



Matti Kivistö

LÄMPÖTILA-ANTURIN KÄYTTÖNOTTO ZIGBEE- YMPÄRISTÖSSÄ

LÄMPÖTILA-ANTURIN KÄYTTÖÖNOTTO ZIGBEE- YMPÄRISTÖSSÄ

Matti Kivistö
Opinnäytetyö
14.11.2011
Tietotekniikan koulutusohjelma
Oulun seudun ammattikorkeakoulu

OULUN SEUDUN AMMATTIKORKEAKOULU TIIVISTELMÄ

Koulutusohjelma	Opinnäytetyö	Sivuja	+	Liitteitä
Tietotekniikka	Insinöörityö	38	+	2
Suuntautumisvaihtoehto	Aika			
Langaton tietoliikenne	2011			
Työn tilaaja	Työn tekijä			
Tietotekniikan osasto	Matti Kivistö			
Työn nimi				
Lämpötila-anturin käyttöönotto ZigBee-ympäristössä				
Avainsanat				
Anturit, E-blocks, Flowcode, Lämpötilan mittaus, ZigBee				

Insinöörityön tarkoituksena oli tutustua E-blocks-tuotteisiin, ZigBee-tekniikkaan, Vernierin kehittämään analogiseen lämpöanturiin sekä vuokaavio-ohjelmointiin ja niitä hyödyntäen muodostaa lämpötilaa mittaava kokonaisuus. Tavoitteena oli tehdä lämpömittarisovellus, jossa lämpöanturin mittaamat resistanssiarvot muutetaan celsiusasteiksi, jonka jälkeen ne lähetetään langattomasti ZigBee-tekniikkaa hyödyntäen alustalta toiselle ja lopuksi tulostetaan lämpötila graafiselle LCD-näytölle. Työn tilaajana toimi Oulun seudun ammattikorkeakoulun tietotekniikan osasto.

Työ toteutettiin pääosin Oulun seudun ammattikorkeakoulun laboratoriotiloissa. Työn toteutusaika oli syksyllä 2011. Työn lopputuloksena muodostui lämpömittarisovellus, joka mittausten perusteella mittasi luotettavia lämpötila-arvoja. Jatkossa sovellusta voidaan kehittää esimerkiksi niin, että tulostusvaiheessa lämpötilat esitetään tarkemmin, desimaaleja käyttäen. Jatkossa työtä voidaan myös hyödyntää hyvinvointipuolen opiskelijoiden laboratorioharjoituksissa.

ALKUSANAT

Kiitän Oulun seudun ammattikorkeakoulua, joka mahdollisti opinnäytetyöni tekemisen. Kiitokset lisäksi lehtori Riitta Ronnulle työn ohjauksesta ja projektisuunnittelija Juha Juutille työn aiheesta. Kiitokset myös lehtori Tuula Hopeavuorelle opinnäytetyön tekstin ohjauksesta. Erityiskiitokset vanhemmilleni, jotka olivat tukenani koko opiskelujen ajan.

Oulussa 14.11.2011

Matti Kivistö

SISÄLLYS

TIIVISTELMÄ	3
ALKUSANAT	4
SISÄLLYS	5
LYHENTEET JA TERMIT	7
1 JOHDANTO.....	9
2 MATRIX MULTIMEDIA.....	10
2.1 E-blocks	10
2.2 Käytettävät kortit	11
2.2.1 AVR multiprogrammer -mikrokontrollerikortti	11
2.2.2 Sensorikortti	12
2.2.3 Graafinen LCD-näyttö	12
2.3 Flowcode-ohjelmointityökalu	13
3 ZIGBEE	15
3.1 Tekniset tiedot.....	15
3.2 Toimintamuodot	16
3.3 Topologiat	17
3.3.1 Tähtitopologia	17
3.3.2 Cluster tree -topologia.....	18
3.3.3 Mesh-topologia	19
3.4 Tietoturva	20
3.5 ZigBee-kortti.....	21
4 ANTURIN KÄYTTÖÖNOTTO	22
4.1 Lämpöanturi	22
4.2 Käytettävä kehitysympäristö.....	23
4.3 Alkuasetukset.....	24
4.4 Lämpömittari ja ohjelmien muodostus.....	27
4.4.1 Päätelaite	28
4.4.2 Koordinaattori	30
4.4.3 Valmis kokoonpano.....	32
5 YHTEENVETO JA POHDINTA	35

LÄHTEET 37

LIITTEET

Liite 1. Päätelaitteen vuokaavio

Liite 2. Koordinaattorin vuokaavio

LYHENTEET JA TERMIT

AES	Advanced Encryption Standard. Salausmenetelmä.
ARM	Advanced RISC Machine. 32-bittinen mikroprosessoriarkitehtuuri.
AVR	Atmelin kehittämä mikrokontrolleriperhe, joka sisältää 8-bittisiä mikrokontrollereita.
CSMA-CA	Carrier Sense Multiple Access with Collision Avoidance. MAC:n käyttämä tekniikka, joka määrää, milloin ZigBee-laite voi lähettää tietoa.
DSSS	Direct Sequence Spread Spectrum. Tekniikka lähetettävän tiedon muokkaukseen.
IEEE	Institute for Electrical and Electronics Engineers. Amerikkalainen yhdistys, joka on vastuussa tietoliikennetekniikoista ja niiden kehittämisestä.
LCD	Liquid Crystal Display. Nestekidenäyttö.
LED	Light-Emitting Diode. Loistediodi.
MAC	Medium Access Control. Menetelmä tiedonsiirron kontrolloimiseen.
O-QPSK	Offset Quadrature Phase Shift Keying. Modulointimenetelmä.

PCB	Printed Circuit Board. Piirilevy.
PIC	Peripheral Interface Controller. Mikrokontrolleriarkkitehtuuri.
RS232	Recommended Standard 232. Sarjaväylä. Kahden laitteen väliseen kommunikointiin tarkoitettu tietoliikenneportti.
USB	Universal Serial Bus. Sarjaväyläarkkitehtuuri oheislaitteiden liittämiseksi tietokoneeseen.
WPAN	Wireless Personal Area Network. Likiverkko. Tiedonsiirtoverkko, jossa henkilökohtaiset laitteet voivat kommunikoida keskenään.
ZigBee	IEEE 802.15.4 -standardin mukainen lyhyen kantaman tietoliikenneverkko.
802.15.4	IEEE:n standardi.

1 JOHDANTO

Tässä työssä oli tarkoituksena tutustua Matrix Multimedia E-blocks-rakennus-sarjaan sekä vuokaavio-ohjelmointiin ja niitä hyödyntämällä muodostaa lämpötilaa mittaava kokonaisuus. Lämpötilan mittaukseen käytettiin Vernierin analogista lämpöanturia. Tavoitteena oli kokonaisuus, jossa mitatut lämpötilat lähetetään langattomasti lämpötilalukeman tulostavaan vastaanottolaitteistoon.

Tärkeitä lähtökohtia työn tekemiselle olivat anturin hyödyntäminen sekä langattoman tekniikan käyttö tiedonsiirrossa. E-blocks-ympäristö mahdollisti kyseiset lähtökohdat, jolloin käyttöön otettiin saatavilla ollut Vernierin anturi. Työn langattoman osuuden mahdollisti ZigBee-tekniikan hyödyntäminen.

Työn aihe tuntui turvalliselta ja loogiselta, koska neljäntenä opintovuonna tutustuttiin E-blocks-ympäristöön laboratorioharjoitusten muodossa. Koska aihe oli siis jo ennestään tuttu ja anturit vaikuttivat yleisesti kiinnostavilta laitteilta, päätti ottaa opinnäytetyöaiheen vastaan. Työ vaikutti sopivan haasteelliselta, koska siinä sovellettaisiin E-blocks-ympäristöä yhdistämällä siihen laite, joka ei kuulu E-blocks-tuotteisiin. Työ on myös ajankohtainen, koska se sisältää langatonta tiedonsiirtoa. Nykyään yhä useammat tietotekniset sovellukset hyödyntävät langattomuutta, joten tämä työ sopii erinomaisesti jatkoksi niille.

2 MATRIX MULTIMEDIA

Matrix Multimedia on vuonna 1993 perustettu yritys, joka kehittää tietoteknisiä tuotteita. Tuotteet sisältävät sovelluksia teollisuus-, koulutus- ja kotikäyttöön. Keskeisimpiä tuotteita ovat Flowcode-ohjelmointityökalut ja E-blocks-rakennussarjat. Muita tuotteita ovat Loctronics-kehitysympäristöt, MIAC-kontrollointiyksiköt sekä ECIO-mikrokontrollerit. (1.)

2.1 E-blocks

E-blocks-kehitysympäristöt ovat hyvin keskeisessä asemassa niin Matrix Multimedian kannalta kuin tässä opinnäytetyössäkin. Kehitysympäristöt koostuvat pääyksiköistä, joita ovat multiprogrammer-mikrokontrollerit, ja niihin liitettävistä laajennusosista, joita on saatavilla yli 150 erilaista. Halutut laajennuskortit yhdistetään mikrokontrollereiden RS232 (Recommended Standard 232) -portteihin ja valmiit yhdistelmät ruuvataan kiinni peltilevyihin. Lopputuloksena on yhtenäinen tietoliikennesysteemi ja kehitysympäristö. (2, s. 3.)

Piirilevyjä on kahdentyyppisiä: upstream-kortit ja downstream-kortit. Upstream-kortit ovat yksiköitä, jotka kontrolloivat tiedon kulkua systeemissä. Niitä ovat yleensä ohjelmointiyksiköt, esim. multiprogrammer-mikrokontrollerit. Downstream-korteiksi luetaan laitteet, joiden toiminnan määrää toinen laite eli jokin upstream-laite. Downstream-kortteja ovat esim. LED (Light-Emitting Diode) -kortit, LCD (Liquid Crystal Display) -kortit ja painonappikortit. (2, s. 3.)

2.2 Käytettävät kortit

2.2.1 AVR multiprogrammer -mikrokontrollerikortti

Kehitysympäristön kannalta tärkein osa on AVR-mikrokontrollerikortti (upstream). Se on koko systeemin pääyksikkö, johon halutut laajennuskortit liitetään. Pääyksikön (kuva 1) kautta saadaan virta systeemiin, ja sen kautta ohjelmoidaan kehitysympäristöön laajennuskorttien mukaiset toiminnot.

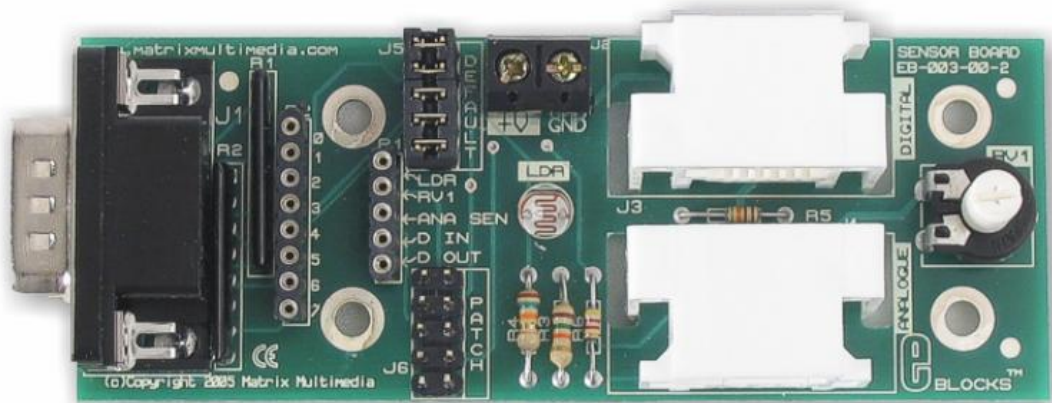


KUVA 1. AVR multiprogrammer -mikrokontrollerikortti (3)

Pääyksikössä on neljä RS232-porttia, joihin voidaan liittää ohjelmoitavat laajennuskortit. Pääyksikön ja lisäkorttien ohjelmointi tapahtuu ISP (In-System Programmer) -ohjelmointiyksiköllä, jonka toinen pää yhdistetään tietokoneen USB (Universal Serial Bus) -porttiin ja toinen pää mikrokontrollerikortin ohjelmointiliittimeen. Liittimiä on kolme, jotka eroavat toisistaan pinnimääriltään. Liittimet ja pinnit ovat siis J9 (28-pin), J10 (8-pin/20-pin) ja J3 (40-pin). Keskimääräinen käyttöjännite on +14 V, minimi on +7 V ja maksimi on +20 V. (3, s. 3–7.)

2.2.2 Sensorikortti

Sensorikortin (kuva 2) avulla saadaan yhdistettyä käytettävät anturit järjestelmään ja toimintavalmiuteen. Valittavana on yli 40 erilaista anturia, jotka koostuvat digitaalisista ja analogisista tuotteista. Sensorikortti toimii myös sellaiseena anturina. Siihen on sisäänrakennettu valosensori, jolla voidaan tarkastella järjestelmän toimivuutta. (4, s. 3.)

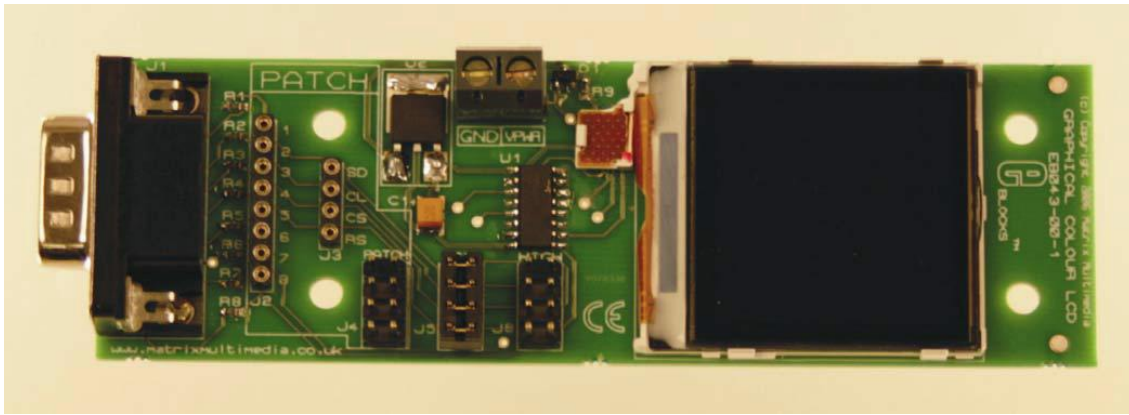


KUVA 2. Sensorikortti (4)

Sensorikortissa on valittavana joko digitaalinen tai analoginen sensoriliitäntä käytettävän sensorin tyypin mukaan. Lisäksi kortin takaosassa on potentiometri jännitteen säätöä varten. Sopiva käyttöjännite eri antureille löytyy väliltä 3,3–5 V. (4, s. 4–6.)

2.2.3 Graafinen LCD-näyttö

LCD-näytön (kuva 3) avulla saadaan anturikorttien toiminta havainnollistumaan eli mittaustulokset saadaan näkyville reaaliajassa. Näytön resoluutiotarkkuus on 130 x 130 pikseliä ja sillä voidaan esittää jopa 65 535 erilaista väriä. Tekstiä näytöllä pystytään tulostamaan 16 riviä, jossa yksi rivi sisältää 21 merkkiä. (5, s. 3.)

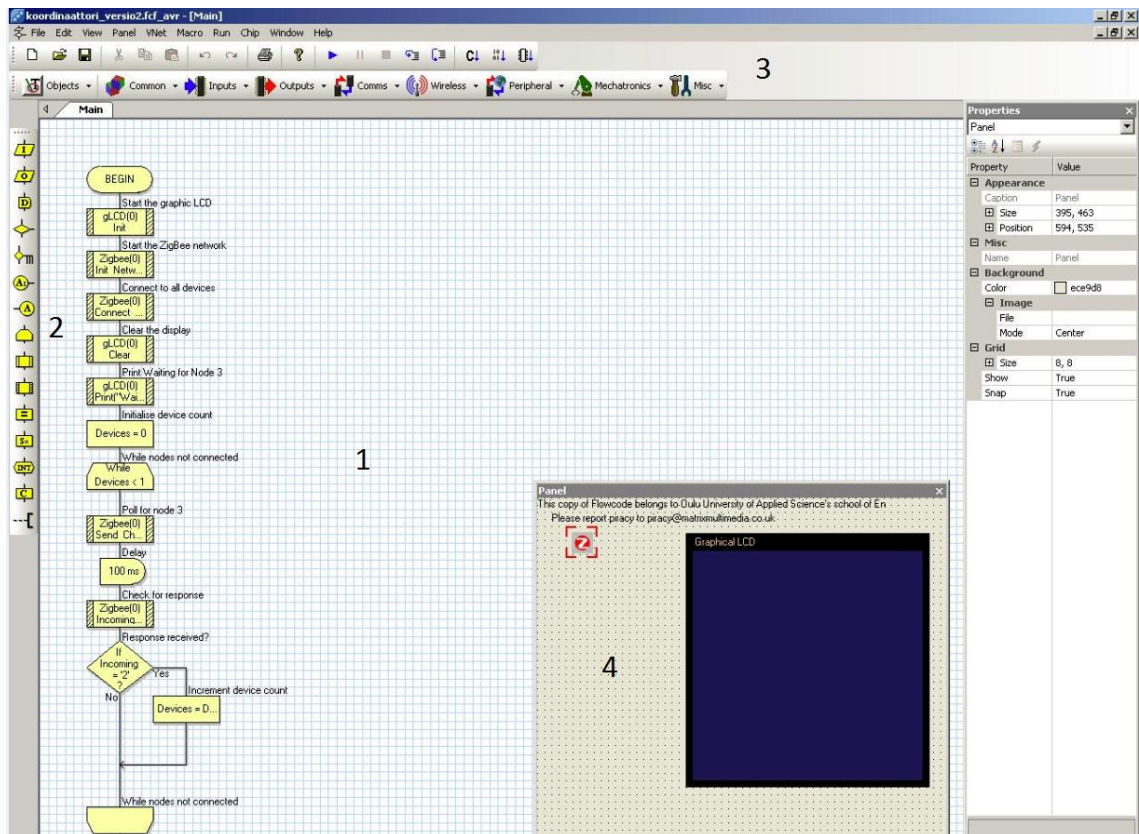


KUVA 3. Graafinen LCD-näyttö (5)

Edellä esitelyjen korttien lisäksi kokoonpanoihin kuului ZigBee-kortit, joilla anturitiedot saatiin siirrettyä langattomasti alustalta toiselle. Näistä korteista ja ZigBee-tekniikasta kerrotaan enemmän luvussa 3 (3.5).

2.3 Flowcode-ohjelmointityökalu

E-blocks-tuotteita ohjelmoidaan Flowcode-ohjelmalla. Flowcoden (kuva 4) etuna muihin ohjelmointityökaluihin on, että ohjelmointi toteutetaan monimutkaisten tekstirivien kirjoittamisen sijaan havainnollisilla vuokaavioilla. Valmiin ohjelman käänövaiheessa Flowcode muuttaa sen automaattisesti C-kielelle ja mikrokontrollerin ymmärtämälle Assembly-konekielelle. Flowcode on siis helposti lähestyttävä ohjelmointityökalu ja tutustumisen arvoinen varsinkin ohjelmoinnista kiinnostuneille henkilöille, joilla ei ole aiempaa kokemusta ohjelmoinnista.



KUVA 4. Flowcode-ohjelmointityökalun käyttöliittymä

Tässä työssä käytettiin uusinta Flowcoden versiota 4, joka tuki AVR-mikrokontrolleriarkkitehtuuria. Muita arkkitehtuureita ovat ARM (Advanced RISC Machine)- ja PIC (Peripheral Interface Controller) -arkkitehtuurit.

Kuvaa 4 mukailleen ohjelmointityökalun käyttöliittymä koostuu seuraavista keskeisistä osista:

1. Alue, johon vuokaavio-ohjelma muodostetaan.
2. Vasemmalla laidalla sijaitsevat kuvakkeet, joilla ohjelma rakennetaan. Niihin kuuluvat mm. ehtorakenteet, komponenttimakrot ja luupit.
3. Yläriveiltä löytyvät laitteet ja yhteydet, joita voidaan hyödyntää. Näitä ovat esim. LCD-näyttö ja ZigBee-yhteys. Yläriviltä hoituu myös ohjelman simulointi ja siirto mikrokontrolleriin.
4. Ohjelmassa hyödynnettävät laitteet ja yhteydet näkyvät tässä ikkunassa.

3 ZIGBEE

ZigBee on vuonna 2004 kehitetty lyhyen kantaman langaton tiedonsiirtotekniikka, joka perustuu IEEE (Institute for Electrical and Electronics Engineers) 802.15.4 -standardiin. Standardi määrittelee verkon fyysisen ja siirtoyhteyskerroksen, ja se puolestaan perustuu 802.15-standardiin (Wireless Personal Area Network). Sen päätavoitteena on kehittää edullisia, pienitehoisia ja pienikokoisia verkkolaitteita. (6.)

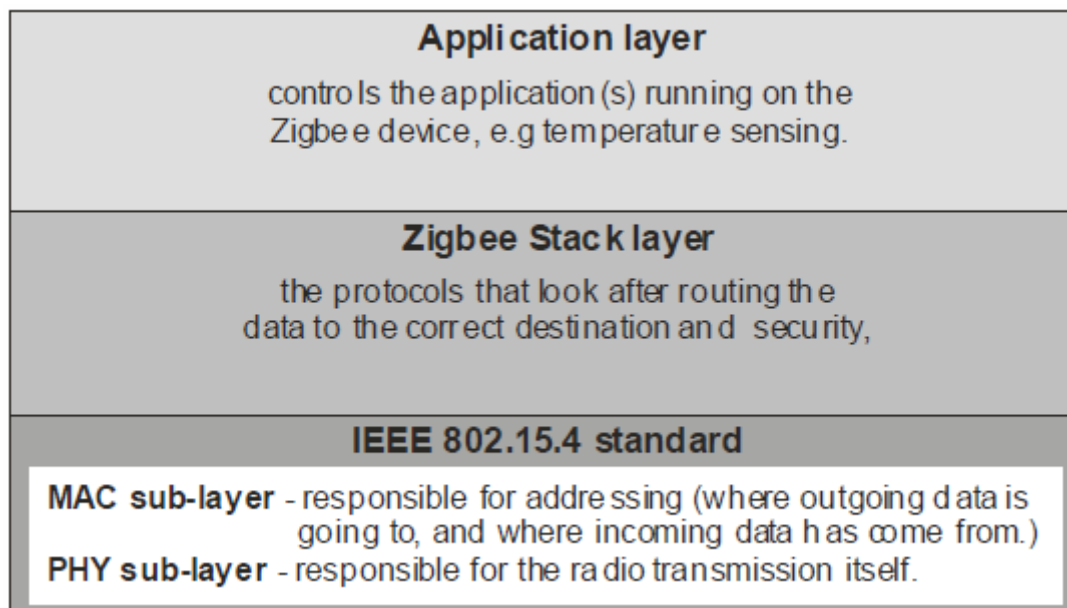
3.1 Tekniset tiedot

ZigBeen yleisin toimintataajuus on 2,4 GHz, jonka lisäksi Amerikan markkinoille taajuus on asetettu 915 MHz:iin. Euroopan ja Japanin alueilla ZigBee-laitteet voivat toimia yleisen arvon lisäksi 868 MHz:n taajuudella. Taajuudella 2,4 GHz tietoa voidaan siirtää 250 kbit/s:n nopeudella. Alhaisempien taajuuksien tiedonsiirtonopeudet asettuvat arvoihin 40 kbit/s ja 20 kbit/s. ZigBee-verkko voi sisältää 256 laitetta, ja kahden laitteen välinen toimintaetäisyys voi vaihdella kymmenen ja sadan metrin välillä. (7, s. 7–10.)

Tiedon välittämiseen käytetään DSSS (Direct Sequence Spread Spectrum) -tekniikkaa, jossa lähetettävään tietoon lisätään bittejä, jotta lähetettävä signaali saadaan levitettyä käytössä olevalle taajuusalueelle. Modulointimenetelmänä käytetään digitaalista O-QPSK (Offset Quadrature Phase Shift Keying) -modulaatiota, jossa kantoaaltoa käsitellään siirtotielle sopivaksi muuttamalla sen vaihetta. Modulaatiossa lähetykseen käytetään neljää erilaista vaiheen arvoa ja vaihetta muutetaan kerrallaan 90 astetta. Näin minimoidaan muutokset vastaanottopään amplitudissa. (7, s. 10–11.)

MAC (Medium Access Control) -kerroksen tehtävänä on määrittää, kuinka useat samalla alueella toimivat 802.15.4-radiot jakavat ilmarajapinnan. Se käyttää CSMA-CA (Carrier Sense Multiple Access with Collision Avoidance) -menetelmää kontrolloidakseen tietoa siirtävää ZigBee-laitetta (node). Näin vältetään yhteentörmäyksiltä, kun laitteet lähettävät tietoa samaan aikaan. Tiedonsiirrossa suurin paketin koko on 127 tavua. (7, s. 10.)

Kuvassa 5 on esitetty ZigBee-protokollapino, joka koostuu kolmesta osasta. Alimpana sijaitsee edellä esitelty 802.15.4-standardi, joka OSI-mallin mukaisesti sisältää fyysisen ja siirtoyhteyskerroksen. Keskellä sijaitsee kerros, joka huolehtii tiedon reitityksestä oikeaan osoitteeseen. Kerros huolehtii myös turvallisuudesta. Pinon huipulla on sovelluskerros, joka kontrolloi laitteissa toimivia sovelluksia. (7, s. 9.)



KUVA 5. ZigBee-protokollapino (7)

3.2 Toimintamuodot

802.15.4-laitetyyppejä ovat FFD (Full Function Device) ja RFD (Reduced Function Device). FFD-laite toimii tyypillisesti verkon hallitsijana, jonka tehtäviin kuuluu esimerkiksi verkon muodostus. RFD-laite on puolestaan suoraan

vuorovaikutuksessa sovelluksiin ja antureihin. RFD on pienitehoisempi kuin FFD, ja se voi olla yhteydessä ainoastaan FFD-laitteeseen. (7, s. 12.)

ZigBee-protokollan verkkotasolla loogiset laitteet voidaan jakaa kolmeen eri tyyppiin, joihin kuuluvat koordinaattori, reititin ja päätelaite.

Koordinaattori on FFD-laite, joka on vastuussa verkon luomisesta, kanavien ja osoitteiden valinnasta sekä muiden laitteiden lisäämisestä verkkoon. ZigBee-verkossa voi olla ainoastaan yksi koordinaattori. Uudet laitteet voivat liittyä verkkoon milloin vain, kun koordinaattori on aktiivinen. Koordinaattorin ollessa suljettuna uudet laitteet eivät voi liittyä verkkoon, mutta siellä jo olevat ja osoitteen omaavat laitteet pysyvät toiminnassa. (7, s. 13.)

Reititin on FFD-laite, joka on vastuussa pakettien puskuroinnista ja signaalien reitityksestä laitteiden välillä. Reitittimen avulla voidaan laajentaa verkkoa, ja se mahdollistaa verkkoon liittymisen tilanteessa, jossa koordinaattori on kytketty pois päältä. (7, s. 13.)

Päätelaitteena voi toimia sekä FFD-, että RFD-laite. Sillä voidaan kommunikoida verkossa sekä verkon ulkopuolella. Päätelaitteet ovat pienitehoisia, ja ne voidaan asettaa lepotilaan, joka vähentää edelleen tehon kulutusta. (7, s. 13.)

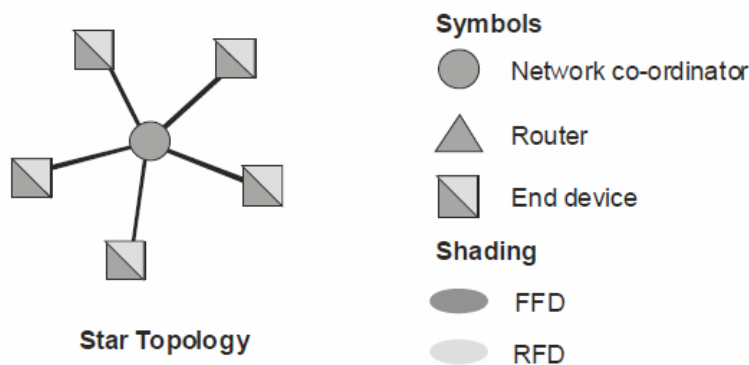
3.3 Topologiat

Verkkotopologia on tietokoneverkon perusrakenne eli tapa, jolla verkon laitteet liittyvät toisiinsa. ZigBee-tekniikassa käytetään kolmea erilaista topologiaa, joita ovat tähti-, Cluster tree- ja Mesh-topologia.

3.3.1 Tähtitopologia

Tähtitopologian (kuva 6) mukainen verkko koostuu yhdestä koordinaattorista ja yhdestä tai useammasta päätelaitteesta. Kaikki laitteet kommunikoivat ainoastaan koordinaattorin kanssa, joten tiedonsiirrossa päätelaitteelta lähetetty tieto

kulkee määränpään koordinaattorin kautta. Tähtitopologia on single-hop-tyyppinen, eli kaikki tiedonsiirrot kulkevat yhden koordinaattorin kautta. Vaihtoehtoisia reittejä ei siis ole. Tästä seuraa se, että tähtiverkon luotettavuus kärsii. (7, s. 14.)

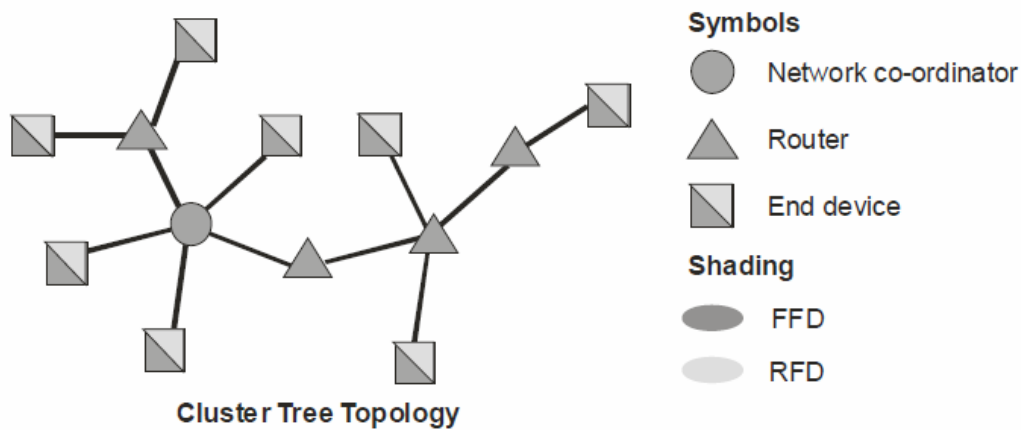


KUVA 6. Tähtitopologia (7)

3.3.2 Cluster tree -topologia

Cluster tree -topologian tapauksessa koordinaattori on yhdistetty useisiin reitittimiin ja päätelaitteisiin. Näin ollen verkkoa voidaan laajentaa merkittävästi, koska reitittimiä käytettäessä liittyvien päätelaitteiden määrää voidaan kasvattaa huomattavasti. Lisäksi reitittimillä verkon fyysistä kantamaa voidaan kasvattaa niin, että päätelaitteen ei tarvitse olla koordinaattorin kanssa kuuluvuusalueella toimiakseen verkossa. Reititin toimii tällöin eräänlaisena toistimena, jolloin päätelaitteen ja koordinaattorin välinen yhteys saadaan pidettyä yllä. (7, s. 15.)

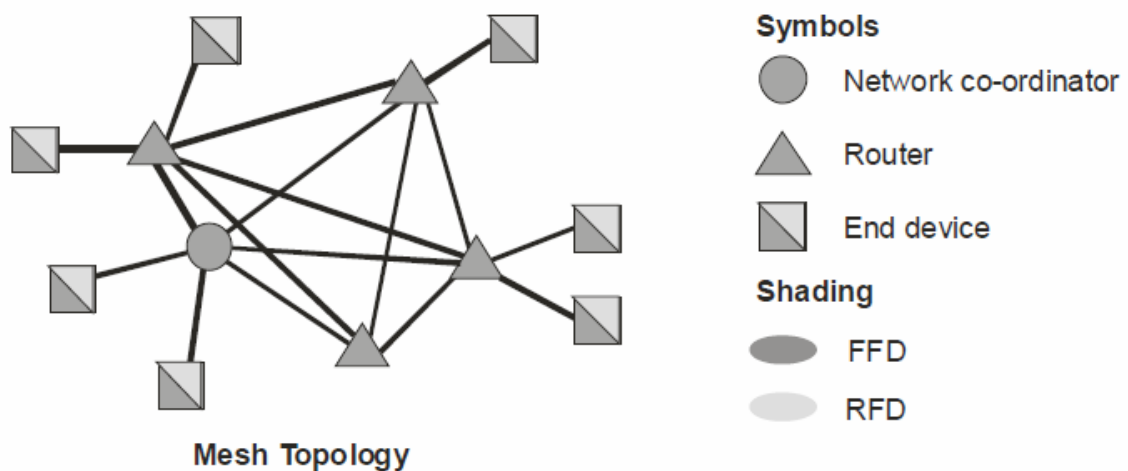
Cluster tree -topologia (kuva 7) on multi-hop-tyyppinen, jossa tiedot välitetään joko reitittimen tai koordinaattorin kautta. Tiedon välittämiseen voi löytyä useita mahdollisia reittejä, joista reititin valitsee sopivimman vaihtoehdon. (7, s. 15.)



KUVA 7. Cluster tree -topologia (7)

3.3.3 Mesh-topologia

Mesh-topologia (kuva 8) on muunnos edellä käsitellystä Cluster tree -topologiasta. Muunnoksessa verkon koordinaattori ja kaikki reitittimet ovat yhdistettynä suoraan toisiinsa. Näin ollen kommunikointi on vaivatonta. RFD-päätelaitteet eivät edelleenkään pysty kommunikoimaan suoraan keskenään, mutta FFD-päätelaitteiden kesken kommunikointi onnistuu, jolloin tree-rakennetta ei tarvitse noudattaa. Mesh-topologian etuna voidaan pitää pieniä viiveitä, joita voidaan saavuttaa reittien optimoinneilla. Etuna Cluster tree -topologian tavoin voidaan myös pitää luotettavuutta, joka taataan useilla siirrettävän tiedon reititysvaihtoehtoilla. (7, s. 15.)



KUVA 8. Mesh-topologia (7)

3.4 Tietoturva

ZigBee-verkot ovat turvallisia käyttää. Niiden tietoturva koostuu menetelmistä, joiden ansiosta vältytään hakkereiden ja naapuriverkkojen tunkeutumisilta. Keskeisenä menetelmänä toimii pääsyylista (Access Control List), jonka vaikutuksesta vain ennalta tunnetut laitteet voivat liittyä verkkoon. Verkon koordinaattori sisältää listan sallitujen laitteiden MAC-osoitteista. (7, s. 22.)

Pääsyylistojen lisäksi toimivana turvatoimena voidaan myös pitää Message Freshness Timer -menetelmää, jossa aikarajan ylittäneet vanhentuneet viestit torjutaan. Menetelmällä siten ehkäistään nauhoitushyökkäykset. Kyseisiä hyökkäyksiä syntyy, kun hyökkääjä nauhoittaa lähetyksen ja uusii sen myöhemmin. Esimerkkinä tilanne, jossa hyökkääjä nauhoittaa oven lukituskoodin päästäkseen sisälle sopivana ajankohtana. (7, s. 22.)

Tiedon salaukseen käytetään 128-bittistä AES (Advanced Encryption Standard) -salausmenetelmää. AES-algoritmit ovat sisäänrakennettuina ZigBee-kortteihin, joiden kautta ne myös toimivat. Kun salaus on aktivoitu, koordinaattori käynnistyy käyttäen 128-bittistä salausavainta. Tällöin ainoastaan laitteet, joilla on sama avain, pystyvät kommunikoimaan verkossa. Verkkoon liittyvillä reitittimillä ja päätelaitteilla täytyy olla kyseinen salausavain. Jos avainta ei ole, täytyy verkkoon pyrkivän laitteen hankkia se salaamattomalla siirrolla, josta seuraa hetkellinen tietoturva-aukko. (7, s. 22.)

Salauksen ollessa aktivoituna kaikki lähetettävä tieto siis salataan 16-tavuisella salausavaimella. Vastaanottopäässä kaikki salaamaton tieto hylätään. AES-salaus on siis tehokas osa tietoturvaa, mutta huonojakin puolia siinä on. Multi-hop-tyyppisessä tiedonsiirrossa reitin varrella jokaisen reitittimen kohdalla joudutaan purkamaan salaus ja sitten taas salaamaan tieto uudelleen. Seurauksena siitä on viiveen kasvu. Tämän lisäksi lähetettävien pakettien koko kasvaa. (7, s. 22.)

3.5 ZigBee-kortti

Työssä keskeinen ZigBee-yhteys toteutettiin kahdella ZigBee-kortilla (kuva 9), joista ensimmäinen toimi koordinaattorina ja toinen päätelaitteena. Toimiessaan koordinaattorina kortti pystyy hallinnoimaan kahdeksaa laitetta samanaikaisesti (reitittimet ja päätelaitteet). ZigBee-kortin keskeisimpiä yksityiskohtia ovat XBEE ZigBee -moduuli, joka vastaa yhteyden muodostuksesta, sekä merkintä kortin oikeassa ylä laidassa, josta selviää toimintarooli. Roolia havainnollistamassa on LED-valo. Kortin toimiessa koordinaattorina valo vilkkuu 2 Hz:n taajuudella, kun taas päätelaitteena vilkkuminen tapahtuu 1 Hz:n tahtiin. (8, s. 3–5, 7, s. 23.)

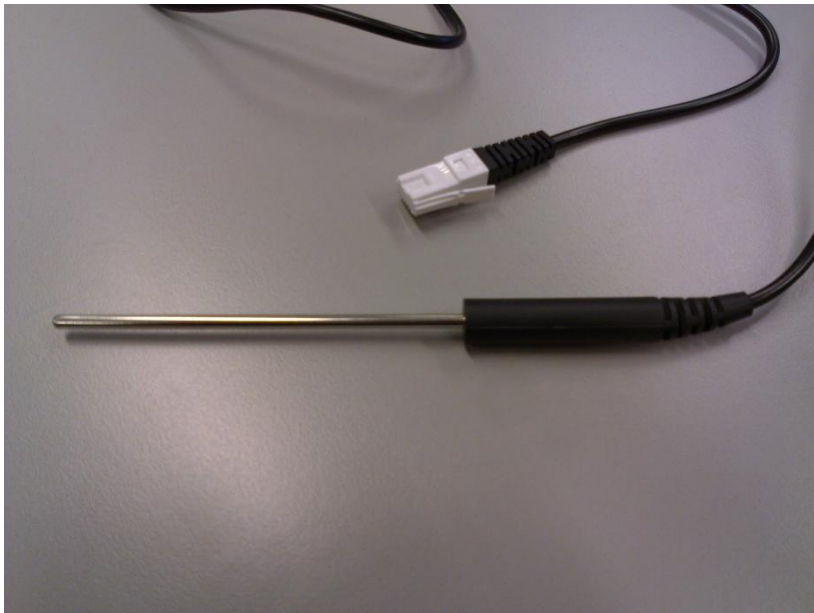


KUVA 9. ZigBee-kortti (8)

4 ANTURIN KÄYTTÖNOTTO

4.1 Lämpöanturi

Käyttöönoton kohteena oli Vernierin valmistama analoginen lämpöanturi (kuva 10). Sen mittapää on valmistettu ruostumattomasta teräksestä ja se on tarkoitettu käytettäväksi yleisesti laboratorioissa. Lämpötilan mittauksessa anturi lähettää sensorikortille tai muulle vastaanottimelle termistorin resistanssiarvoja. Lämpöanturi toimii NTC (Negative Temperature Coefficient) -termistorina, eli resistanssi pienenee, kun lämpötila kasvaa. (9, s. 1–3.)

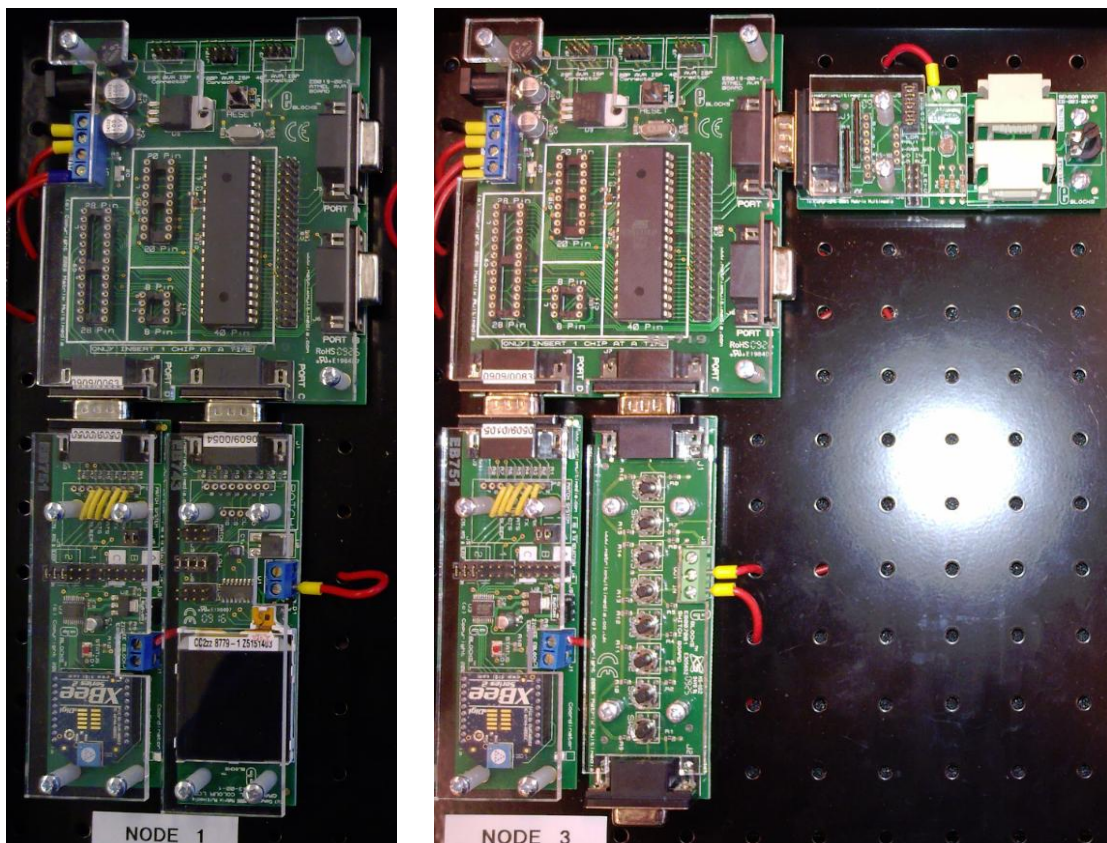


KUVA 10. Lämpöanturi

Lämpötilan mittaus voidaan suorittaa -40 :n ja $+135$ celsiusasteen välillä. Mittaustarkkuus 0 asteessa on $\pm 0,2$ °C ja $+100$ asteessa $\pm 0,5$ °C. Kyseiset arvot pätevät vain mittauspäähän. Muoviosa ei kestä kuumuutta eikä ole vedenkestävä. (9, s. 1–2.)

4.2 Käytettävä kehitysympäristö

Anturin käyttöönotto tapahtui ympäristössä, johon kuului node 1 (koordinaattori) ja node 3 (päälaite). Ympäristö (kuva 11) rakentui edellä esitellyistä E-blocks-rakennussarjaan kuuluvista ohjelmointiyksiköistä ja laajennuskorteista. Node 1 koostui mikrokontrollerikortista, ZigBee-kortista ja graafisesta LCD-näytöstä. Node 3 koostui mikrokontrollerikortin ja ZigBee-kortin lisäksi sensorikortista ja lämpöanturista. Kuvassa näkyvä painonappikortti ei liittynyt työhön mitenkään. Työn suorittamiseen olisi voinut käyttää node 1:n lisäksi node 2:ta, mutta tällöin sekä koordinaattori että päälaite olisivat olleet asennettuina samalle peltialustalle. Ajatus käyttää erillisiä alustoja tuntui selkeämmältä, joten node 2:n sijasta päätin käyttää node 3:a.

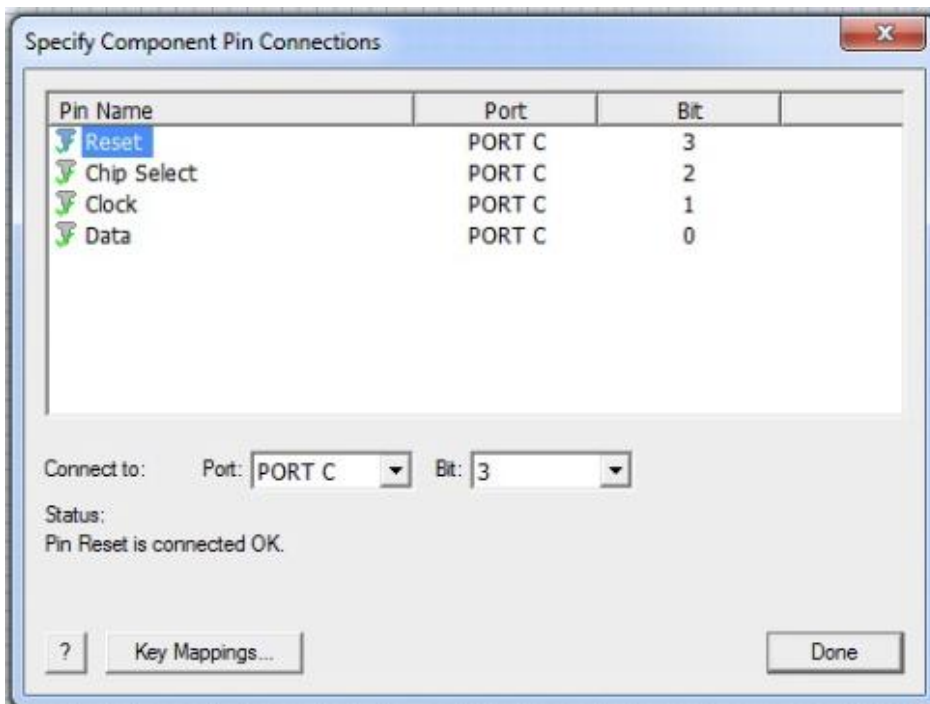


KUVA 11. Node 1 ja 3

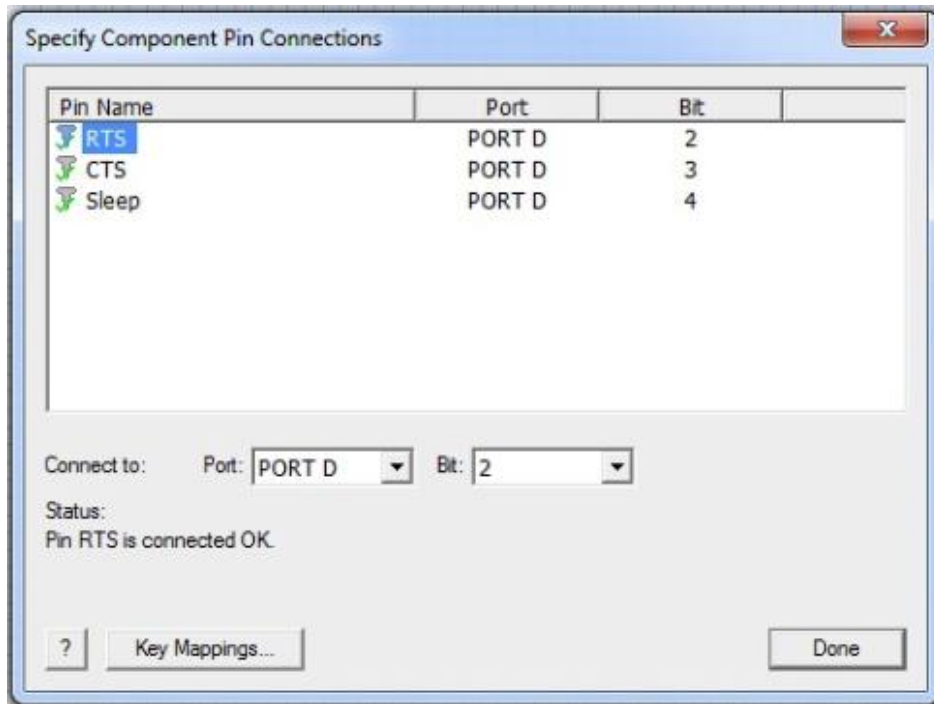
4.3 Alkuasetukset

Ennen varsinaista käyttöönottoa piti tehdä muutamia alkuasetuksia komponentteja ja ZigBee-yhteyttä varten. Ensimmäiseksi valittiin mikrokontrollerityypiksi Atmega324P, kun Flowcode-ohjelmointityökalussa aloitettiin uuden ohjelman tekeminen. Koordinaattoriin kuului graafinen LCD-näyttö, jonka asetuksissa (kuva 12) käytettäväksi portiksi valittiin C.

ZigBee-yhteyden muodostamiseksi sekä koordinaattoriin että päätelaitteeseen lisättiin ZigBee-komponentit, joiden toimintaportit ja bittijärjestykset asetettiin kuvan 13 mukaisesti. Portiksi valittiin siis D. Kytkenät asetettiin niin, että RTS (Request to Send) vastasi bittiä 2, CTS (Clear to Send) vastasi bittiä 3 ja Sleep vastasi bittiä 4.

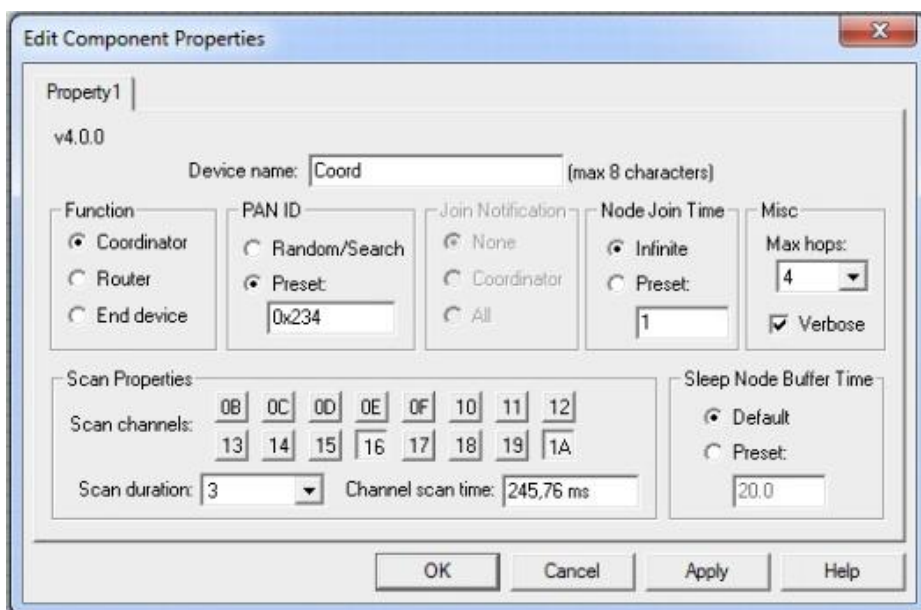


KUVA 12. LCD-näytön portin fyysisiä asetuksia

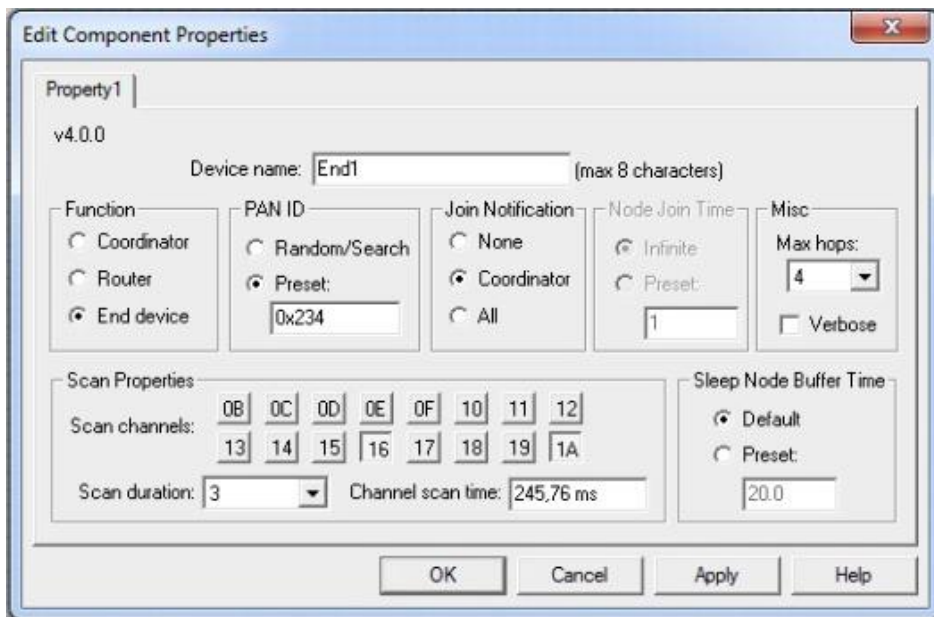


KUVA 13. ZigBee-komponentin fyysisiä asetuksia

Näiden asetusten lisäksi ZigBee-komponenteilla oli myös koordinaattori- ja päätelaittekohtaiset asetukset. Koordinaattorin ja päätelaitteen ZigBee-asetukset näkyvät kuvissa 14 ja 15.



KUVA 14. Koordinaattorin ZigBee-asetukset



KUVA 15. Päätelaitteen ZigBee-asetukset

ZigBee-asetukset-ikkuna koostuu erilaisista osista. Device name -kohtaan kirjoitetaan laitteen nimi. Function-kohdassa määritetään, toimiiko laite koordinaattorina, reitittimenä vai päätelaitteena. PAN ID tarkoittaa verkon tunnusta. Se voidaan määrittää itse tai käyttää kuvissa valittuna olevaa esiasetusta. Join Notification -kohdassa määritellään ilmoittamisen tarve, kun liitytään verkkoon. Esimerkkinä on kuva 15, jossa päätelaitteen roolissa oleva laite joutuu ilmoittamaan koordinaattorille verkkoon liittymisestä. (7, s. 31.)

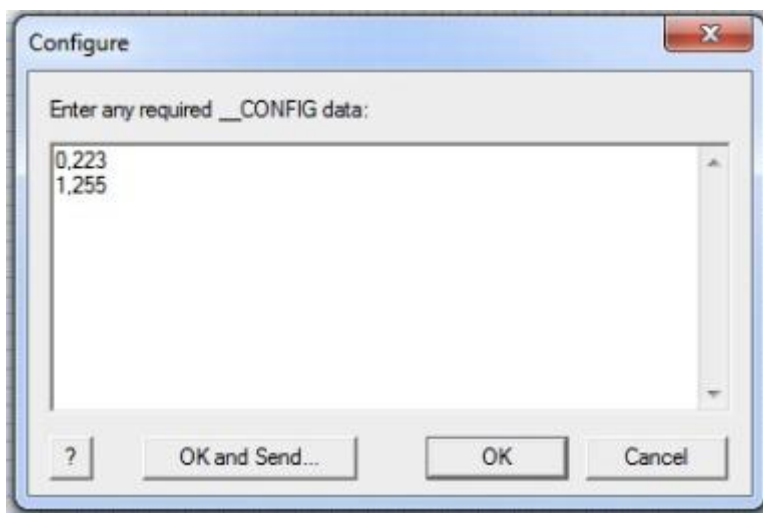
Laitteen toimiessa koordinaattorina voidaan Node Join Time -kohdassa määrittää aika, jolloin päätelaitteet voivat liittyä verkkoon. Misc määrää ylärajan toistettavien lähetysten määrälle. Scan Properties -kohdassa asetetaan aktiiviseksi kanavat, joita skannataan, kun etsitään kommunikoimaan pyrkiviä laitteita. Tässä kohdassa myös asetetaan skannauksen kesto. Sleep Node Buffer Time asettaa ajan, jolloin koordinaattori puskuroida dataa ja päätelaite asettuu lepotilaan. (7, s. 31.)

Onnistuneen laitteen välisen yhteyden muodostuksen kannalta on erityisen tärkeää, että PAN ID -tunnus on kummassakin päässä sama ja että Scan Properties -kohdan skannattavat kanavat ovat myös samat.

Kaikkien edellisten alkuasetusten lisäksi ennen ohjelmien kääntämistä ja siirtoa mikrokontrollerille piti tehdä kuvan 16 mukainen piirin konfiguroiminen, jossa arvot 0,223 ja 1,255 kirjoitettiin tekstikenttään. Lopuksi painettiin OK and Send -painiketta, jolloin arvot siirtyivät piirille. Lisäksi piti kääntäjän asetuksista (Chip > Compiler Options) määrittää polku ja parametrit oikeanlaisiksi. Asetukset, jotka tulivat Programmer/Ohjelmoija-kohtaan, olivat seuraavanlaiset:

Location/Polku: C:\Program Files\Atmel\AVR Tools\STK500\Stk500.exe

Parameters/Parametrit: -cUSB -d%t -e -if"%D%\%f.hex" -pf -vf



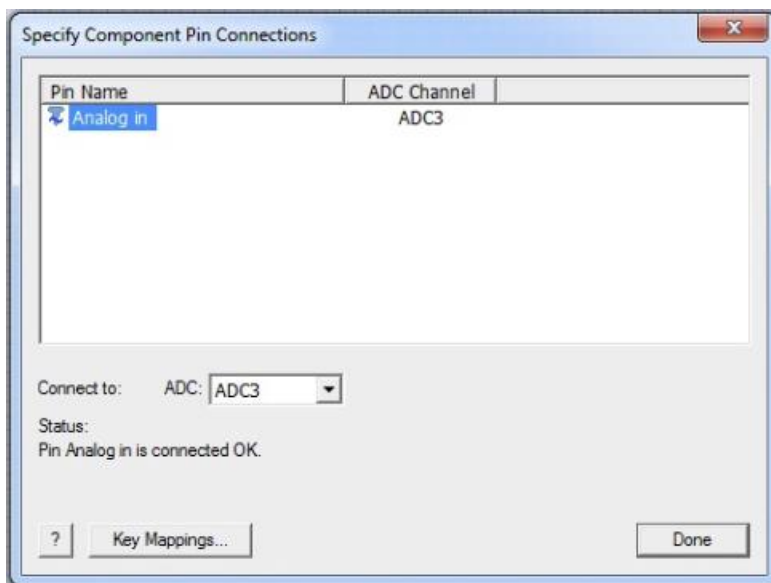
KUVA 16. Piirin konfigurointi

4.4 Lämpömittari ja ohjelmien muodostus

Käyttöönottossa oli tarkoituksena tehdä lämpömittarisovellus ja saada se toimimaan niin, että node 3:lta lähetetään lämpötila-arvoja langattomasti node 1:een, ja tulostetaan arvot celsiusasteina reaaliaikaisesti graafiselle LCD:lle. Toimiva ohjelmakokonaisuus koostui kahdesta osasta, joista toista käytettiin lämpötila-arvoja lähettävässä node 3:ssa eli ZigBeen kannalta päätelaitteessa, ja toista käytettiin lämpötila-arvoja vastaanottavassa node 1:ssä eli koordinaattorissa. Seuraavaksi esiteltävät vuokaavio-ohjelmat muodostuivat komponenttimakroista, ehtorakenteista, luupeista, viiveistä, ohjelmointi- ja laskentakomponenteistä sekä muuttujista.

4.4.1 Päätelaitte

Päätelaittepuolen ohjelman tekeminen aloitettiin lisäämällä paneeliin ZigBee-komponentti ja tekemällä siihen edellä esiteltyt päätelaitteen mukaiset asetukset. ZigBee-yhteyden lisäksi paneeliin lisättiin ADC (Analog to Digital Converter) -komponentti, jota tarvittiin anturilta tulevien resistanssiarvojen vastaanottamiseen. ADC-komponentin kanavaksi valittiin kuvan 17 mukaisesti kanava kolme (ADC3), joka oli tarkoitettu analogiselle tiedonsiirrolle.



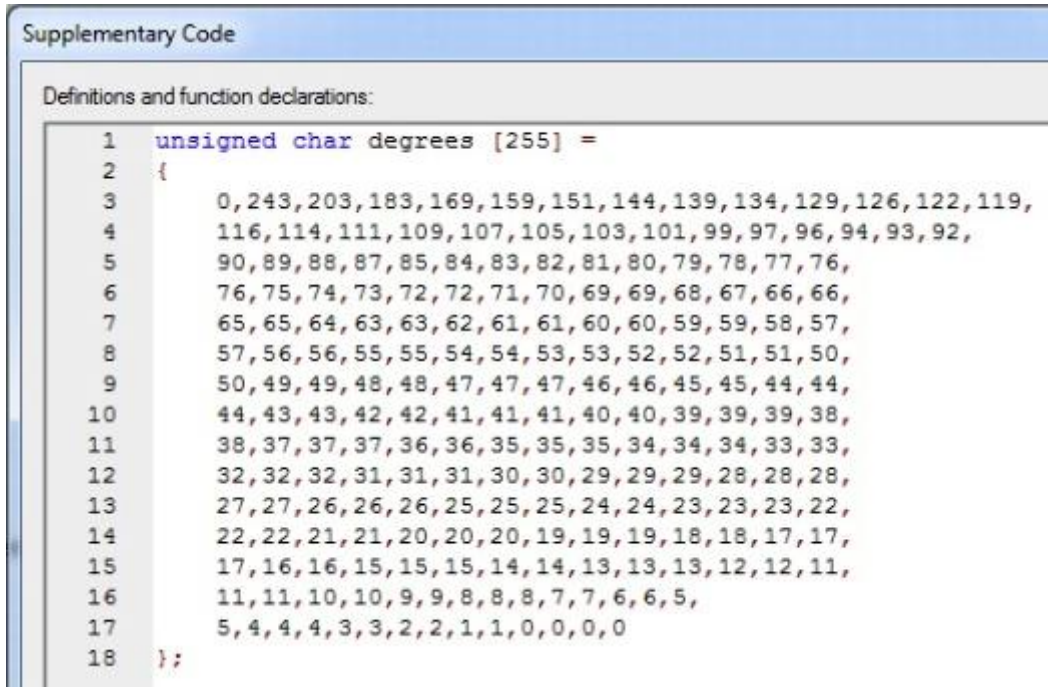
KUVA 17. ADC-komponentin asetukset

Seuraavaksi vuorossa oli rakentaa varsinainen vuokaavio-ohjelma. Ensimmäisenä piti aktivoida ZigBee-yhteys ja määrittää laite ottamaan yhteyttä koordinaattoriin. Sen jälkeen tehtiin yhteyden muodostuksen kannalta tärkeä rakenne, jossa koordinaattorille lähetettiin kiittauksena tiedonsiirrosta numero 2, jos koordinaattorilta oli onnistuneesti vastaanotettu numero 1.

Yhteyden muodostuksen jälkeen oli vuorossa pääohjelma, jossa tapahtui resistanssiarvojen vastaanotto. ADC-komponenttia hyödyntäen saapuva tavutyyppinen resistanssitiieto muunnettiin analogisesta digitaaliseksi. Digitaaliset resistanssiarvot muunnettiin edelleen varsinaisiksi celsiustyyppisiksi lämpötila-arvoiksi, eli tavutyyppisiksi Temp-nimisiksi celsiusasteiksi.

Resistanssiarvojen muunnos celsiusasteiksi tapahtui C-koodia apuna käyttäen. Koodi toimi niin, että se haki taulukosta (kuva 18) sen hetkistä resistanssiarvoa vastaavan celsiusasteen. Resistanssiarvot olivat nollan ja 255:n välillä, jolloin periaatteessa arvoa yksi vastasi +243 °C, ja arvoa 254 vastasi –67 °C. Taulukossa pienin arvo on nolla, joten sen perusteella kyseisellä taulukolla ei voida esittää kylmempiä arvoja kuin 0 °C.

Taulukon tekemiselle tarkoitettu tila ei löytynyt suoraan käyttöliittymän vasemmasta laidasta, vaan se piti erikseen valita projektin asetuksista. Tila löytyi seuraavien valikkojen kautta: View > Project Options > Supplementary Code.



```
Supplementary Code
Definitions and function declarations:
1 unsigned char degrees [255] =
2 {
3     0, 243, 203, 183, 169, 159, 151, 144, 139, 134, 129, 126, 122, 119,
4     116, 114, 111, 109, 107, 105, 103, 101, 99, 97, 96, 94, 93, 92,
5     90, 89, 88, 87, 85, 84, 83, 82, 81, 80, 79, 78, 77, 76,
6     76, 75, 74, 73, 72, 72, 71, 70, 69, 69, 68, 67, 66, 66,
7     65, 65, 64, 63, 63, 62, 61, 61, 60, 60, 59, 59, 58, 57,
8     57, 56, 56, 55, 55, 54, 54, 53, 53, 52, 52, 51, 51, 50,
9     50, 49, 49, 48, 48, 47, 47, 47, 46, 46, 45, 45, 44, 44,
10    44, 43, 43, 42, 42, 41, 41, 41, 40, 40, 39, 39, 39, 38,
11    38, 37, 37, 37, 36, 36, 35, 35, 35, 34, 34, 34, 33, 33,
12    32, 32, 32, 31, 31, 31, 30, 30, 29, 29, 29, 28, 28, 28,
13    27, 27, 26, 26, 26, 25, 25, 25, 24, 24, 23, 23, 23, 22,
14    22, 22, 21, 21, 20, 20, 20, 19, 19, 19, 18, 18, 17, 17,
15    17, 16, 16, 15, 15, 15, 14, 14, 13, 13, 13, 12, 12, 11,
16    11, 11, 10, 10, 9, 9, 8, 8, 8, 7, 7, 6, 6, 5,
17    5, 4, 4, 4, 3, 3, 2, 2, 1, 1, 0, 0, 0, 0
18 };
```

KUVA 18. Celsiusasteiden taulukko (12)

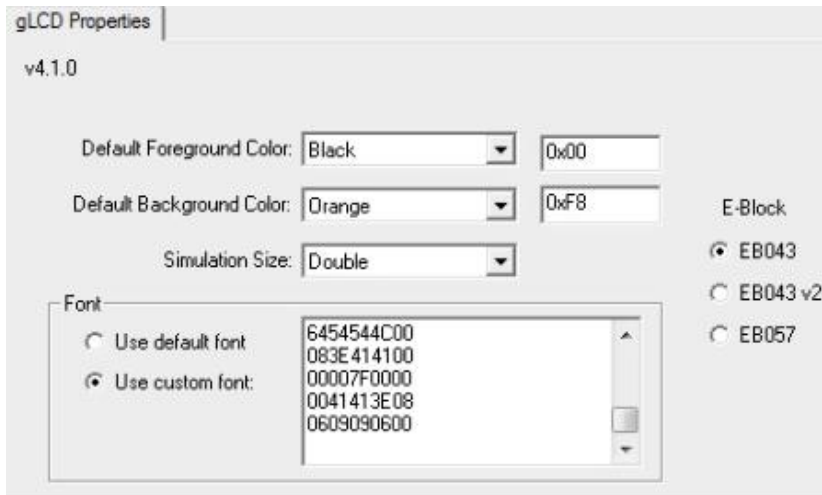
Taulukosta huomataan, että useita lämpötila-arvoja on enemmän kuin yksi. Esimerkiksi arvo 26 esiintyy taulokossa kolme kertaa. Tämä johtuu arvojen pyöristämisestä. Kyseinen celsiusarvo sijoittuu resistanssiarvojen 142 ja 144 välille, kun käytetään normaalia pyöristyssääntöä.

Vuokaavion loppuvaiheessa celsiusasteet lähetettiin langattomasti koordinaattorille sekunnin välein riippumatta siitä, oliko lämpötila muuttunut tuona aikana vai ei. Valmis päätelaitteen vuokaavio-ohjelma on liitteessä 1. Liitteessä näkyvät yksityiskohtaisesti asetukset ja muuttujat, joita vuokaaviossa käytettiin.

Päätelaitteen vuokaavion muodostuksessa käytettiin apuna lähdettä 10 (s. 16–19) ZigBee-yhteyttä koskeissa rakenteissa ja asetuksissa. Laskurakenteiden ja lämpöarvojen lähetysten tekemiseen hyödynnettiin lähdettä 11 (s. 42). Resistanssiarvojen vastaanotossa ja celsiusasteiden taulukon muodostuksessa hyödynnettiin lähdettä 12.

4.4.2 Koordinaattori

Koordinaattori toimi yhteyden luonnin ja ylläpitämisen lisäksi lämpötilatietojen vastaanottajana ja tulostajana. Ohjelman muodostaminen aloitettiin päätelaitteen tavoin lisäämällä paneeliin ZigBee-komponentti ja tekemällä siihen tarvittavat aikaisemmin esiteltyt asetukset. Sen lisäksi paneeliin lisättiin graafinen LCD-näyttö, johon aiemmin esiteltyjen asetusten lisäksi piti ohjelmoida asteen merkki, koska sitä ei alunperin ollut. Ohjelmointi tapahtui kuvan 19 mukaisesti valitsemalla asetukset "Use custom font", ja korvaamalla numerosarjojen alin rivi sarjalla 0609090600 (korvasi merkin ~). Samalla asetettiin näytön taustaväri oranssin sävyiseksi. Ohjeet astemerkin luomiseen löytyivät Matrix Multimedian foorumilta.



KUVA 19. Graafisen LCD-näytön asetusten muokkausta

Koordinaattorin vuokaavion rakentaminen aloitettiin aktivoimalla näyttö ja Zig-Bee-yhteys toimintavalmiuteen. Koordinaattori määritettiin ottamaan yhteyttä päätelaitteeseen ja tulostamaan englanninkielinen teksti "Waiting for Node 3", joka suomennettuna oli "Odottaa Node 3:a". Tekstin oli tarkoitus pysyä näytöllä, kunnes yhteys päätelaitteen kanssa oli muodostettu.

Seuraavaksi vuorossa oli laitteiden välisen yhteyden muodostus, joka toteutettiin luupissa. Yhteyden muodostamiseksi päätelaitteelle lähetettiin numero 1. Kun päätelaite vastaanotti numeron onnistuneesti, se lähetti kiitauksena koordinaattorille numeron 2. Tämä tarkoitti sitä, että yhteys toimi kumpaankin suuntaan. Onnistuneen yhteyden muodostuksen merkiksi näytölle tulostettiin teksti "Node 3 connected" eli "Node 3 yhdistetty".

Vuokaavion muodostus eteni niin, että seuraavaksi tehtiin pääohjelma, joka päätelaitteen tavoin toimi luupissa. Luupin alussa vastaanotettiin päätelaitteelta sekunnin välein langattomasti saapuneet Temp-nimiset celsiusastearvot, jonka jälkeen ohjelman suoritus eteni ehtorakenteeseen. Rakenteessa näytölle tulostettiin teksti "Temperature" eli suomennettuna "Lämpötila", ja sen lisäksi asteen merkki lämpötila-arvon perään. Lopuksi näytölle tulostettiin varsinainen lämpötila, joka päivittyi sekunnin välein.

Ohjelman loppuun tehtiin rakenne, joka tulosti näytölle ZigBee-yhteyden signaalin voimakkuuden. Näytön ylälaitaan tulostettiin teksti "Signal level" eli "Signaalin taso", jonka perään tulostui signaalin voimakkuutta havainnollistava luku.

Koordinaattorin vuokaaviossa kaikki muuttujat olivat tavutyyppejä, lukuun ottamatta signaalin voimakkuutta, joka oli string-tyyppinen. Koordinaattorin vuokaavio löytyy liitteestä 2, jossa näkyvät tarkasti kaikki käytetyt muuttujat, parametrit ja muut asetukset.

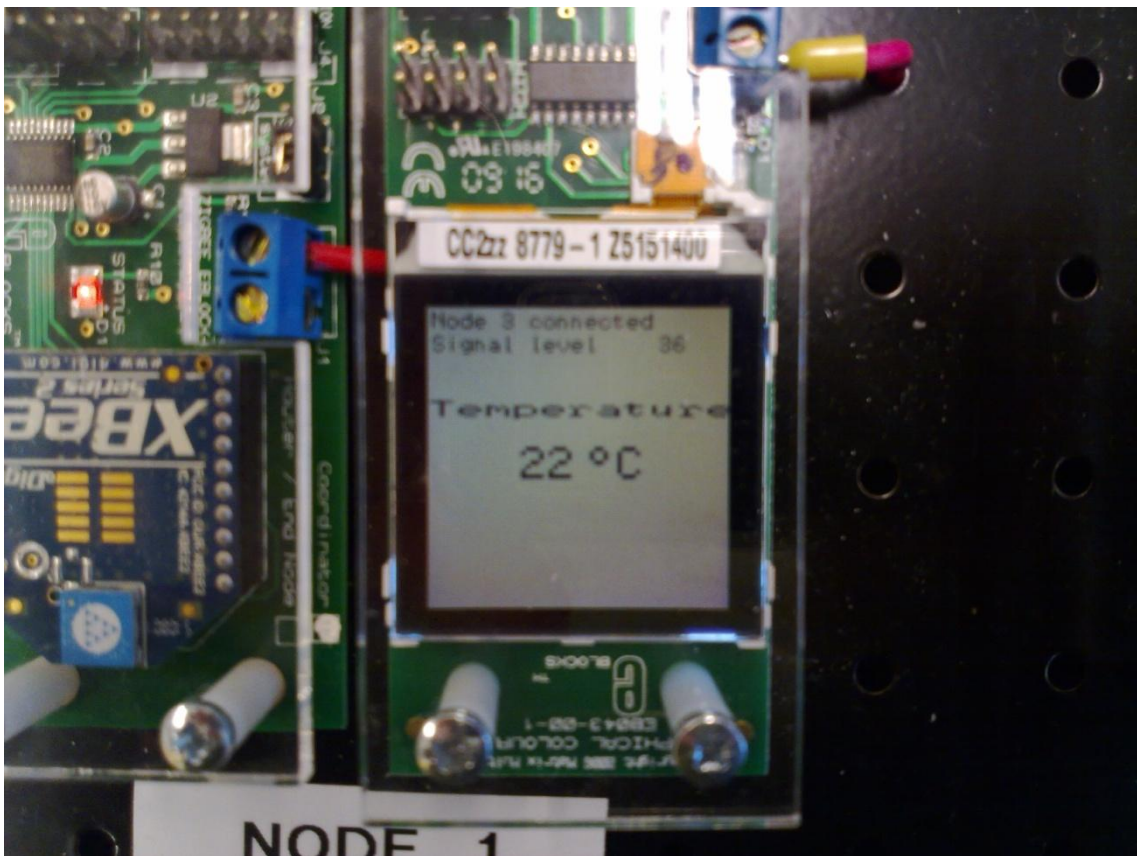
Koordinaattorin vuokaavion muodostuksessa hyödynnettiin lähettä 10 (s. 16–18) ZigBee-verkon muodostuksessa ja signaalin voimakkuuden tulostuksessa. Lähettä 11 (s. 38) hyödynnettiin laskurakenteissa sekä lämpötila-arvojen vastaanotossa ja tulostuksessa.

4.4.3 Valmis kokoonpano

E-blocks-tuotteet, Vernierin lämpöanturi sekä vuokaavio-ohjelmat muodostivat yhdessä kokoonpanon, jossa ilman lämpötila mitattiin, lähetettiin se langattomasti alustalta toiselle, ja tulostettiin arvo näytölle. Kokoonpanon aktivointi tapahtui niin, että ensiksi käännettiin ja siirrettiin vuokaavio koordinaattorille. Siirrossa ohjelmointiyksikön toinen pää asetettiin tietokoneen USB-porttiin, ja toinen pää mikrokontrollerikortin J3-liittimeen (40pin). Varsinainen ohjelman siirto tapahtui painamalla ohjelmointityökalun ylärivillä sijaitsevaa Compile to Chip (Käännä vuokaavio ja siirä prosessorille) -painiketta. Onnistuneen siirron merkiksi ohjelman siirron vaiheita kuvaavassa ikkunassa paluuarvoksi tulostui nolla.

Onnistuneen koordinaattorin ohjelman kääntämisen ja siirron jälkeen mikrokontrollerikortin reset-nappia painettiin kerran, jolloin ohjelman suoritus alkoi alusta. Ohjelman aktivoitumisen merkiksi koordinaattorin näytölle ilmestyi teksti "Waiting for Node 3", joka siis tarkoitti sitä, että se oli valmis muodostamaan yhteyden node 3:n eli päätelaitteen kanssa.

Päätelaitteen vuokaavio-ohjelman kääntäminen ja siirtäminen mikrokontrollerille tapahtui koordinaattorin tavoin. Reset-napin painamisen seurauksena päätelaitteen ohjelman suoritus alkoi. Onnistuneen laitteiden välisen yhteyden muodostuksen jälkeen koordinaattorin näytön ylälaitaan tulostui teksti "Node 3 connected", ja sen alapuolelle teksti "Signal level" numeroarvoineen havainnollistamaan ZigBee-yhteyden välisen signaalin voimakkuutta. Näytön keskikohtaan tulostui teksti "Temperature", ja sen alapuolelle varsinainen lämpötila, joka päivittyi sekunnin välein. Mikäli yhteys ei muodostunut ensimmäisen node 3:n mikrokontrollerin reset-napin painalluksen jälkeen, täytyi nappia painella tasaisin väliajoin useampaan kertaan. Toiminnassa olevan kokoonpanon lämpötilan tulostus on esitetty kuvassa 20.



KUVA 20. Lämpömittari toiminnassa

Toiminnassa oleva lämpömittarisovellus toimi halutunlaisesti. Lämpötilaa mitattiin sisätilassa, jolloin lämpötila vaihteli +21 ja +22 celsiusasteen välillä. Li-

säksi anturilla mitattiin ruumiin lämpöä asettamalla anturi kainaloon. Tulokseksi saatiin lämpötila, joka vaihteli +35:n ja +36 °C:n välillä. Mitatut arvot vaikuttivat siis järkeviltä, joten voitiin todeta sovelluksen toimivan oikealla tavalla. Myöhemässä vaiheessa luotettavuuden ja mielenkiinnon takia mittauksia laajennettiin niin, että mitattavien kohteiden lämpötilaeron tuli olla mahdollisimman suuri. Mittausten kohteiksi valittiin kiehuva ja sulamispisteessä oleva vesi. Kiehuvan veden mittauksessa vedenkeittimellä lämmitettiin vesi kiehumispisteeseen. Kun vesi kiehui, asetettiin anturi välittömästi veteen, jottei se ehtisi viiletä. Lämpötila kohosi nopeasti lukemaan +101 °C, jossa se pysyi hetken. Kiehuvan veden mittaus onnistui siis odotetunlaisesti. Huomiona mainittakoon, että kun lämpötila kohosi kolminumeroiseen lukemaan, siihen ilmestyi ylimääräinen viiva. Häiriö ei hävinnyt näytöltä, vaikka lukema palasi takaisin kaksinumeroiseen arvoon. Sulamispisteessä olevan veden mittaus suoritettiin asettamalla lämpöanturi jääveeseen. Hetkellisesti lukemaksi saatiin +2 °C. Jäävesi ei luultavasti ollut tarpeeksi kylmää, joten sulamispisteestä jäätiin kahden asteen verran.

5 YHTEENVETO JA POHDINTA

Työn tarkoituksena oli ottaa käyttöön Vernierin valmistama analoginen lämpöanturi. Lopputuloksena syntyi lämpömittarisovellus, jossa lämpötila lähetettiin langattomasti node 3:lta node 1:lle, jossa se havainnollistettiin tulostamalla lämpötila graafiselle LCD-näytölle celsiusasteina.

Lämpömittarisovelluksen kokoonpano koostui lämpöanturin lisäksi node 1:stä ja node 3:sta, joihin kumpaankin kuului mikrokontrollerikortti. Node 1:ssä oli lisäksi graafinen LCD-näyttö, ja node 3:ssa sensorikortti. Lisäksi kumpikin node sisälsi ZigBee-kortit, joilla mahdollistettiin laitteiden välinen langaton tiedonsiirto. Sovelluksen varsinaisen toimivuuden mahdollisti Flowcode-ohjelmointityökalu, jolla tehtiin yksilölliset vuokaavio-ohjelmat kummankin noden mikrokontrollerin suoritettavaksi.

Mittauksissa anturilla mitattiin kohteita, jotka ulottuivat veden sulamispisteestä kiehumispisteeseen. Mittaustulosten perusteella voitiin todeta lämpömittarin mittaavan luotettavia ja oikeanlaisia lämpötila-arvoja.

Työ onnistui pääpiirteissään hyvin, mutta ongelmiakin oli. Työn tekemisen pääpaino oli ohjelmoinnissa, joka tuotti vaikeuksia, koska ohjelmointitaitoni olivat alkeiden tasolla. Aloittelijaystävällinen Flowcode-ohjelmointityökalu ja muutamat esimerkkiohjelmat ja avut kuitenkin mahdollistivat työn tekemisen. Valmiin kokoonpanon kanssa oli myös muutamia ongelmia. Laitteiden välisen ZigBee-yhteyden muodostus tuntui välillä olevan mahdottomuus, ja kun yhteys saatiin muodostettua, ei lämpömittarisovellus välttämättä lähtenyt toimimaan oikealla tavalla. Lisäksi signaalin voimakkuuden tulostuksessa oli vikaa, vaikka se osa vuokaaviosta tehtiin ohjeiden mukaan (10, s. 18). Arvoitukseksi jäi, johtuivatko nämä ongelmat vääränlaisista rakenteista vuokaavioissa vai vioista itse laitteissa.

Oppimisen kannalta työ oli melko haastava, koska teoriaa oli niukasti saatavilla. E-blocks-tuotteiden datalehdet olivat yleisesti ottaen pelkistettyjä eli niissä ei ollut kovinkaan syvällistä tietoa tarjolla. Erityisesti yllätyksenä tuli vuokaavio-ohjelmointi, josta ei sähköisessä muodossa eikä kirjana löytynyt minkäänlaista ”tietopakettia”. Tämä johti siihen, että monet asiat jäivät arvailujen varaan. Helpotusta epätietoisuuteen tarjosi jonkin verran Matrix Multimedian foorumi, jossa yrityksen henkilökunta tai muut foorumin käyttäjät saattoivat vastata foorumilla esitettyyn kysymykseen. Vaikka teoriatieto oli edellä mainituissa asioissa hyvin rajallista, tuli ZigBee-standardi tutuksi. Mieleen palautui asioita, jotka olivat unohtuneet, ja sen lisäksi tietoutta laajensivat asiat, jotka koskivat ZigBee-laitteiden rooleja verkossa. Yhteyden muodostukseen ja tietoturvaan liittyvät asiat olivat myös mieleenpainuvia. Opinnäytetyötä kirjoitettaessa tekstin oikeinkirjoituksen taito parani huomattavasti.

Lämpöanturilla pystytään mittaamaan huomattavasti tarkempia arvoja kuin tämän työn kokoonpanolla. Luontevaa olisikin, että mahdollisessa jatkokehityksessä ohjelmaa muokattaisiin niin, että lämpötiloja voitaisiin tulostaa desimaalin tarkkuudella. Lisäksi voitaisiin keskittyä ääriarvojen ja yleensäkin laajamittaisemmin lämpötilojen mittaamiseen. Työ tehtiin hyvinvointipuolen opiskelijoita varten, jolloin sitä voidaan mahdollisesti hyödyntää heidän tulevissa laboratorioharjoituksissaan.

LÄHTEET

1. About. 2011. Matrix Multimedia. Saatavissa:
<http://www.matrixmultimedia.com/about.php>. Hakupäivä 6.9.2011.
2. E-blocks user guide. 2011. Matrix Multimedia. Saatavissa:
<http://www.matrixmultimedia.com/datasheets/EB355-80-2.pdf>. Hakupäivä 7.9.2011.
3. Atmel AVR multiprogrammer system. 2011. Matrix Multimedia. Saatavissa:
<http://www.matrixmultimedia.com/datasheets/EB194-30-2.pdf>. Hakupäivä 10.9.2011.
4. Sensor board datasheet. 2011. Matrix Multimedia. Saatavissa:
<http://www.matrixmultimedia.com/datasheets/EB003-30-2.pdf>. Hakupäivä 10.9.2011.
5. Graphical LCD Display Datasheet. 2011. Matrix Multimedia. Saatavissa:
<http://www.matrixmultimedia.com/resources/files/datasheets/EB043-30-1.pdf>.
Hakupäivä 12.9.2011.
6. IEEE 802.15.4. 2011. Wikipedia. Saatavissa:
http://en.wikipedia.org/wiki/IEEE_802.15.4. Hakupäivä 22.9.2011.
7. ZigBee Student Notes. 2009. Matrix Multimedia. Saatavissa:
<http://www.matrixmultimedia.com/LearningCentre/EB538-82-03.pdf>. Hakupäivä 22.9.2011.
8. Zigbee board datasheet. 2008. Matrix Multimedia. Saatavissa:
<http://www.matrixmultimedia.com/datasheets/EB051-30-1.pdf>. Hakupäivä 9.10.2011.

9. Stainless Steel Temperature Probe. 2011. Vernier Software & Technology. Saatavissa: <http://www.vernier.com/files/manuals/tmp-bta.pdf>. Hakupäivä 26.10.2011.
10. ZigBee Instructor Guide. 2009. Matrix Multimedia. Saatavissa: <http://www.matrixmultimedia.com/LearningCentre/EB538-82-03.pdf>. Hakupäivä 13.11.2011.
11. Manninen, Onni 2010. ZigBee-verkon luominen Matrix Multimedian E-Blocks-opetussarjalla. Tampere: Tampereen ammattikorkeakoulu, tietotekniikan yksikkö. Insinööriyö. Saatavissa: https://publications.theseus.fi/bitstream/handle/10024/12482/Manninen_Onni.pdf?sequence=2. Hakupäivä 13.11.2011.
12. Stainless Steel Temperature Probe HSTMP - Look Up Table Appr. 2011. Matrix Multimedia. Saatavissa: <http://www.matrixmultimedia.com/mmforums/viewtopic.php?f=26&t=7020#p22122>. Hakupäivä 13.11.2011.

