on, varsinkin jos saman asian parissa tekee töitä useampi henkilö.

## LÄHTEET

1. Wikipedia, RFID (Radio frequency identification) [verkkodokumentti].  
Saatavissa:  
<http://fi.wikipedia.org/wiki/RFID>
2. Prototype Express, RFID Technology White Paper [verkkodokumentti].  
Saatavissa:  
<http://www.prototypexpress.com/rfidwhitepaper.htm>
3. Wikipedia, Electronic paper [verkkodokumentti]. Saatavissa:  
[http://en.wikipedia.org/wiki/Electronic\\_paper](http://en.wikipedia.org/wiki/Electronic_paper)
4. Coroflot, Nikita Golovlev Portfolio [verkkodokumentti]. Saatavissa:  
[http://www.coroflot.com/public/individual\\_set.asp?from\\_url=true&sort\\_by=1&set\\_id=83619&individual\\_id=144654](http://www.coroflot.com/public/individual_set.asp?from_url=true&sort_by=1&set_id=83619&individual_id=144654)
5. Wikipedia, Hopea [verkkodokumentti]. Saatavissa:  
<http://fi.wikipedia.org/wiki/Hopea>
6. Wikipedia, Serial Peripheral Interface Bus [verkkodokumentti]. Saatavissa:  
[http://en.wikipedia.org/wiki/Serial\\_Peripheral\\_Interface\\_Bus](http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus)
7. National Semiconductor, LM837 datasheet [verkkodokumentti]. Saatavissa:  
<http://www.national.com/ds/LM/LM837.pdf>
8. Analog Devices, ADG706/ADG707 datasheet [verkkodokumentti]. Saatavissa:  
[http://www.analog.com/static/imported-files/data\\_sheets/ADG706\\_707.pdf](http://www.analog.com/static/imported-files/data_sheets/ADG706_707.pdf)
9. Altera, Nios II Development Kit - Cyclone II Edition [verkkodokumentti].  
Saatavissa:  
<http://www.altera.com/products/devkits/altera/kit-nios-2c35.html>
10. Altera, Design Software [verkkodokumentti]. Saatavissa:  
<http://www.altera.com/products/software/sfw-index.jsp>

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity clock_div is
  port(
    in_clk           : in std_logic;
    in_divider       : in std_logic_vector(7 downto 0);
    div_out          : out std_logic
  );
end clock_div;

architecture RTL of clock_div is
  signal count_val : integer range 0 to 255;
begin
  process(in_clk)
    variable pdiv : std_logic;
    variable counter : integer range 0 to 255;
  begin
    if rising_edge(in_clk) then
      if in_divider /= "00000000" then
        if(counter = (to_integer(unsigned(in_divider))-1) then
          pdiv := not pdiv;
          counter := 0;
        else
          counter := counter +1;
        end if;
        div_out <= pdiv;
      else
        div_out <= 'Z';
      end if;
    end if;
  end process;
end RTL;
```



```

library ieee;
use ieee.std_logic_1164.all;

package adc_int_package is
    constant ADC_RESOLUTION : integer := 12;
    constant WAIT_MULTIPLIER : integer := 1;
end adc_int_package;

library ieee;
library work;
use ieee.std_logic_1164.all;
use work.adc_int_package.all;

entity adc_interface is
    port(
        in_clk           : in  std_logic;

        -- Nios interface
        in_rst           : in  std_logic;
        in_do             : in  std_logic;
        in_ack           : in  std_logic;

        done_o           : out std_logic;
        sdata_o          : out std_logic_vector(7 downto 0);

        -- ADC interface
        in_sdata         : in  std_logic;
        sclk_o           : out std_logic;
        n_adc_en_o       : out std_logic
    );
end adc_interface;

architecture RTL of adc_interface is
    constant ADC_TRAILING_ZEROES : integer := 12 - ADC_RESOLUTION;

    type STATE_TYPE is (INIT, START, TRIGGER, RD_BIT, EOC, WAIT_RD);

    signal sclk           : std_logic;
    signal n_adc_en       : std_logic := '1';
    signal conv_done      : std_logic;
    signal sdata          : std_logic_vector((19) downto 0);
begin
    done_o    <= conv_done;
    sclk_o    <= sclk;
    n_adc_en_o <= n_adc_en;

    process(in_clk)
        variable wait_counter : integer range 0 to 33330000;
        variable sample_counter : integer range 0 to 19;
        variable state : STATE_TYPE := INIT;
        variable next_state : STATE_TYPE := INIT;
    begin
        if(in_rst='1') then
            state := INIT;
            n_adc_en <= '1';
            sclk <= '1';
            conv_done <= '0';
        elsif rising_edge(in_clk) then
            if(wait_counter > 0 and next_state /= EOC) then

```

```

wait_counter := wait_counter -1;
else
state := next_state;
case state is
when INIT =>      -- CS ylhäällä, SCLK ylhäällä.
n_adc_en <= '1';
sclk <= '1';
conv_done <= '0';
if(in_do = '1') then
next_state := START;
end if;

when START =>    -- CS alas, SCLK ylhäällä
sample_counter := 0;
n_adc_en <= '0';
sclk <= '1';
next_state := TRIGGER;^M
wait_counter := 1 * WAIT_MULTIPLIER;

when TRIGGER =>  -- CS alhaalla, SCLK alas
sclk <= '0';
next_state := RD_BIT;
wait_counter := 4 * WAIT_MULTIPLIER;

when RD_BIT =>   -- SCLK ylös, lue data. Jos on
                -- kerätty 19 näytettä,
                -- siirrytään pois. Tällöin CS
                -- alhaalla, SCLK ylhäällä
sdata(sample_counter) <= in_sdata;
sclk <= '1';
if(sample_counter = 15) then
next_state := EOC;
wait_counter := 30 * WAIT_MULTIPLIER;
conv_done <= '1';
for i in 4 to 11 loop
-- käännetään LSB...MSB -> MSB...LSB
sdata_o(11- i) <= sdata(i);
end loop;
else
next_state := TRIGGER;
sample_counter := sample_counter +1;
wait_counter := 4 * WAIT_MULTIPLIER;
end if;

when EOC =>     -- CS ylhäällä, SCLK ylhäällä,
                -- ilmoitetaan saatavilla
                -- olevasta datasta. Jos data on
                -- luettu, nollataan ilmoitus.
n_adc_en <= '1';
if(wait_counter = 0) then
next_state := WAIT_RD;
else
wait_counter := wait_counter - 1;
end if;

if(in_ack = '1') then
conv_done <= '0';
end if;

when WAIT_RD => -- Odotetaan datan hakemista
if(in_ack = '1') then
conv_done <= '0';
end if;

```

```
        if(conv_done = '0') then
            next_state := INIT;
        end if;

        when others =>
            end case;
        end if;
    end process;
end RTL;
```