

Programmering av inbyggt system med trådlös kommunikation via ZigBee

Mälardalens Högskola, 2007-06-03
Institutionen för Datavetenskap och Elektronik, IDE

Student: Martin Granath
Uppdragsgivare: SecureU HB

Handledare IDE: Mika Seppänen
Handledare SecureU HB: Per Forsberg

Examinator: Mikael Ekström

Abstract

SecureU HB has developed a product prototype, designed to detect and prevent water damage caused by failing dishwashers. When a waterleak is detected by a sensor, a signal is sent on a wireless network to the mainunit, which cuts power and water to the machine. The wireless network technology used is based on IEEE802.15.4.

The purpose of this thesis was to do an analysis of the existing system, and apply modifications to both hard and software. A new prototype was built and new software was developed. Time was spent on microcontroller program development, design of schematics and attempts to reduce the cost of components. Wanted functionality was verified through testing, but a test during real conditions is still to be done.

Sammanfattning

SecureU HB har tagit fram en prototyp till en produkt, gjord för att upptäcka och förhindra vattenskador orsakade av trasiga diskmaskiner. Då ett vattenläckage upptäcks av en sensor skickas en signal via ett trådlöst nätverk till en huvudenhet som stänger av ström och vattentillförsel. Nätverkstekniken som används är baserad på IEEE 802.15.4.

Examensarbetet bestod av att göra en analys av befintligt system, samt att genomföra modifikationer av hård och mjukvara. Det medförde att en ny prototyp togs fram och en helt ny mjukvara utvecklades. En stor del av arbetet bestod av programmering av mikrokontrollers, design av elschemor samt att försöka reducera komponent kostnader. Efterfrågad funktionalitet verifierades genom testning, det återstår dock att genomföra ett test under verkliga förhållanden.

1 Bakgrund och syfte.....	4
2 Problembeskrivning	4
3 IEEE 802.15.4 / ZigBee	5
3.1.0 IEEE 802.15.4	5
3.2.0 ZigBee	6
3.3.0 Kretsar/Tillverkare	8
3.4.0 ZMR2A ZigBee modul	8
4 Analys av ursprunglig hårdvara.....	11
4.1.0 Styrenhet.....	11
4.2.0 Alarmenhet	14
4.3.0 Komponenter/Pris.....	15
5 Förslag till ny prototyp	16
5.1.0 Styrenhet.....	16
5.2.0 Alarmenhet	22
6 Programmering	23
6.1.0 Nätverk Coordinator.....	23
6.2.0 Nätverk Endpoint	25
6.3.0 Huvudprogram Coordinator	27
6.4.0 Huvudprogram Endpoint.....	32
6.5.0 Möjliga förbättringar	33
7 Resultat och slutsats	34
8 Komponentkostnad ny prototyp	35
9 PCB.....	38
10 Referenser	39
11 Förkortningar	39
12 Använd hård och mjukvara	40

1 Bakgrund och syfte

Det här jobbet görs åt företaget SecureU HB, de har tagit fram en första prototyp till en produkt. Viss funktionalitet i prototypen saknades, vilket de ville ha hjälp med att utveckla. De ville även ha hjälp med att försöka reducera komponentkostnaderna.

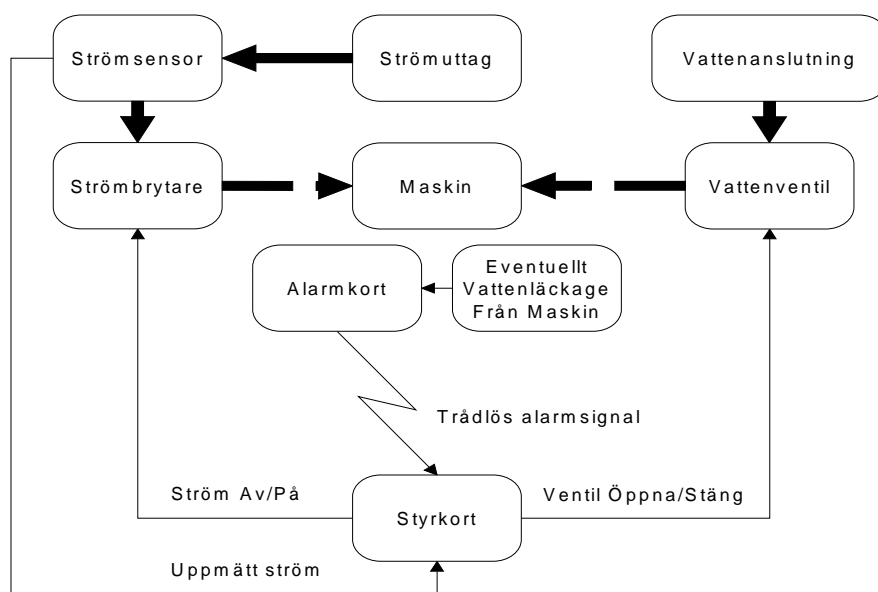
Den här rapporten innehåller bland annat elschema till en ny prototyp, beskrivning av den nya källkoden och komponentkostnader.

Alla priser i denna rapport är exkl moms, i de fall då priser räknats om ifrån USD användes en dollarkurs på 6.8kr.

2 Problembeskrivning

Den ursprungliga prototypen består av två enheter, en styrenhet och en alarmenhet.

En vattenventil är kopplad till styrenheten. Ett flödesschema över produktens funktion ges i figur 1.



Figur 1 Flödesschema produktfunktion

Krav som ställs på enheterna:

Alarmenhetens uppgifter är:

- Ansluta till det trådlösa nätverket.
- Upptäcka vattenläckage och då skicka en alarmsignal till styrenheten.

Styrenhetens uppgifter är:

- Hålla i det trådlösa nätverket.
- Om en alarmsignal mottagits ska styrenheten gå in i alarmläge och bryta ström och vattentillförsel till maskinen.
- Upptäcka om batteriet till alarmlarmenheten tagit slut, gå in i batterislutsläge och bryta ström samt vattentillförseln till maskinen.
- Kontinuerligt mäta strömmen till maskinen för att avgöra om den används.
- Om maskinen inte används ska vattentillförseln brytas.
- Indikera status med lysdioder.
- Alarm och batterislutläge ska kunna återställas

Vattenventilen:

- Är normalt stängd.
- Öppnar av $230V_{RMS}$ och förbrukar 7W.

Om maskinen antas:

- Maximal effektförbrukning på 2300W.
- Okänt induktiv last.
- Maskinen används om den förbrukar mer än 15W.

Alarmenheten är batteridrivna och ska förbruka så lite ström som möjligt.

Möjlighet till att använda sig av flera alarmenheter efterfrågades. Komponentpriset ska vara så lågt som möjligt.

Vad som saknas i den ursprungliga prototypen:

- Strömmen till maskinen kan inte brytas.
- Det finns inget sätt att återställa alarmläge.
- Det finns inget sätt att avgöra om batteriet är slut.
- Priset på komponenter är för högt.
- Stödjer enbart en alarmenhet.
- Alarmenheten förbrukar för mycket ström.

3 IEEE 802.15.4 / ZigBee

3.1.0 IEEE 802.15.4

IEEE 802.15.4[10] är en specifikation för trådlösa nätverk speciellt framtagna för låg strömförbrukning och datahastighet.

IEEE 802.15.4 definierar tre frekvensband för enheter, se figur 2.

Frekvens	Datahastighet	Kanaler
2.4Ghz (Global)	250kbps	16
915Mhz (USA)	40kbps	10
868Mhz (Europa)	20kbps	1

Figur 2 Frekvenser IEEE 802.15.4

Den verkliga hastigheten blir dock något lägre. Räckvidden kan nå upp till 100m och modulationen är bandspridning med direktsekvens. Enligt specifikationen måste varje enhet ha en 64-bitars global unique MAC adress (EUI-64). EUI-64 adressen består av två delar. De första 24-bitarna är en Organizational unique Identifier (OUI), den andra delen består av 40-bitar och dess värde är upp till tillverkaren att välja. OUI nummer måste köpas från IEEE och kostar 1650USD [23]. Standarden stödjer två adresseringslägen, enheter kan kommunicera antingen via EUI-64 adresser eller med kortare 16-bitars adresser. Den maximala längden på ett IEEE 802.15.4 MAC paket är 127 bytes. Varje paket har ett pakethuvud och en 16-bitars kontrollsumma. Alla enheter har ett 16-bitars nätverks ID. För att ett skickat paket ska nå

fram till avsedd enhet, måste nätverks ID och destinations adress i pakethuvudet stämma överens med mottagarens. Det finns en broadcast adress som alla enheter i nätverket måste lyssna på. Standarden har även stöd för kryptering av data, men inte distribution av krypteringsnycklar. Standarden använder sig av CSMA-CA, vilket innebär att alla enheter tillåts att sända på kanalen då den är outnyttjad. Det finns även stöd för beaconnätverk, då tillåts enheter enbart att sända i vissa tidsluckor.

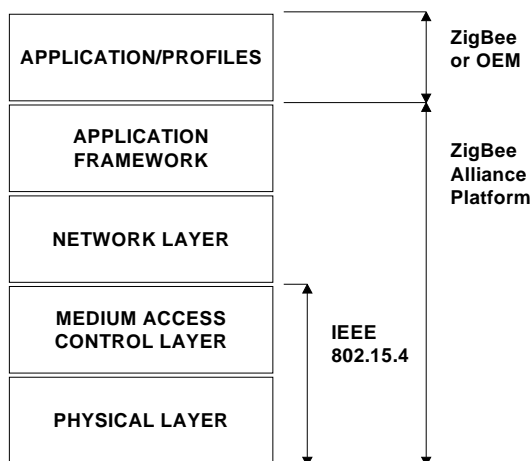
IEEE 802.15.4 specificerar två enhetstyper

- Full Functional Device(FFD)
- Reduced Functional Device(RFD)

En RFD enhet behöver inte implementera alla funktioner i standarden och används typiskt till batteridrivna enheter.

3.2.0 ZigBee

ZigBee[19] är ett nätverksprotokoll för trådlösa nätverk och använder sig av IEEE 802.15.4 specifikationen. ZigBee tillför ett antal lager utöver IEEE 802.15.4, se figur 3.



Figur 3 IEEE 802.15.4/ZigBee lager

ZigBee standarden har stöd för säker dataöverföring och använder sig av enhetsprofiler. En ZigBee enhetsprofil är ett standard protokoll för hur vissa typer av produkter kommunicerar med varandra. Det finns två olika typer av enhetsprofiler i ZigBee standarden, publika och privata. De publika profilerna tas fram av ZigBee alliansen och definierar ett protokoll för en viss funktionalitet. De publika profilerna är tänkta att kunna underlätta utvecklingsarbetet vid framtagning av nya produkter. En egen privat profil kan användas då det inte finns någon passande publik profil, eller då man inte önskar sig ett öppet system. För att få sälja produkter som använder sig av ZigBee standarden måste man vara medlem i ZigBee alliansen, vilket kostar från 3500USD om året [24]. Medlemskapet innebär också att man får tillgång till de publika profilerna. För att få en produkt ZigBee certifierad måste den lämnas in för testning.

ZigBee standarden definierar tre enhetstyper ovanpå IEEE 802.15.4 enhetstyper, se figur 4.

ZigBee Enhet	IEEE Enhet	Funktion
Coordinator	FFD	Håller i nätverket.
Router	FFD	Används för att öka räckvidden och antal noder.
Endpoint	FFD eller RFD	Utför typiskt någon form av funktion.

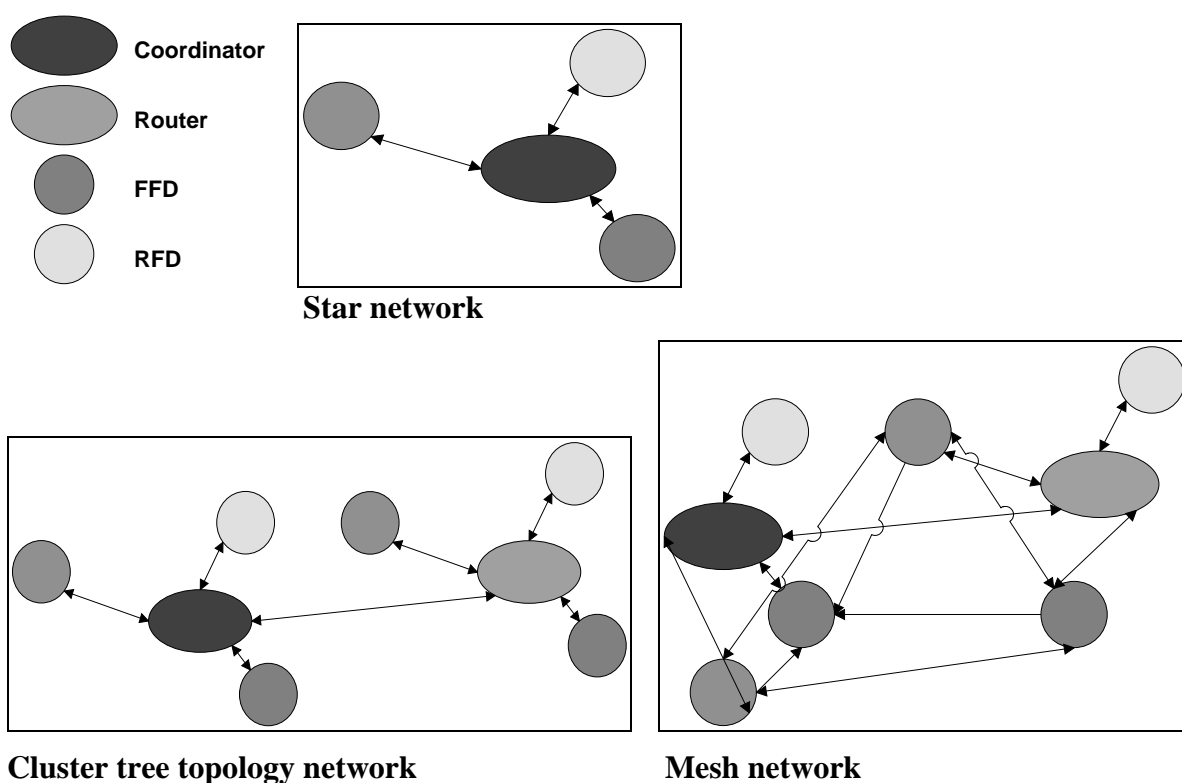
Figur 4 ZigBee enhetstyper

ZigBee nätverk stödjer tre olika nätverks topologier, se figur 5.

I starnetwork topology kan endpoints enbart kommunicera via coordinator.

I Cluster tree topology network kan endpoints även kommunicera via routers.

Slutligen i Mesh network kan även FFD enheter kommunicera direkt med varandra.



Figur 5 ZigBee nätverks topologier

3.3.0 Kretsar/Tillverkare

Det finns många kretstillverkare som erbjuder IEEE 802.15.4/ZigBee kompatibla lösningar, för exempel se figur 6. Oftast så säljer de kretsar med tillhörande ZigBee kompatibel mjukvara, som är bunden till deras mikrokontrollers. Det finns även System-on-chip (SOC) lösningar (mikrokontroller och transceiver på samma krets).

Tillverkare	Krets	IEEE 802.15.4 API	ZigBee Stack
Atmel	AT86RF230	Ja	Nej
Microchip	MRF24J40	Ja	Ja
Chipcon(Texas Instruments)	CC2420 CC2430(SOC)	Ja	Ja
Freescale	MC13213 (SOC) MC13192 MC13202	Ja	Ja

Figur 6 Några exempel på kretsar.

3.4.0 ZMR2A ZigBee modul

IEEE 802.15.4/ZigBee modulen som används i prototypen heter ZMR2A.

Den använder sig av en mikrokontroller ATmega128L från Atmel och en 2.4GHz IEEE 802.15.4 transceiver CC2420 från Chipcon. Modulen har en inbyggd antenn [12] och spänningsregulator REG102 [11]. Anslutning till ett eventuellt moderkort görs via en 30 pinnars kontakt. För komplett schema till ZMR2A, se figur 8.

Några egenskaper hos mikrokontroller och transceiver:

ATmega128L [3]

- 128kB Flash Minne.
- 4kB SRAM.
- 4kB EEPROM.
- 64 pinnar varav 53 är konfigurerbara som I/O.
- 2.7 – 5.5V matningsspänning
- Upp till 8MHz klockfrekvens.
- 6 st strömsparlägen.
- 8 kanalers 10-bitars ADC, varav 2 kanaler kan användas i differentiellt läge med programmerbar förstärkning (1x 10x 200x).
- JTAG interface.
- 2st USARTs.

CC2420 [2]

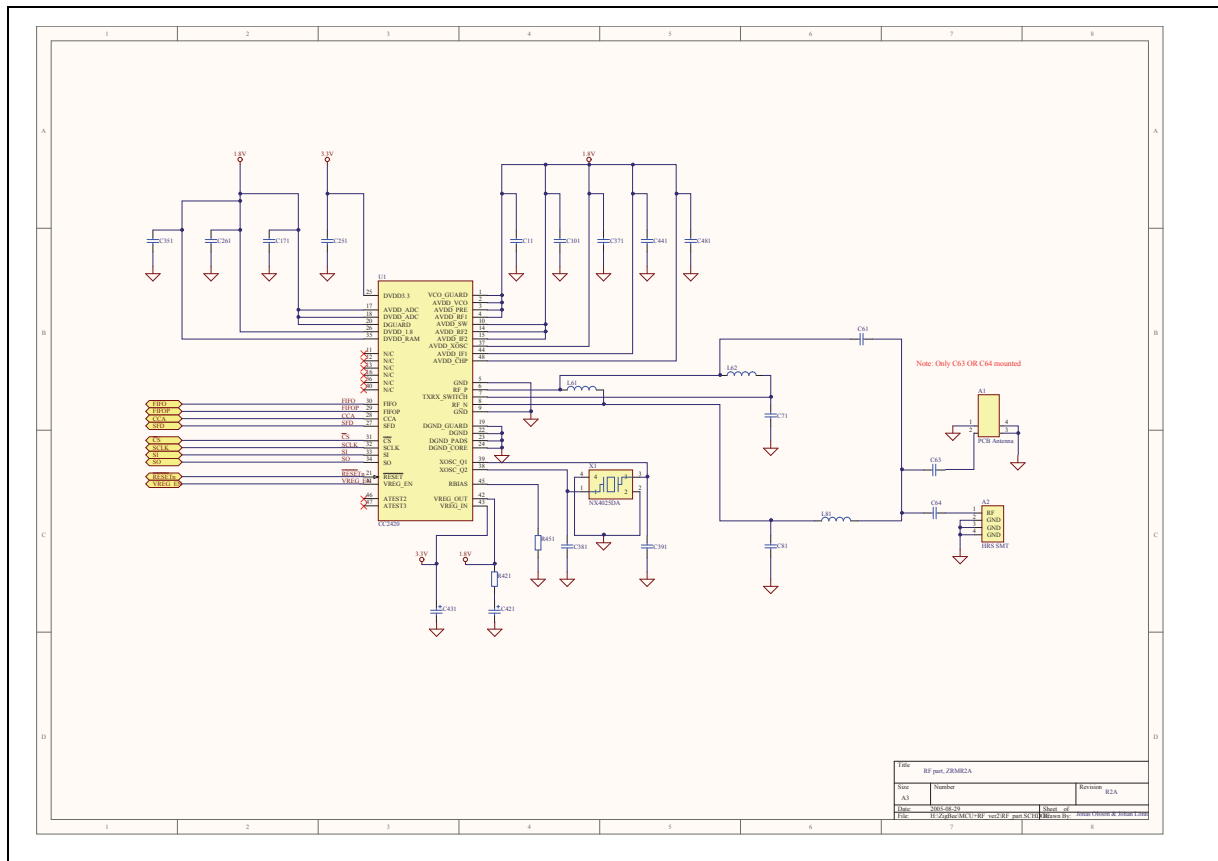
- 802.15.4/ZigBee kompatibel.
- 2.4 GHz med 250kbps data hastighet.
- 4 pinnars SPI interface.
- Passar för både till FFD och RFD
- Låg strömförbrukning (RX: 18.8mA , TX 17.4 mA)
- Inbyggd spänningsregulator, kan användas med 3.3V matningsspänning.

Ungefärligt komponentpris till ZMR2A ZigBee modul, ges i figur 7.

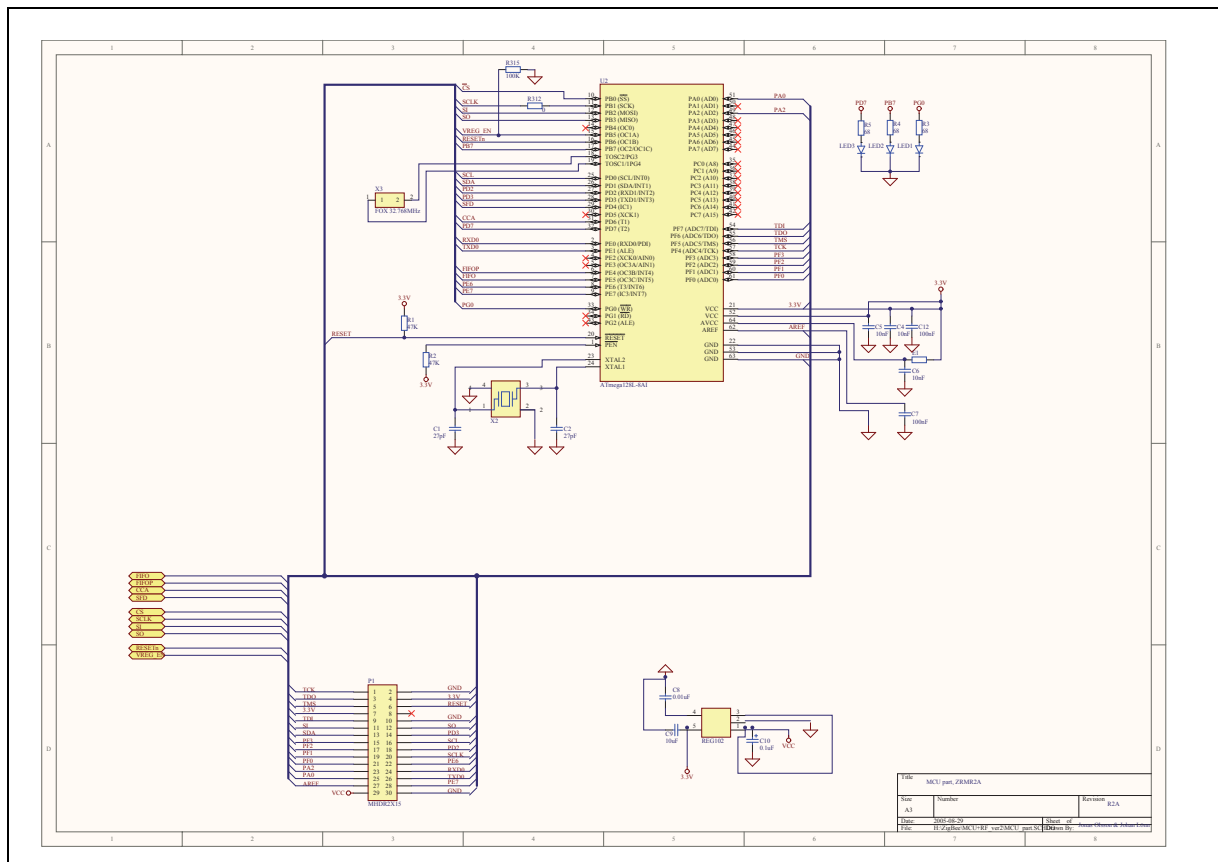
Priser för resistorer, kondensatorer och induktanser är inte inräknade.

Beskrivning	Namn	Om 1st	Om 100st		
Transceiver	CC2420	44,88 kr	30,29 kr	Digikey	296-19585-1-ND
Mikrokontroller	ATMEGA128L	103,09 kr	59,91 kr	Digikey	ATMEGA128L-8MU-ND
Sp-regulator	REG102	12,51 kr	9,90 kr	Farnell	8435278
Antenn	MICA	28,56 kr	20,40 kr	Digikey	627-1009-1-ND
8MHz kristall	FOXLF080-20	3,60 kr	2,55 kr	Digikey	631-1083-ND
16MHz kristall	NX4025DA	7,14 kr	4,52 kr	Digikey	644-1107-1-ND
32,768 KHz kristall	FSRLF327	7,00 kr	4,01 kr	Digikey	631-1000-1-ND
		206,79 kr	131,58 kr		

Figur7 Ungefärlig komponentkostnad ZMR2A



Figur 8 ZMR2A ZigBee modul



Figur 8 ZMR2A ZigBee modul

4 Analys av ursprunglig hårdvara.

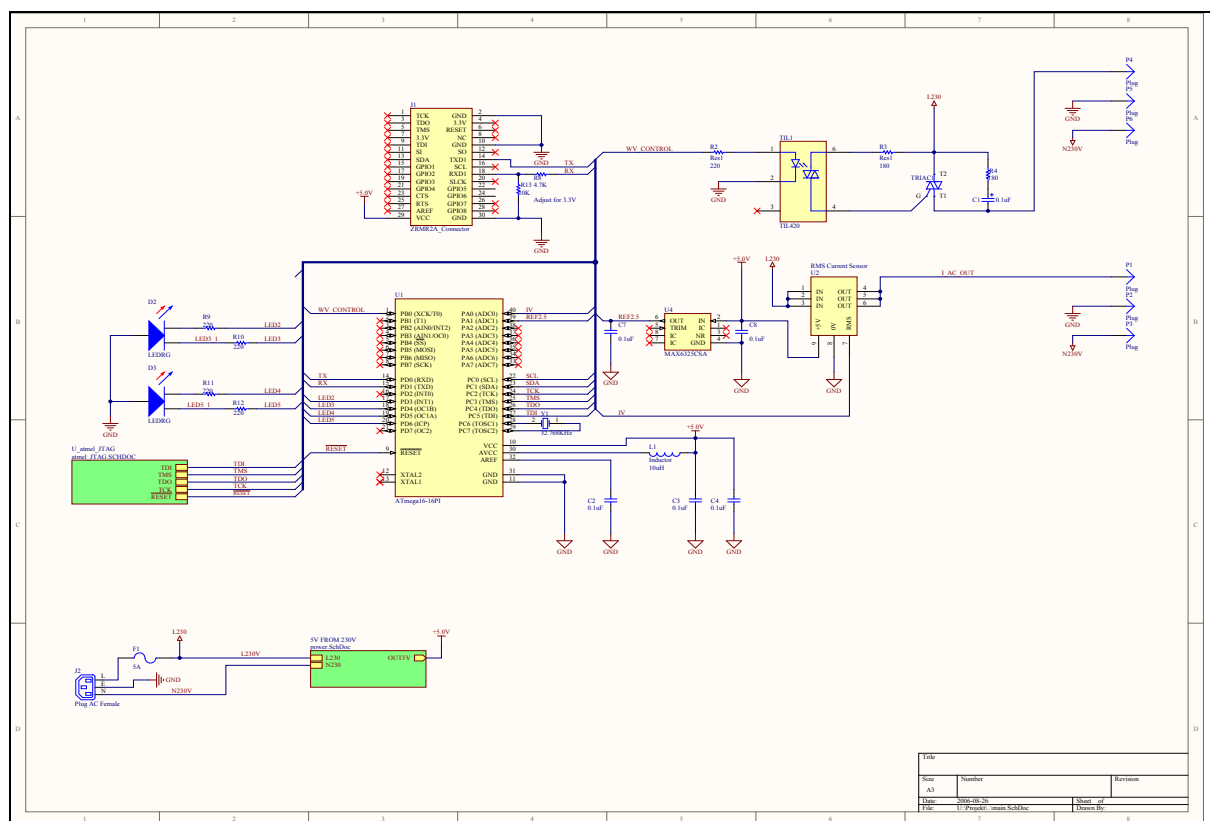
Hårdvaran består av två enheter, en alarmenhet och en styrenhet.

Alarmenheten är batteridriven och ska därför förbruka så lite ström som möjligt.

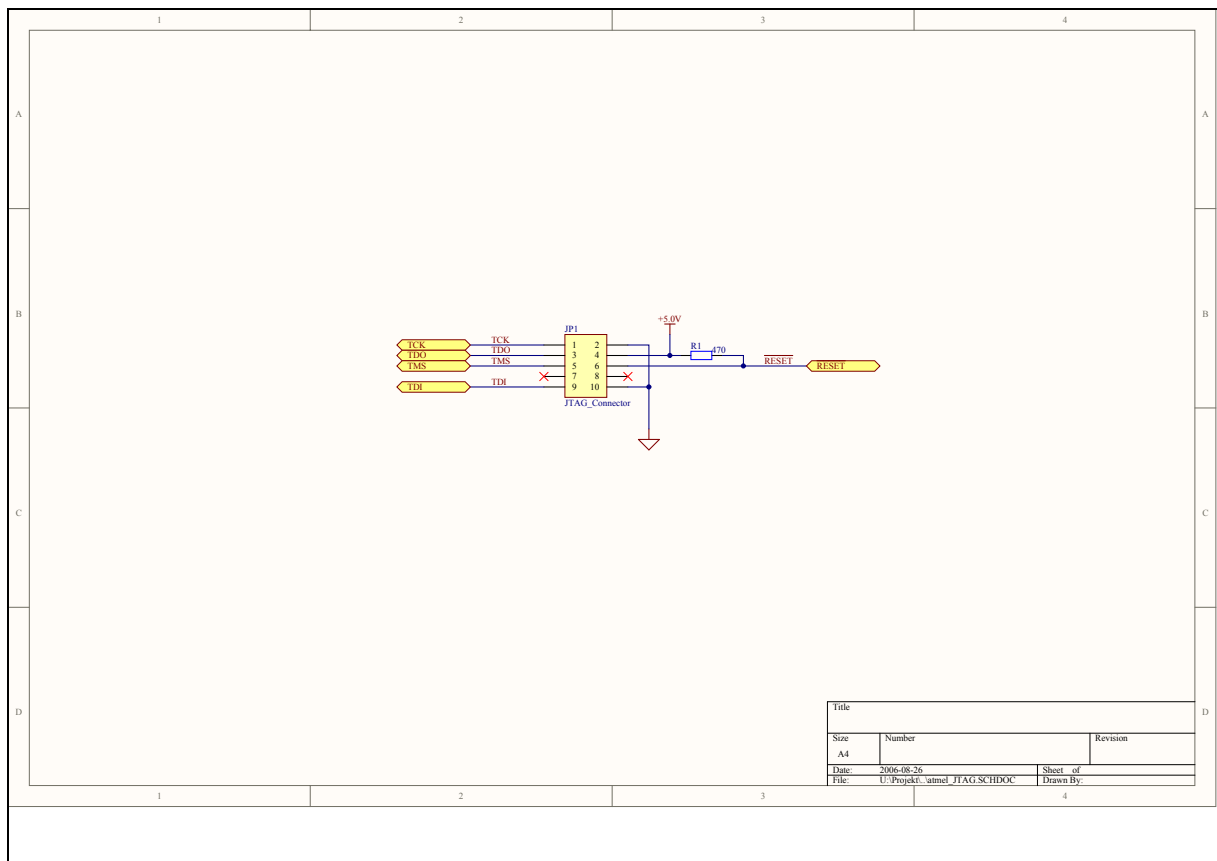
Båda enheterna har var sitt dotterkort som består av en ZMR2A ZigBee modul.

4.1.0 Styrenhet

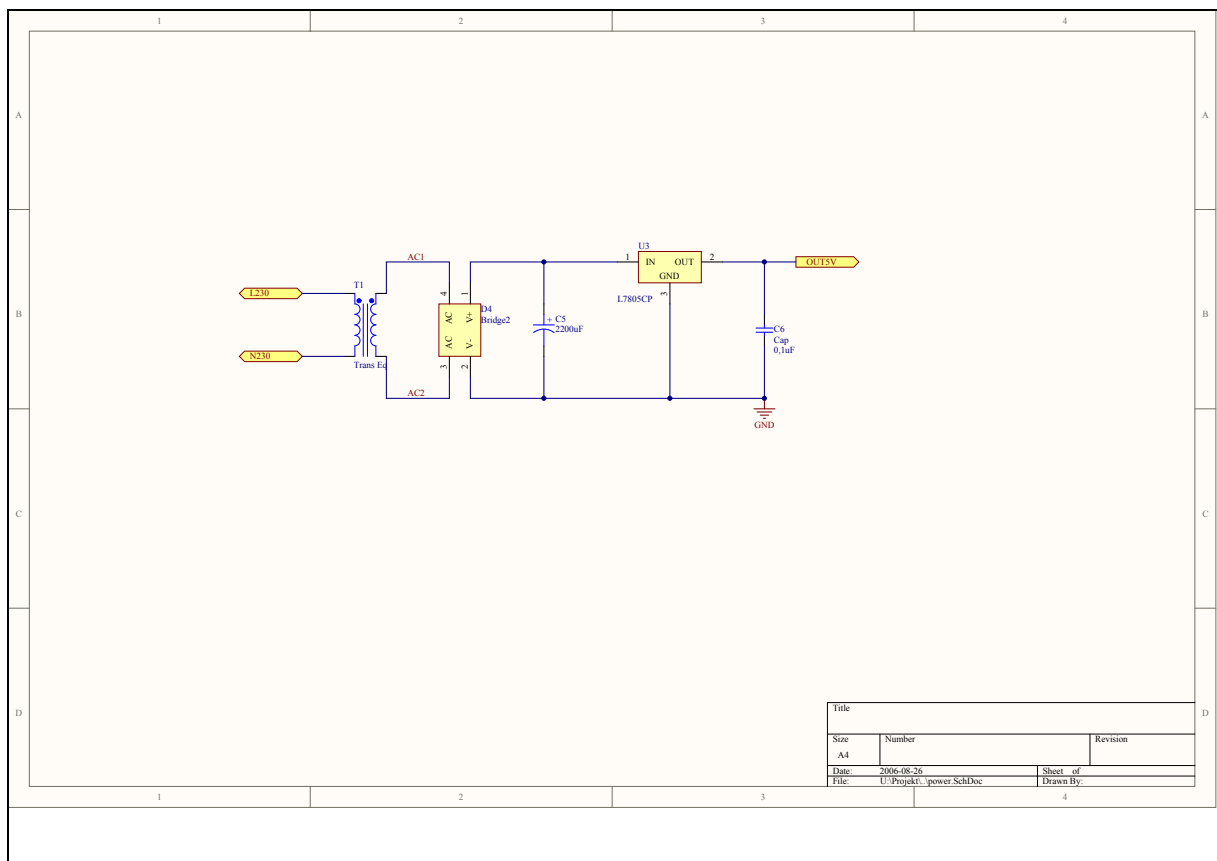
Schema till den ursprungliga styrenheten, se figur 9.



Figur 9 Schema ursprunglig styrenhet



Figur 9 Schema ursprunglig styrenhet



Figur 9 Schema ursprunglig styrenhet

Den ursprungliga styrenheten använder sig av två mikrokontrollers.

Förutom mikrokontrollern på ZMR2A modulen finns en Atmel ATmega16 på kretskortet för styrenheten. De båda mikrokontrollerna kommunicerar med varandra via USART.

Då Mikrokontrollern på ZMR2A tagit emot en alarmsignal, skickar den en signal vidare till mikrokontrollern på styrenheten via USART.

Mikrokontrollern på styrenheten har till uppgift att mäta strömmen genom maskinen med en halleffekts-strömsensor, samt att öppna och stänga vattenventilen.

Strömsensorn som används heter LTS515-NP.

I den här kopplingen ger den ifrån sig en spänning som beror av strömmen enligt ekv 1:

$$v(i) = 2.5 + \frac{0.625}{15} * i \quad (\text{ekv 1})$$

ADC omvandlaren i mikrokontrollern på styrenheten används i differentiellt läge med 10 ggr förstärkning och 5V intern spänningsreferens.

Omvandlingen sker enligt ekv 2:

$$ADC = \frac{(V_+ - V_-) * GAIN * 512}{V_{ref}} \quad (\text{ekv2 resultat i tvåkomplement})$$

Den positiva ingången till ADC omvandlaren är kopplad till utsignalen från strömsensorn och den negativa ingången är kopplad till en 2.5V spänningsreferens.

Efter insättning av ekv1 och värden i ekv2 ser man att $ADC(i) = 42.67 * i$.

Maskinen antas användas om den förbrukar mer än 15W.

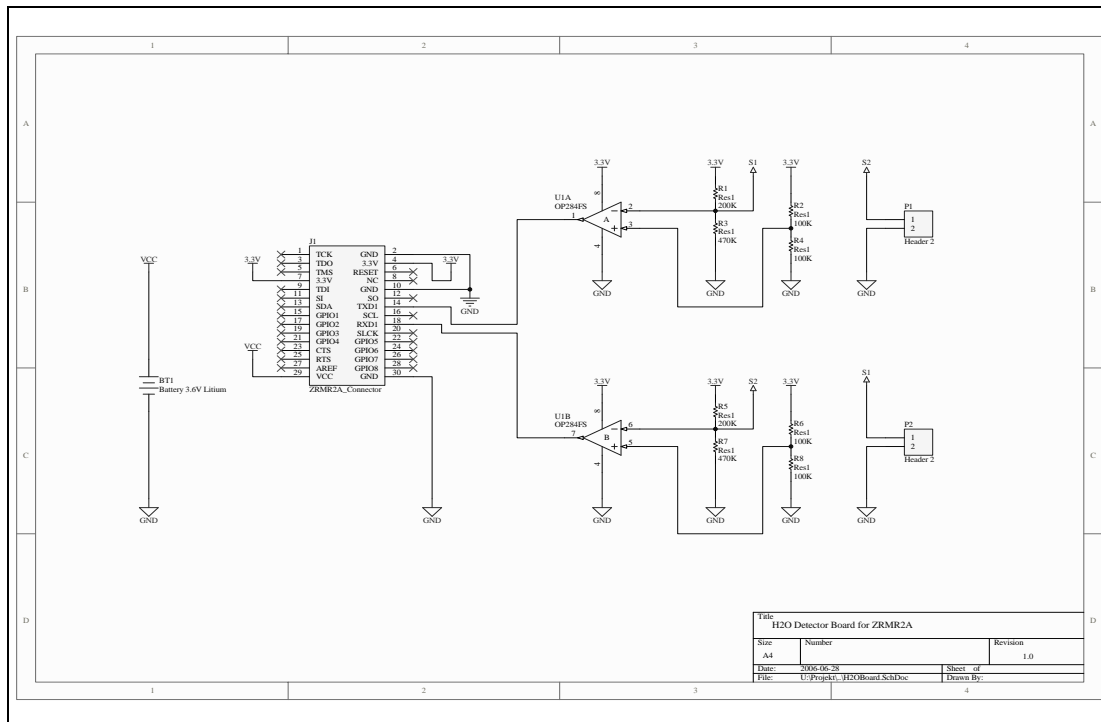
Vid 15W blir toppvärdet av strömmen $i_{\max} = \frac{P\sqrt{2}}{U} = 92\text{mA}$, $P=15\text{W}$, $U = 230\text{Vrms}$

Vid 92mA används bara tre bitar av ADC omvandlarens tio möjliga. Det innebär att ADC omvandlarens mätområde utnyttjas på ett dåligt sätt och gör mätningen onödigt känslig för störningar. Ett lågpasfilter saknas vilket kan orsaka vikningsdistortion.

För att öppna vattenventilen används en triac, den drivs av en optokopplare med triac utgång. Ett RC-nät används för att skydda mot överspänningar, då strömmen till vattenventilen bryts.

4.2.0 Alarmenhet

Schema till den ursprungliga alarmenheten, se figur 10.



Figur 10 Schema ursprunglig alarmerhet

Alarmkortet använder sig av två stycken mätbryggor kopplade till var sin operationsförstärkare i komparator koppling.

Då spänningen på operationsförstärkarens positiva ingång är mindre än spänningen på den negativa ingången ligger utgången logiskt låg. Det är normalfallet.

Normalfallet med utsignal låg för ena bryggan:

$$V_+ = \frac{R4}{R4 + R2} * 3.3V = 1.65V$$

$$V_- = \frac{R3}{R1 + R3} * 3.3V = 2.3V$$

Då V_- blir mindre än V_+ slår utsignalen om till hög, det sker då resistansen mellan V_- till jord är mindre än R_1 .

$$R1 = R3 // R_S1$$

$$R_{-S1} = \frac{R1 * R3}{R3 - R1}$$

Dvs utsignalen hög då resistansen mellan S1 och jord är mindre än 348 kohm.

Den andra bryggan är kopplad på samma sätt.

Utsignalerna från operationsförstärkarna är kopplade till var sin ingång på ZMR2A modulen.

4.3.0 Komponenter/Pris

De dyraste komponenterna identifierades, en lista över dem ges i figur 11.

Komponent	Pris (i styckvis kvantitet)	Leverantör	Kommentar
MAX6325	156.20kr	Farnell[17]	Behövs ej. Alldeles för dyr.
LTS515-NP	132.98kr	Farnell	Finns billigare alternativ.
ATmega16	44.61kr	Digikey[16]	Behövs ej.
OP284FS	42.84kr	Farnell	Finns billigare alternativ.

Figur 11 Lista över de dyraste komponenterna

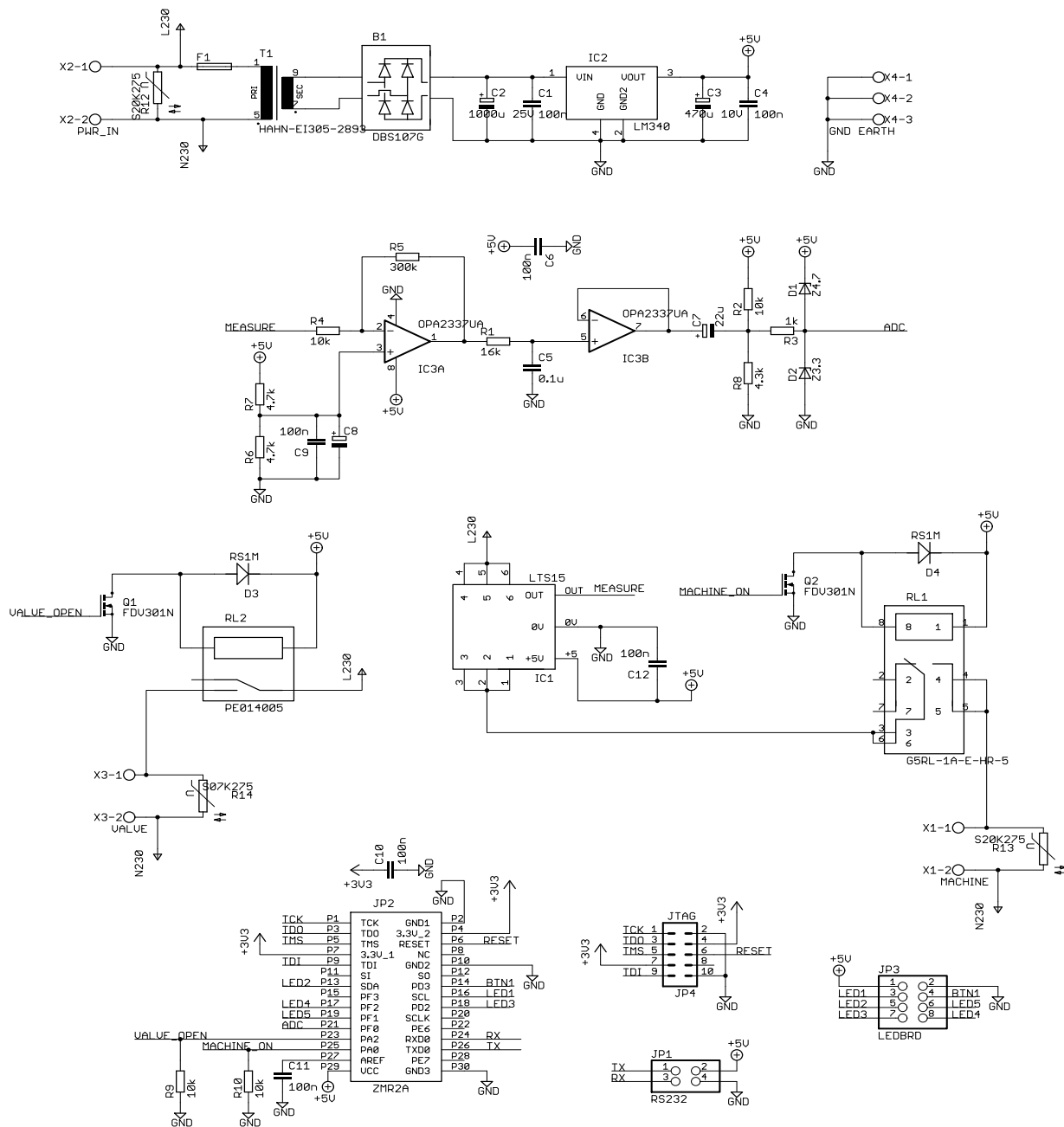
5 Förslag till ny prototyp

En ny prototyp togs fram, följande problem togs i åtanke

- Ett sätt att slå av/på maskinen måste finnas.
- Billigare komponent bör väljas.
- Lägre strömförbrukning hos alarmkortet.
- Möjlighet till "in circuit programming".
- En fysiskt mindre design för kretskortet.

5.1.0 Styrenhet

Schema till den nya styrenheten, se figur 12.



Figur 12 Schema ny styrenhet

Det första som gjordes var att besluta att bara använda sig av en mikrokontroller. Då ligger det nära till hands att även placera en tranceiverkrets direkt på kretskortet och ta bort ZMR2A modulen. Idén fick överges på grund av svårigheter att löda tillgängliga tranceiverkretsar för hand. Istället fick ZMR2A modulen vara kvar och mikrokontrollern på styrenheten togs bort. Mikrokontrollern på ZMR2A modulen använder sig av matningsspänningen 3.3V, därför måste utsignalen ifrån strömsensorn nivåanpassas och dessutom bör den lågpassfiltreras. Det medförde att ett analogt förstärkningssteg byggdes. ADC omvandlaren används nu i single-ended mode, vilket gör designen mindre beroende av mikrokontrollers från Atmel.

En ny billigare halleffekt-strömsensor hittades, Allegro ACS712-20[13].

Den fungerar på samma sätt som den ursprungliga strömsensorn LTS15-NP, skillnaden är att den är lite känsligare. Då den använder sig av en liten SO8 kapsel bör speciell hänsyn tas vid design av kretskort[13]. Det fick konsekvensen att det nya kretskortet designades för den ursprungliga LTS15-NP. Ett extra kretskort för ACS712-20 tillverkades, vilket gjordes speciellt för att direkt kunna ersätta LTS15-NP på det nya kretskortet.

Några egenskaper hos ACS712-20

- Max 20A
- 2.1kV Isolations spänning.
- SO8 kapsel
- $v(i) = 2.5 + 0.1 * i$

Ett sätt att bryta strömmen till maskinen var ett av kraven för den nya prototypen, alternativen triac och relä togs i åtanke.

Då maskinen i värsta fall kan dra upp till $10 A_{RMS}$, uppstod tvivel om värme utveckling i triac alternativet. Triac Ekvationer enligt [9]:

$$P = V_o * I_{T(AVE)} + R_s * I_{T(RMS)}^2 \quad (\text{ekv 3})$$

$$I_{T(AVE)} = \frac{2 * \sqrt{2} * I_{T(RMS)}}{\pi} \quad (\text{ekv 4})$$

P = Triac power dissipation.

V_o = Triac knee voltage.

$I_{T(AVE)}$ = Average load current.

R_s = Triac slope resistance.

$I_{T(RMS)}$ = RMS load current.

$$T_j = T_a + P * R_{th(j-a)} \quad (\text{ekv 5})$$

T_j = Junction temperature.

T_a = Ambient temperature.

$R_{th(j-a)}$ = Junction to ambient thermal resistance. $^{\circ}C / W$

$$R_{th(j-a)} = R_{th(j-mb)} + R_{th(mb-hs)} + R_{th(hs-a)} \quad (\text{ekv 6})$$

$R_{th(j-mb)}$ = Junction to mounting base thermal resistance. $^{\circ}C / W$

$R_{th(mb-hs)}$ = Mounting base to heatsink thermal resistance. $^{\circ}C / W$

$R_{th(hs-a)}$ = Heatsink to ambient thermal resistance. $^{\circ}C / W$

Om man använder en BT139 TRIAC [8] och en maximal effekt på 2300W då blir $I_{T(RMS)} = 10A$, vilket enligt ekv 4 ger $I_{T(AVE)} = 9A$.

BT139 datablad ger:

$$V_o = 1.195V$$

$$R_s = 0.018\Omega$$

$$T_j \text{ absolute max} = 125^\circ C$$

$$R_{th(j-a)} \text{ TO220 utan kylfläns} = 60^\circ C/W$$

$$R_{th(j-mb)} = 1.2^\circ C/W$$

$$R_{th(mb-hs)} = 1.6^\circ C/W \text{ (fett och 0.05 mm mica isolations bricka)}$$

Enligt ekv 3 blir effektförlusten i triacen då $P = 12.56W$

I temperaturberäkningarna nedan används en rumstemperatur på $T_a = 40^\circ C$.

Enligt ekv 5, med BT139 utan kylfläns ($R_{th(j-a)}=60^\circ C/W$) blir $T_j = 793.6^\circ C$, uppenbarligen måste en kylfläns användas. Om man använder sig av ekv 5 och sätter $T_j = 100^\circ C$ kan man räkna ut att det behövs en kylfläns med $R_{th(hs-a)}$ på max $1.98^\circ C/W$.

Det innebär att en ganska kraftig kylfläns måste användas, därför bör ett relä vara ett bättre val. Det bör dock påpekas att en diskmaskin knappast konstant förbrukar 2300W.

Valet föll på ett relä från Omron G5RL-1A-E-HR-5[4] (RL1)

- 16A 250 V~
- Klarar strömstötter på upp till 100A
- 5V spolspänning.
- Spolresistans 62.5 ohm

Till vattenventilen går det alldeles utmärkt att använda sig av en triac utan kylfläns, men då kretskortet ändå inte blev reläfritt, så beslöts att använda ett relä även till ventilen.

Relä schrack PE014005[5] (RL2)

- 5A 250V~
- 5V spolspänning
- Spolresistans 125 ohm

Vattenventilen förbrukar 7W, vilket reläet klarar med stor marginal.

Då mikrokontrollern inte kan driva reläerna direkt används logik styrda MOSFET [6] för det ändamålet. De arbetar i triode då de leder.

$$i_D = K(2(V_{GS} - V_{to})V_{Ds} - V_{Ds}^2) \text{ och } V_{Ds} = 5V - I_D * R \text{ (ekv 7, ekv 8)}$$

$$V_{GS} = 3.3V$$

R = Reläets spolresistans

K, V_{to} = Transistor parametrar

Med hjälp av ekv 7 och ekv 8 kan man lösa ut I_D, V_{DS} , men då det fanns tillgång till noggrannare PSpice-modeller för transistorerna, simulerades de båda reläkretsarna istället. Både ström och Effektförbrukning visade sig vara under tillåtna gränser.

Dioderna D3, D4 är "flyback" dioder, de är till för att skydda transistorerna mot höga spänningar då strömmen till reläspolen bryts. Den maximala strömmen genom dioderna är den samma som den maximala strömmen genom reläspolarna.

För att skydda reläerna mot överspänningar vid brytning av induktiva laster valdes metalloxid varistorer från Epcos[20]. De kan förenklat ses som olinjära motstånd med en hög resistans vid låga spänningar och slår om till en låg resistans då spänningen nått en viss tröskelnivå. Det enda man kan anta om maskinlasten är att den kan dra upp till $10 A_{RMS}$, så en beräkning på det absolut värsta fallet är att föredra. Det sker då lasten är rent induktiv och strömmen bryts då den nått sitt maximum. Detta fall är egentligen omöjligt i praktiken men det lämnar en god säkerhets marginal.

$$U = 230V_{RMS}, I = 10A_{RMS}, \text{ ger } Z = 23\Omega$$

$$R=0 \text{ ger } L = \frac{Z}{2\pi 50} = 73mH$$

$$i_{max} = I * \sqrt{2} = 14.14A$$

$$\text{Energien i induktansen är } W = \frac{1}{2} L * i_{max}^2 \text{ (ekv 9) [26]}$$

I detta fallet ger ekv 9, $W=7.29J$.

De viktiga parametrarna vid val av varistor i detta fall är:

- i_{max} max ström genom varistor
- v_{max} max spänning över varistor då strömmen i_{max} går igenom den
- W_{max} energin som måste absorberas av varistorn under tiden τ

Varistorn S20k275 klarar större ström än i_{max} , V/I kurva från datablad[25] ger

$$v_{max} = 650V \text{ och } R_{SIOV} = 46\Omega \text{ vid } i_{max}.$$

$$\text{Tiden tills strömmen avtagit till ca 37\% av ursprunglig ström: } \tau = \frac{L}{R + R_{SIOV}} \text{ (ekv10) [26].}$$

Insättning av värdena i ekv10 ger $\tau = 1.6ms$.

Från en tabell ur databladet kan man se att varistorn klarar strömpulser av storlek i_{max} med längden τ , ungefär 10000 ggr.

$$\text{Energien i en sådan puls är } W_{max} = v_{max} * i_{max} * \tau \text{ (ekv 11) [26].}$$

Enligt ekv 11 blir energin i pulsen $W_{max} = 14.7J$, vilket är mindre än de 7.29J som induktansen kan innehålla. Det innebär att varistorn S20K275 (R13) kan användas med en stor säkerhetsmarginal. Samma varistor valdes till att skydda vid anslutningspunkten för nätspänningen, då induktansen på nätsidan rimligtvis inte kan överstiga induktansen hos den här teoretiska lasten. Varistorn R14 beräknas på samma sätt, fast med en mycket lägre maximal ström, där visade det sig att en billigare S07K275 räcker.

Transformatorn T1 som används är på 2.3VA och lämnar $7.5V_{rms}$ vid full last.

Diodbryggan B1 helvågsl riktar spänningen och klarar 1A i medelström. Den har ett framspänningsfall på ca 1.1V vid 1A. Den linjära spänningsregulatorn IC2 LM340 [7] levererar 5V och klarar 1A, den har ett minsta spänningsfall på 2V. Om man drar bort spänningsfallen över dioderna och regulatorn från spänningen ifrån transformatorn, kommer man fram till att spänningsriplet får vara max:

$$V_r = 7.5V_{rms} * \sqrt{2} - 2 - 1.1 = 2.5V.$$

Spänningsriplet kan i detta fall ungefärligt uppskattas med ekv 12 [1]

$$V_r = 2 * C * I * (1 / F) \text{ (ekv 12)}$$

C = capacitans.

I = konstant lastström här 180mA

F = 50Hz

Om man löser ut C ur ekv 12 så ger det 666uF, filterkondensatorn C1 väljs till 1000u för att få en marginal. Strömmen genom maskinen mäts av IC1 (LTS15-NP alt ACS712-20) utsignalen går sedan till det analoga steget uppbyggt kring operationsförstärkaren IC3.

Det analoga steget består först av en inverterad förstärkning(IC3a).

$$A_v = -\frac{R5}{R4} = 30$$

Sedan lågpas filtreras signalen.

$$F = \frac{1}{2\pi * R1 * C5} = 99.5\text{Hz}$$

Sedan går den till IC3b vilket är en spänningsföljare, därefter högpas filtreras signalen.

$$F = \frac{1}{2\pi * (R2 // R8) * C7} = 2.4\text{Hz}$$

R2 och R8 ser till att addera en DC offset på 1.5V.

$$V = \frac{R8}{(R2 * R8)} * 5V = 1.5V$$

Zenerdioden D2 begränsar utsignalen över 3.3V.

Max ström genom D2 blir (4V-3.3V) / R3 = 0.7mA.

Zenerdioden D1 begränsar utsignalen under (5V-4.7V) = 0.3V.

Max ström genom D1 blir (1V+0.3V) / R3 = 1.3mA.

Utsignalen begränsas till max 3.3V och min 0.3V, för att skydda ingångarna på mikrokontrollern.

Med det nya analoga steget blir ADC(i):

- LTS15-NP ger $ADC(i) = 465 + 388 * i$
- ACS712-20 ger $ADC(i) = 465 + 931 * i$

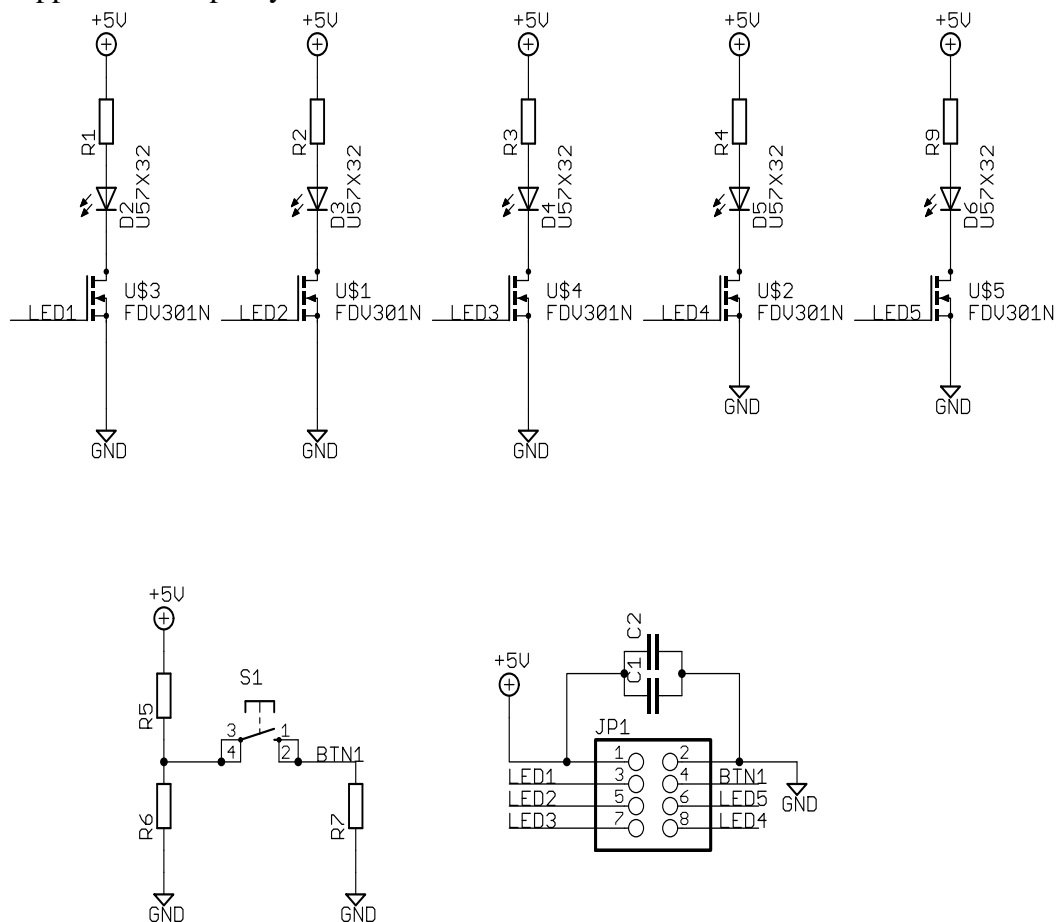
Den totala strömförbrukningen av 5V uppskattas till ca 180mA, ungerfärlig total medeleffektförbrukning simulerades fram i PSpice och visade sig vara ungefär 1.8W.

JP1 är kopplad till en USART på ZMR2A modulen den används för avlusning.
 JP2 är anslutningen för ZMR2A modulen
 JP3 är anslutningen till ett dotterkort med status LED:s och återställningknapp.
 JP4 är anslutningen för JTAG interface för ZMR2A modulen.

5.1.0.1 Dotterkort för LED och knapp

Kretskortet består av fem status LEDs med tillhörande drivtransistorer, samt en tryckknapp för återställning av alarm och batterislut läge. För schema, se figur 13.

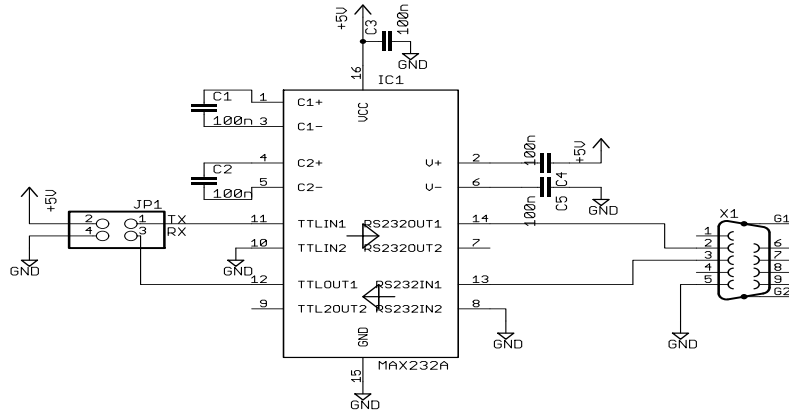
Kopplas till JP3 på styrenheten.



Figur 13 Schema dotterkort

5.1.0.2 RS232 kort

Detta kort är tänkt att användas för avlusning och testmätning av lastström och består av en MAX232 krets för att omvandla signalnivåerna mellan mikrokontroller och en dator. För schema, se figur 14. Kopplas till JP1 på styrenheten.



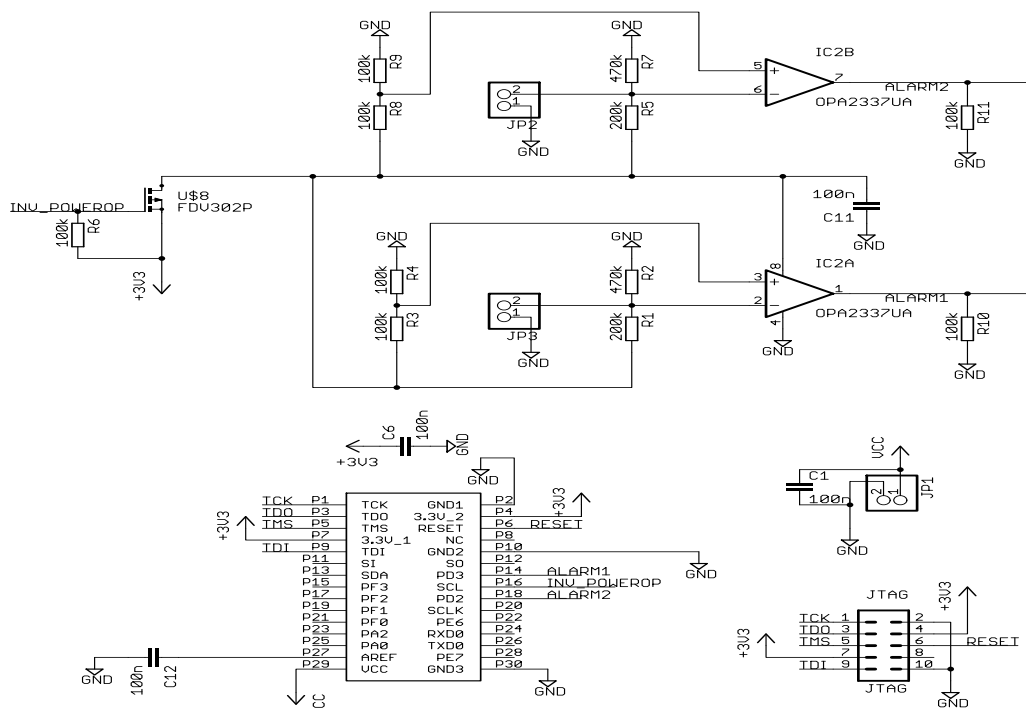
Figur 14 Schema RS232 kort

5.2.0 Alarmenhet

Schema till den nya alarmenheten, se figur 15.

Det nya alarmkortet är nästan identiskt med det gamla, skillnaderna är:

- Ett JTAG interface har lagts till.
- En PMOS MOSFET switch för att stänga av/på matningen till operationsförstärkarna har tillkommit. Detta för att minska strömförbrukningen.
- Operationsförstärkaren har ersatts med en billigare.



Figur 15 Schema nytt alarmkort

6 Programmering

Den ursprungliga källkoden var i dåligt skick så en helt ny mjukvara utvecklades.

Förutom den ursprungliga funktionaliteten behöver följande problem lösas:

- Ett sätt att återställa alarmläge måste implementeras.
- Ett sätt att upptäcka om batteriet är slut måste finnas.
- Möjlighet till stöd för flera alarmentheter.
- Strömförbrukning hos alarmentheten bör minskas.

Programmeringen är uppdelad i två delar, den första delen handlar om den trådlösa kommunikationen. Den andra delen handlar om huvudprogrammen för alarm och styr enheterna. Tyvärr fanns det inte tillgång till någon ZigBee stack för kombinationen ATmega128 och CC2420. Det fanns dock ett API för de två lägsta lagren i IEEE 802.15.4 specifikationen, API:et är utvecklat av Chipcon och heter CC_MAC-V1.3.0.

Ett eget enkelt protokoll för kommunikation mellan enheterna baserat på Chipcons API togs fram, det använder sig av starnetwork topologin.

Några förtydningar för programmeringsdelen:

- När det skrivs ALARM, PING och RESET signaler så är det ASCII strängar som menas.
- Med coordinator så menas ZMR2A modulen på styrenheten, och med endpoint så menas ZMR2A modulen på alarmkortet.

6.1.0 Nätverk Coordinator

Coordinatorn är implementerad som en FFD och den går aldrig ner i strömsparläge.

I EEPROM lagras:

- Nätverks ID (PANID)
- Egen EUI-64 adress.
- EUI-64 adresser till de endpoints som tillåts ansluta.
- Ett 16-bitars värde som används som basvärde för de korta adresserna som tilldelas endpoints då de ansluter.

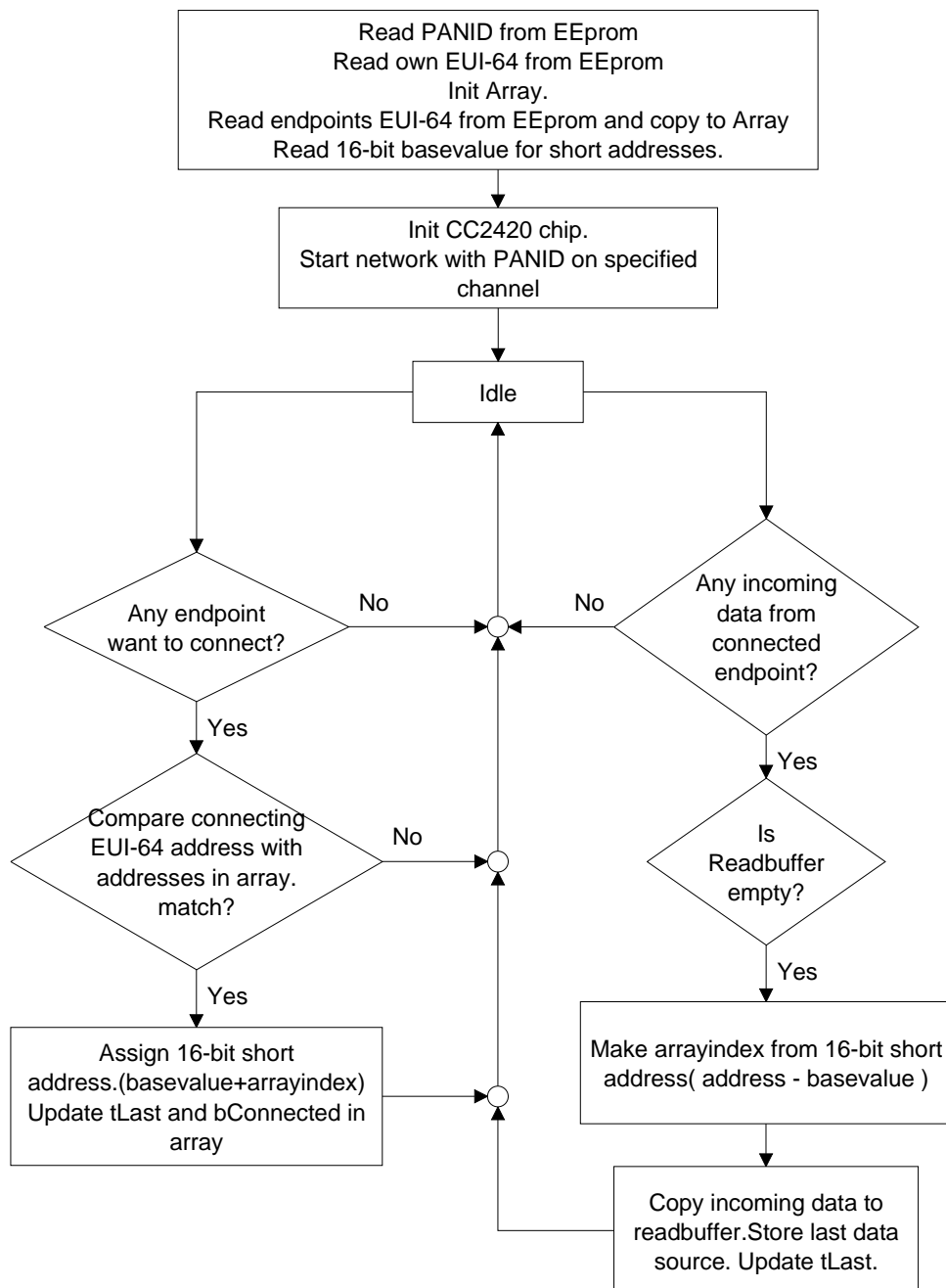
Det maximala antalet endpoints som coordinatorn stödjer definieras i källkoden av MAX_ASSOC. En lista av storleken MAX_ASSOC innehåller DEVICE strukturer, se figur 16. Vid uppstart så kopieras EUI-64 endpoint adresserna från EEPROM in i strukturen och bConnected sätts till FALSE och tLast nollställs.

Datatyp	Namn	Beskrivning
QWORD	Extended	EUI-64 endpoint adress
BYTE	bConnected	Anslutnings flagga. TRUE = Ansluten.
UINT32	tLast	Tidpunkt för senast mottagen data

Figur 16 DEVICE struktur.

Då en endpoint försöker ansluta till coordinatorn så kontrolleras om dess EUI-64 adress stämmer överens med någon adress som finns i DEVICE struktur listan, i sådana fall tilldelas endpointen en kort 16-bitars adress. De korta 16-bitars adresserna räknas fram genom att

addera det index i DEVICE struktur listan som motsvarar en viss endpoint, till det 16-bitars basvärde som är sparat i EEPROM. För mottagning av data används en läsbuffer. Läsbuffern kan enbart innehålla data ifrån en endpoint i taget. Vid indikation från API:et att data har mottagits från en ansluten endpoint samt att läsbuffern är tom, kopieras data till läsbuffern, annars ignoreras mottaget data. Det kan leda till att data försvinner om flera endpoints skickar data vid ungefär samma tidpunkt. Då data kopieras till läsbuffern så sparas också antal mottagna bytes och endpointens index i listan. För flödesschema se figur 17.



Figur 17 Flödesschema nätverk coordinator.

Funktioner i Coordinators wireless.c:

int WIRELESS_init(void)

Initiera nätverk.

void WIRELESS_read_eeprom_settings(void)

Läs data från EEPROM.

int WIRELESS_isconnected(void)

Returnerar TRUE om någon endpoint är ansluten.

UINT32 WIRELESS_getdevstatus(int iNr)

Returnerar tidpunkten för senast mottagen data för endpointen med listindex iNr.

int WIRELESS_read(unsigned char ppRet , int* iNr)**

Returnerar antal bytes i läsbuffer , pekare till läsbuffer via ppRet och endpointens listindex via iNr.

int WIRELESS_rx_cnt(void)

Returnera antal bytes i läsbuffer.

void WIRELESS_rx_flush(void)

Töm läsbuffer.

int WIRELESS_write(unsigned char* pSend , int iCnt , int iNr)

Skickar iCnt antal bytes från buffer pSend till endpointen med listindex iNr

void WIRELESS_tx_flush(void)

Väntar på att data skickats klart eller misslyckats.

int WIRELESS_EXT_address_match(QWORD a1 , QWORD a2)

Jämför EUI-64 nätverks adresser.

void mlmeAssociateIndication(ADDRESS deviceAddress, BYTE capabilityInformation, BOOL securityUse, UINT8 aclEntry)

Anropas då en endpoint försöker ansluta. deviceAddress är endpointens EUI-64 adress.

void mcpsDataIndication(MCPS_DATA_INDICATION *pMDI)

Anropas då data mottagits. Strukturen pMDI innehåller data, datalängd och adress.

void mcpsDataConfirm(MAC_ENUM status, BYTE msduHandle)

Anropas då en sändning avslutats.

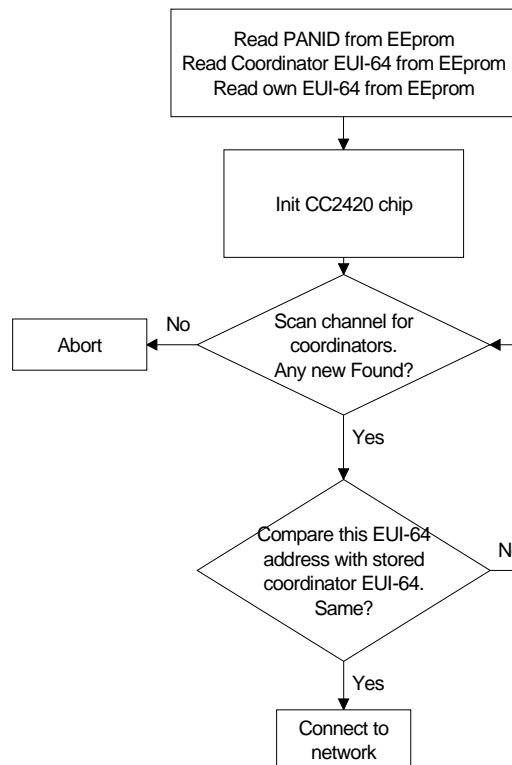
6.2.0 Nätverk Endpoint

Endpoints är implementerade som RFD och har normalt transceiveren i strömsparläge.

I EEPROM lagras:

- Nätverks ID (PANID)
- Egen EUI-64 adress.
- Coordinators EUI-64 adress.

Vid uppstart av en endpoint så söker den efter IEEE 802.14.5 nätverk. Om en coordinator svarar med samma EUI-64 adress som finns lagrad i EEPROM, försöker den ansluta till det nätverket. Läsbuffer är implementerad på samma sätt som för coordinator, men enda möjliga datakällan är coordinatoren. Då en endpoint skickar data till coordinatoren används EUI-64 adressen. För flödesschema se figur 18.



Figur 18 Flödesschema nätverk endpoint.

Funktioner i endpoints wireless.c:

int WIRELESS_init(void)

Initierar nätverk. Söker efter coordinator och försöker ansluta.

void WIRELESS_read_eeprom_settings(void)

Läs data från EEPROM.

int WIRELESS_isconnected(void)

Returnerar TRUE om ansluten till coordinator.

void WIRELESS_rx_enable(void)

Kopplar på strömmen till mottagaren.

void WIRELESS_rx_disable(void)

Stänger av strömmen till mottagaren.

int WIRELESS_read(unsigned char ppRet)**

Returnerar antal bytes i läsbuffer, pekare till läsbuffer via ppRet. Enda möjliga datakälla är coordinatorm.

int WIRELESS_rx_cnt(void)

Returnerar antal bytes i läsbuffer.

void WIRELESS_rx_flush(void)

Töm läsbuffer.

int WIRELESS_write(unsigned char* pSend, int iCnt)

Skickar iCnt antal bytes från buffer pSend till coordinatorm.

void WIRELESS_tx_flush(void)

Väntar på att data skickats klart eller misslyckats.

void WIRELESS_Powerdown(void)

Försätt transceiver i strömsparläge.

void WIRELESS_Powerup(void)

Vakna upp ifrån strömsparläge.

int WIRELESS_EXT_address_match(QWORD a1, QWORD a2)

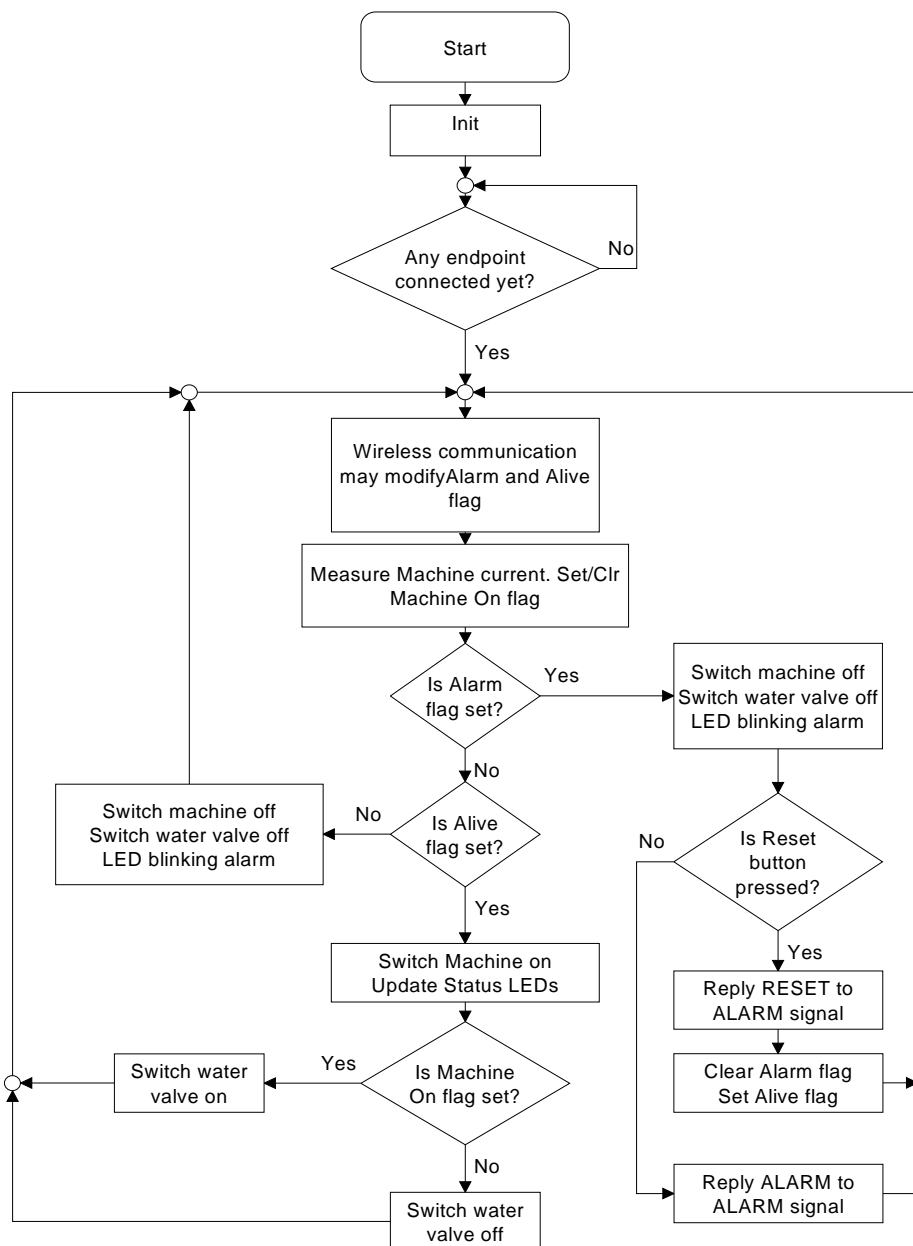
Jämför EUI-64 nätverks adresser.

6.3.0 Huvudprogram Coordinator

Flödesschema till huvudprogrammet för coordinator ges i figur 19.

Huvudprogrammet för coordinator ansvarar för att:

- Avgöra om maskinen används.
- Ta emot ALARM och PING signaler ifrån endpoints.
- Skicka RESET signal till endpoints om återställningknappen är nedtryckt.
- Indikera status via statusdioderna.
- Styra reläerna.



Figur 19 Flödesschema huvudprogram coordinator

Signalerna består av nollterminerade ASCII strängar och det finns tre möjliga:

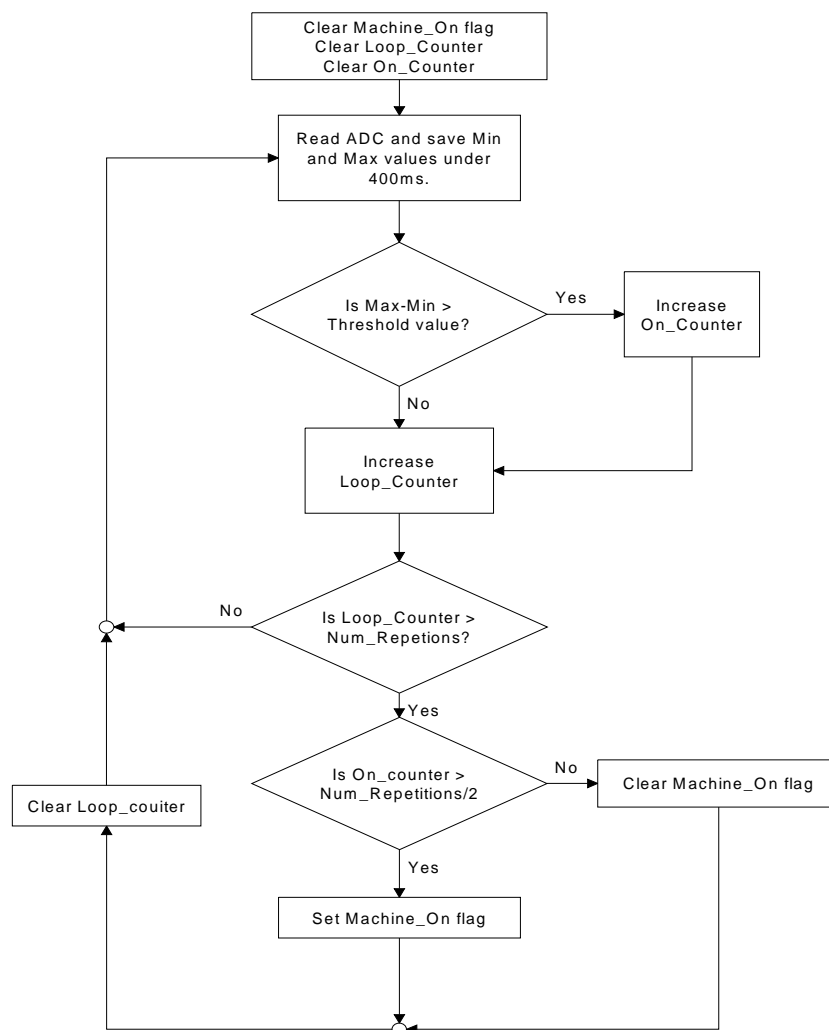
- ALARM skickas till coordinatorn av en endpoint om vattenläckage upptäcks, Signalen återskickas var 4:e sekund, tills alarmläge återställts.
- PING skickas till coordinatorn av en endpoint var 30:e sekund, om PING signaler från en endpoint upphör, antar coordinatorn att dess batteri är slut.
- RESET skickas till en endpoint av coordinatorn som svar på en ALARM signal om återställningsknappen är nertryckt, annars så svarar coordinatorn med ALARM.

Om en ALARM signal har tagits emot av coordinatorn, så sätts alarmflaggan och strömmen till maskinen och vattenventilen bryts. Alarmstatus indikeras av statusdioderna.

Om en endpoint inte hört av sig till coordinatorn inom tidsramen för tre PING signaler, så nollställs aliveflaggan(batterislut flagga) och strömmen till maskinen och vattenventilen bryts. Batterislut läge indikeras av statusdioderna.

För att avgöra om maskinen används mäts strömmen till maskinen.

Maskinen antas användas om den förbrukar mer än 15W, men under vissa tidsperioder kan effektförbrukningen sjunka under detta värde trots att maskinen används, så det behövs en viss tröghet i systemet. En speciell algoritm används för det ändamålet, för flödesschema se figur 20.



Figur 20 Flödesschema maskin algorithm

Eventuella offsetfel ifrån elektroniken påverkar inte algoritmens funktion.

Algoritmen är konfigurerbar i källkoden via ett antal defines:

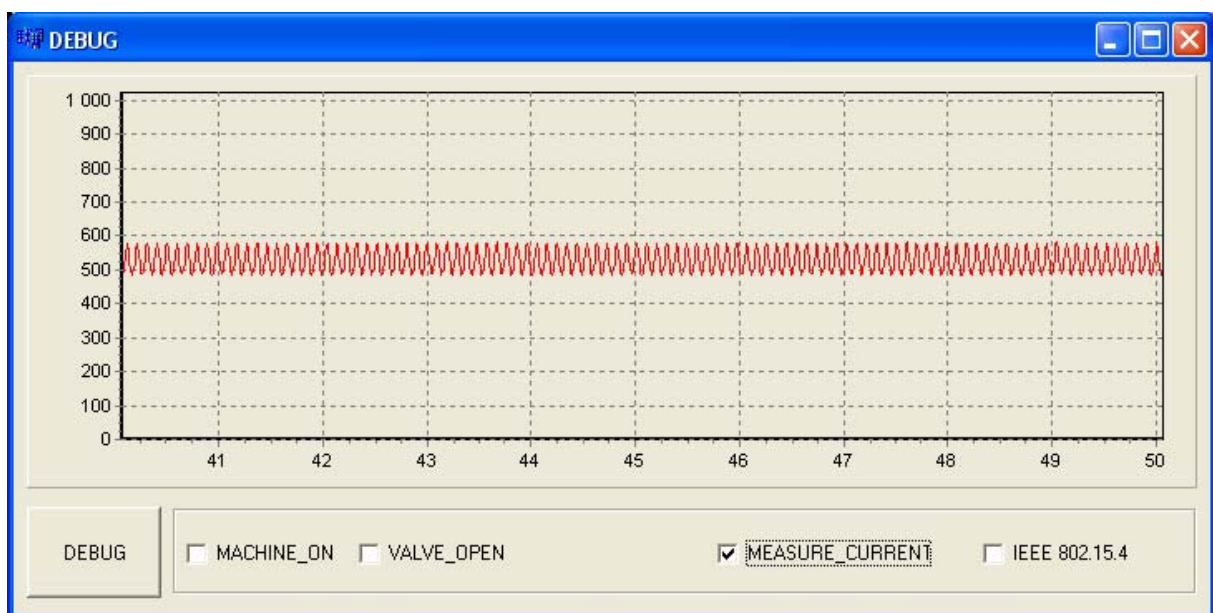
- **DETECT_TIME**
Specifierar tiden under vilken min och max värdena detekteras.
- **DETECT_THRESHOLD**
Detta värde motsvarar en viss ström och är detekteringsnivån.
- **DETECT_REPETITIONS_ON**
Då maskinstatus flagga inte är satt, används detta värde som antal repetitioner för att avgöra om maskinen används.
- **DETECT_REPETITIONS_OFF**
Då maskinstatus flagga är satt, används detta värde som antal repetitioner för att avgöra om maskinen är avstängd.

Enkelt sätt kan algoritmen ses som om att den avgör om maskinen är mer ”på” än ”av”, under en viss tidsperiod. Som källkoden är konfigurerad för tillfället, innebär det att det tar ca 12s att avgöra om maskinen har startat och det tar ca 36s att avgöra om maskinen är av. Då strömmen till maskinen ska brytas, försöker programmet se till att strömmen bryts vid nollgenomgången. Det görs genom att mäta min och max ström under ett några 50Hz perioder. Strömmen bryts då den passerar medelvärde av min och max.

I källkoden finns även ett avlusningsläge med tillhörande PC-program.

Läget kan aktiveras via PC-programmet, där man kan få en graf över uppmätt ström och se vad som mottagits från endpoints. Strömgrafan är tänkt att användas för att finjustera värdena till algoritmen under ett test med verklig maskin, exempel på graf visas i Figur 21.

Kommunikationen mellan PC och coordinator sker via serieporten.



Figur 21 PC-program strömmätning med testlast på 25W

Filer och funktioner tillhörande huvudprogrammet för coordinator:

Usart.c

Funktioner för att skicka/ta emot data på serieporten.

void USART_init(UINT32 baud)

Initiera USART0 med baudrate baud , 8 databitar och ingen paritet.

void USART_tx(unsigned char data)

Skicka en byte på USART0.

void USART_tx_buf(void* data , int ibytes)

Skicka ibytes bytes av buffer data på USART0.

unsigned char USART_chk_rx(void);

Undersök om en byte har mottagits på USART0. (Returnerar TRUE om en byte mottagits)

unsigned char USART_rx(void)

Vänta på att en byte mottagits på USART0 och returnera den.

void USART_rx_buf(void* data , int ibytes);

Läs ibytes bytes från USART0 och kopiera dem till buffer data.

void USART_flush(void);

Nollställ flagga för byte mottagits på USART0.

Ad.c

Funktioner för ADC omvandlaren.

void AD_init(void)

Initiera ADC omvandlaren med:

- PF0 som enda ADC ingång.
- Ingen förstärkning, singel ended mode.
- Intern spänningsreferens VCC.
- Högerjustera resultatet.
- Free running mode.

int AD_read(void)

Läs och returnera senaste värdet från ADC omvandlaren.

Tmr0.c

Funktioner för tidsmätning.

void TMR0_init(void);

Initiera Timer0 med overflow interrupt varje 4,096 ms.

void TMR0_delay_100ms(void)

Vänta ca 100ms och returnera.

UINT32 TMR0_getcount(void)

Returnera värdet av TMR0cnt.

SIGNAL(SIG_OVERFLOW0)

Interrupt funktion som anropas varje 4,096ms.Räknar upp TMR0cnt.

Btn.c

Funktion för att läsa återställningsknappen.

void BTN_init(void)

Initiera PD3 till ingång.

int BTN_chk(void)

Läs PD3 och returnera värdet.

Switch.c

Funktioner för att öppna/sluta reläerna

void SWITCH_init(void)

Initiera portarna.

void SWITCH_water_on(void)

Slut vattenreläet.

void SWITCH_water_off(void)

Bryt vattenreläet.

void SWITCH_machine_on(void)

Slut maskinreläet.

void SWITCH_machine_off(void)

Bryt maskinreläet.

Led.c

Funktioner för att tända/släcka status LEDs

void LED_init(void)

Initiera portarna.

void LED1_none(void)

Släck alla kommunikation status LEDs.

void LED1_green(void)

Tänd kommunikation grön status LED, släck röd ZigBee status LED.

void LED1_red(void)

Tänd kommunikation röd status LED, släck grön ZigBee status LED.

void LED2_none(void)

Släck alla Maskin status LEDs

void LED2_green(void)

Tänd grön Machine status LED, släck gul och röd Machine status LED.

void LED2_yellow(void)

Tänd gul Machine status LED, släck röd och grön Machine status LED.

void LED2_red(void)

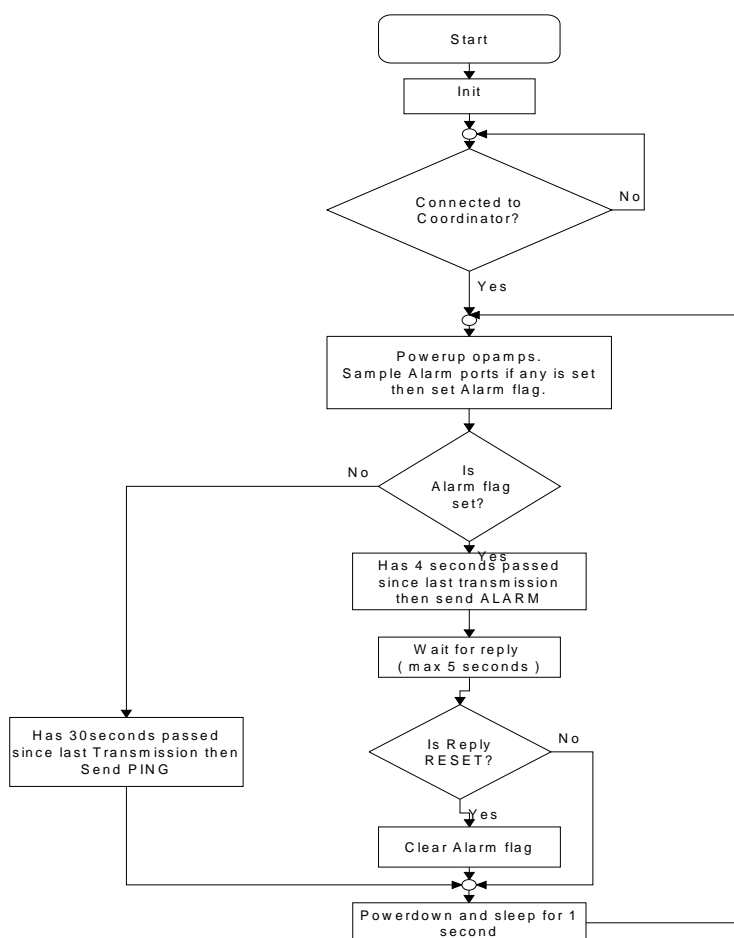
Tänd röd Machine status LED, släck grön och gul Machine status LED.

6.4.0 Huvudprogram Endpoint

Flödesschema till huvudprogrammet för endpoint ges i figur 22.

Huvudprogrammet för endpoint ansvarar för att:

- Läs alarm ingångarna varje sekund och uppdatera alarm flaggan.
- Skicka ALARM signal var 4:e sekund om alarm flaggan är satt.
- Skicka PING signal en gång var 30:e sekund.
- Nollställa alarmflagga om RESET signal mottagits.
- Förbruka så lite ström som möjligt.



Figur 22 Flödesschema endpoint.

För att minska strömförbrukningen i mikrokontrollern så befinner den sig normalt sätt i strömsparläge. En gång i sekunden går mikrokontroller ur strömsparläget och genomför normal programexekvering. Transceivern befinner sig även den normalt i strömsparläge, den går endast ut ur läget då den ska skicka eller förväntas ta emot data.

För att ytterligare minska strömförbrukningen i mikrokontrollern, stängs oanvända enheter som exempelvis ADC omvandlare och analog komparator av.

Då någon av alarmportarna går logiskt hög sätts en alarmflagga och mikrokontrollern börjar skicka ALARM signaler var 4:e sekund. Då en ALARM signal skickats förväntar sig

mikrokontrollern alltid ett svar från coordinatoren. Om svaret är en RESET signal så nollställs alarmflaggan. Då alarmflaggan inte är satt skickas en PING signal var 30:e sekund till coordinatoren, för att indikera att batteriet inte är slut.

Filer och funktioner tillhörande huvudprogrammet för endpoint:

Rtc.c

Funktioner för realtidsklocka.

Den här timern används för att väcka upp mikrokontrollern ur powersave läget.

void RTC_init(void);

Initierar Timer0 för att använda sig av en extern 32khz klocka och generera overflow interrupt en gång i sekunden.

UINT32 RTC_getcount(void)

Returnerar värdet av RTCcnt.

SIGNAL(SIG_OVERFLOW0)

Räknar upp RTCcnt.

Misc.c

Övriga funktioner.

void MISC_init_ports(void)

Initierar portarna.

void MISC_poweron_opamps(void)

Koppla på ström till operationsförstärkarna.

void MISC_poweroff_opamps(void)

Stäng av ström till operationsförstärkarna.

int MISC_read_alarm_pin(void)

Läs alarmportarna , returnerar TRUE om alarm.

void MISC_shutdown_notneeded(void)

Stänger av icke använda funktioner i mikrokontrollern för att minska strömförbrukningen.

- Analog komparator av
- ADC omvandlare av
- WDT av

void MISC_powersave(void)

Försätter mikrokontrollern i powersave mode.

Den vaknar upp av **SIG_OVERFLOW0** från timer0, vilket sker en gång i sekunden.

6.5.0 Möjliga förbättringar

Ingen av punkterna nedan behöver egentligen genomföras, de första tre kan dock vara en god idé.

- Om man ska använda sig av flera alarmenheter, kan det vara en god idé att implementera en mer avancerad typ av läsbuffer.
- Om Alarmenheten inte hittar coordinatoren vid uppstart, borde den med regelbundna tidsintervaller försöka ansluta igen.
- För att minska risken för framtida konflikter med ZigBee nätverk, kan hur anslutningen till nätverket sker göras mer likt ZigBee standarden.
- Ett sätt att låta coordinatoren och alarmenheter att automatiskt lära sig varandras adresser kan implementeras, men då krävs troligtvis en till tryckknapp på både coordinator och alarmenheter.

Den sista punkten kräver lite omtanke, då man inte vill råka ansluta till fel nätverk. Alla punkter utom den sista bör gå fort och vara enkla att implementera.

7 Resultat och slutsats

Prototypen har testats med en testlast och all funktionalitet har verifierats, dock har ingen test med flera alarmkort utförts. Ett test under verkliga förhållanden planeras, men hann inte genomföras innan denna rapport blev färdig. Det bör påpekas att den här lösningen inte använder sig av ZigBee standarden. Utan operationsförstärkare på alarmenheten uppmättes en medelström förbrukning på ca 500uA. Av dessa 500uA förbrukas ca 400uA av spänningsregulatorn på ZMR2A modulen, vilket gör den olämplig för batteridrift. Med en strömsnålare spänningsregulator bör man kunna nå under 100uA, vilket motsvarar ca 625 dagars drift med ett batteri på 1500mAh. När det gäller det totala priset för komponenter, ser man att den största delen utgörs av ZMR2A modulen. Det bör löna sig att söka efter billigare moduler baserade på ATmega och CC2420 kretsar, då koden får plats i billigare varianter av ATmega än ATmega128. Om ZigBee kompatibilitet önskas skulle jag föreslå ett byte till en annan arkitektur, där det finns en tillgänglig ZigBee stack att tillgå. Freescale[14] har exempelvis en SOC lösning MC13213 som kostar 43.88kr i kvantiteter om 100st (Farnell). Det skulle dock innebära att källkoden behöver konverteras till en ny mikrokontroller. Det finns även enklare ZigBee-liknande lösningar, som är licensfria och inte kräver certifiering exempelvis MiWi från Microchip. Slutligen om det räcker med att använda sig av en alarmenhet, kan det vara en nackdel med att använda sig av IEEE 802.15.4 standarden. Den kräver nämligen stor plats i mikrokontrollern, om man istället använder sig av enklare 2.4GHz transceiver kretsar som exempelvis CC2550[21] eller nRF2401[22], kan man använda sig av betydligt billigare mikrokontrollers. Då bör exempelvis en Atmel ATmega16 eller kanske tom en ATTiny räcka, utan att några större ändringar av källkoden måste genomföras. Egentligen krävs det då nästan bara mindre ändringar i källkoden för den trådlösa kommunikationen.

.

8 Komponentkostnad ny prototyp

I nedanstående tabeller ges komponentkostnaderna för de olika kretskorten.

Den billigare transformatorn i figur 24 bör testas i en prototyp, då den kommer att användas nära sitt maximala effektuttag. Priset för ZM2RA modulerna är inte deras inköpspris, utan enbart en ungefärlig komponentkostnad. Slutligen i figur 27 summeras priserna, och en ungefärlig totalkostnad ges.

Schema	Beskrivning	Värde/namn	Om 1st	Om 100st		Artikelnr
R1	SMD 1206	16k	1,81 kr	0,38 kr	ELFA[15]	60-197-80
R2	SMD 1206	10k	1,81 kr	0,38 kr	ELFA	60-197-31
R3	SMD 1206	1k	1,81 kr	0,38 kr	ELFA	60-194-91
R4	SMD 1206	10k	1,81 kr	0,38 kr	ELFA	60-197-31
R5	SMD 1206	300k	1,81 kr	0,38 kr	ELFA	60-200-85
R6	SMD 1206	4.7k	1,81 kr	0,38 kr	ELFA	60-196-57
R7	SMD 1206	4.7k	1,81 kr	0,38 kr	ELFA	60-196-57
R8	SMD 1206	4.3k	1,81 kr	0,38 kr	ELFA	60-196-40
R9	SMD 1206	10k	1,81 kr	0,38 kr	ELFA	60-197-31
R10	SMD 1206	10k	1,81 kr	0,38 kr	ELFA	60-197-31
C1	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C2	Elektrolyt	1000u 25V	8,64 kr	6,24 kr	ELFA	67-321-84
C3	SMD Elektrolyt	470u 10V	6,77 kr	3,76 kr	ELFA	67-321-19
C4	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C5	Plastfilm	100n	5,05 kr	3,14 kr	ELFA	65-557-59
C6	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C7	SMD Elektrolyt	22u	1,70 kr	0,94 kr	ELFA	67-321-35
C9	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C10	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C11	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C12	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
Q1	FET	FDV301N	1,87 kr	1,05 kr	ELFA	71-096-48
Q2	FET	FDV301N	1,87 kr	1,05 kr	ELFA	71-096-48
RL1	Relä	G5RL-1A-E-HR-5	20,74 kr	14,93 kr	Digikey	Z2579-ND
RL2	Relä	PE014005	28,90 kr	18,05 kr	Digikey	PB353-ND
D1	Zenerdiod	BZV55-B4V7	1,92 kr	0,53 kr	ELFA	70-319-41
D2	Zenerdiod	BZV55-B3V3	1,92 kr	0,53 kr	ELFA	70-319-25
D3	Diod	RS1M	2,87 kr	0,95 kr	ELFA	70-001-69
D4	Diod	RS1M	2,87 kr	0,95 kr	ELFA	70-001-69
IC1	Strömsensor	LTS15	132,98 kr	113,01 kr	Farnell	3114557
IC2	Sp-regulator	LM340	9,04 kr	4,81 kr	Digikey	LM340MP-5.0CT-ND
IC3	Opamp	OPA2337UA	10 kr	6,12 kr	Digikey	OPA2337UA-ND
JP2	Connector	M50-3501542	12,85 kr	8,77 kr	Mouser[18]	855-M50-3501542
JP4	JTAG connector	2.54mm 2x5	5,33 kr	2,57 kr	ELFA	43-702-27
T1	Transformator	HAHN EI305 2893	49,50 kr	24,70 kr	ELFA	56-143-18
R14	Metalloxid varistor	S07K275	0,48 kr	0,33 kr	Digikey	495-1408-ND
R12	Metalloxid varistor	S20K275	4,69 kr	3,33 kr	Digikey	495-1417-ND
R13	Metalloxid varistor	S20K275	4,69 kr	3,33 kr	Digikey	495-1417-ND
			343,90 kr	224,140 kr		

Figur 23 Styrenhet med LTS15 strömsensor och original transformator.

Schema	Beskrivning	Värde/namn	Om 1st	Om 100st		Artikelnr
R1	SMD 1206	16k	1,81 kr	0,38 kr	ELFA	60-197-80
R2	SMD 1206	10k	1,81 kr	0,38 kr	ELFA	60-197-31
R3	SMD 1206	1k	1,81 kr	0,38 kr	ELFA	60-194-91
R4	SMD 1206	10k	1,81 kr	0,38 kr	ELFA	60-197-31
R5	SMD 1206	300k	1,81 kr	0,38 kr	ELFA	60-200-85
R6	SMD 1206	4.7k	1,81 kr	0,38 kr	ELFA	60-196-57
R7	SMD 1206	4.7k	1,81 kr	0,38 kr	ELFA	60-196-57
R8	SMD 1206	4.3k	1,81 kr	0,38 kr	ELFA	60-196-40
R9	SMD 1206	10k	1,81 kr	0,38 kr	ELFA	60-197-31
R10	SMD 1206	10k	1,81 kr	0,38 kr	ELFA	60-197-31
C1	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C2	Elektrolyt	1000u 25V	8,64 kr	6,24 kr	ELFA	67-321-84
C3	SMD Elektrolyt	470u 10V	6,77 kr	3,76 kr	ELFA	67-321-19
C4	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C5	Plastfilm	100n	5,05 kr	3,14 kr	ELFA	65-557-59
C6	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C7	SMD Elektrolyt	22u	1,70 kr	0,94 kr	ELFA	67-321-35
C9	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C10	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C11	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C12	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
Q1	FET	FDV301N	1,87 kr	1,05 kr	ELFA	71-096-48
Q2	FET	FDV301N	1,87 kr	1,05 kr	ELFA	71-096-48
RL1	Relä	G5RL-1A-E-HR-5	20,74 kr	14,93 kr	Digikey	Z2579-ND
RL2	Relä	PE014005	28,90 kr	18,05 kr	Digikey	PB353-ND
D1	Zenerdiod	BZV55-B4V7	1,92 kr	0,53 kr	ELFA	70-319-41
D2	Zenerdiod	BZV55-B3V3	1,92 kr	0,53 kr	ELFA	70-319-25
D3	Diod	RS1M	2,87 kr	0,95 kr	ELFA	70-001-69
D4	Diod	RS1M	2,87 kr	0,95 kr	ELFA	70-001-69
IC1	Strömsensor	ACS712-20	19,72 kr	12,28 kr	Digikey	620-1191-1-ND
IC2	Sp-regulator	LM340	9,04 kr	4,81 kr	Digikey	LM340MP-5.0CT-ND
IC3	opamp	OPA2337UA	10 kr	6,12 kr	Digikey	OPA2337UA-ND
JP2	Connector	M50-3501542	12,85 kr	8,77 kr	Mouser	855-M50-3501542
JP4	JTAG connector	2.54mm 2x5	5,33 kr	2,57 kr	ELFA	43-702-27
T1	Transformator	BV030-7341.0	19,04 kr	8,45 kr	Digikey	567-1038-5-ND
R14	Metalloxid varistor	S07K275	0,48 kr	0,33 kr	Digikey	495-1408-ND
R12	Metalloxid varistor	S20K275	4,69 kr	3,33 kr	Digikey	495-1417-ND
R13	Metalloxid varistor	S20K275	4,69 kr	3,33 kr	Digikey	495-1417-ND
			200,18 kr	107,160 kr		

Figur 24 Styrenhet med ACS712-20 strömsensor och billigare transformator.

Schema	Beskrivning	Värde/namn	Om 1st	Om 100st		Artikelnr
C1	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C6	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C11	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C12	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
IC2	Opamp	OPA2337UA	10 kr	6,12 kr	Digikey	OPA2337UA-ND
JTAG	JTAG connector	2.54mm 2x5	5,33 kr	2,57 kr	ELFA	43-702-27
R1	SMD 1206	200k	1,81 kr	0,38 kr	ELFA	60-200-44
R2	SMD 1206	470k	1,81 kr	0,38 kr	ELFA	60-201-35
R3	SMD 1206	100k	1,81 kr	0,38 kr	ELFA	60-199-70
R4	SMD 1206	100k	1,81 kr	0,38 kr	ELFA	60-199-70
R5	SMD 1206	200k	1,81 kr	0,38 kr	ELFA	60-200-44
R6	SMD 1206	100k	1,81 kr	0,38 kr	ELFA	60-199-70
R7	SMD 1206	470k	1,81 kr	0,38 kr	ELFA	60-201-35
R8	SMD 1206	100k	1,81 kr	0,38 kr	ELFA	60-199-70
R9	SMD 1206	100k	1,81 kr	0,38 kr	ELFA	60-199-70
R10	SMD 1206	100k	1,81 kr	0,38 kr	ELFA	60-199-70
R11	SMD 1206	100k	1,81 kr	0,38 kr	ELFA	60-199-70
U\$8	FET	FDV302P	1,93 kr	1,09 kr	ELFA	71-096-14
U\$2	Connector	M50-3501542	12,85 kr	8,77 kr	Mouser	855-M50-3501542
			56,46 kr	23,432 kr		

Figur 25 Alarmkort.

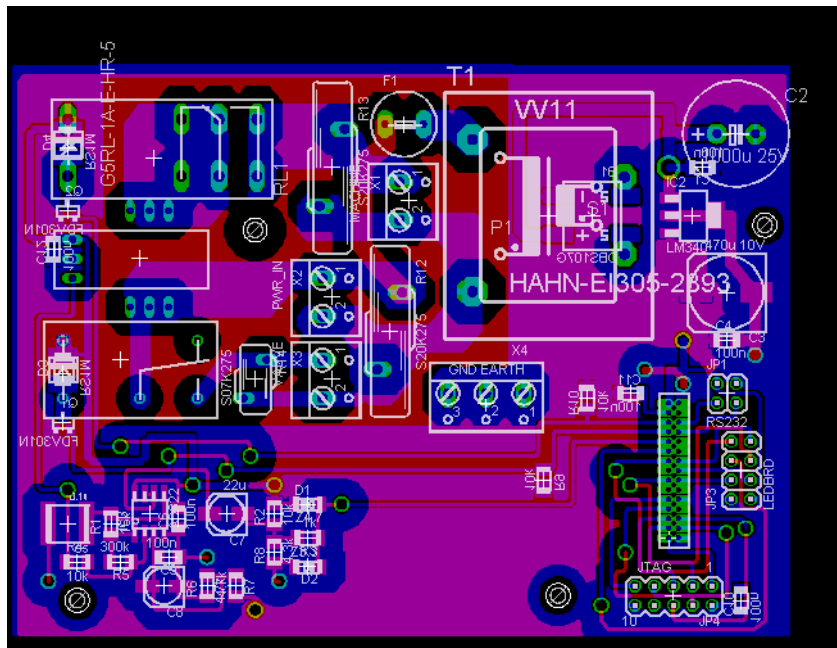
C1	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
C2	SMD 1206	100n	1,56 kr	0,18 kr	ELFA	65-776-88
D2	Röd LED	EL202ID	2,29 kr	0,48 kr	ELFA	75-023-96
D3	gul LED	EL204-10UYD	2,29 kr	0,48 kr	ELFA	75-006-22
D4	grön LED	EL202GD	2,10 kr	0,48 kr	ELFA	75-006-14
D5	Röd LED	EL202ID	2,29 kr	0,48 kr	ELFA	75-023-96
D6	grön LED	EL202GD	2,20 kr	0,48 kr	ELFA	75-006-14
U\$1	FET	FDV301N	1,87 kr	1,05 kr	ELFA	71-096-48
U\$2	FET	FDV301N	1,87 kr	1,05 kr	ELFA	71-096-48
U\$3	FET	FDV301N	1,87 kr	1,05 kr	ELFA	71-096-48
U\$4	FET	FDV301N	1,87 kr	1,05 kr	ELFA	71-096-48
U\$5	FET	FDV301N	1,87 kr	1,05 kr	ELFA	71-096-48
R1	SMD 1206	1k	1,81 kr	0,38 kr	ELFA	60-194-91
R2	SMD 1206	1k	1,81 kr	0,38 kr	ELFA	60-194-91
R3	SMD 1206	1k	1,81 kr	0,38 kr	ELFA	60-194-91
R4	SMD 1206	1k	1,81 kr	0,38 kr	ELFA	60-194-91
R5	SMD 1206	56k	1,81 kr	0,38 kr	ELFA	60-199-13
R7	SMD 1206	100k	1,81 kr	0,38 kr	ELFA	60-199-70
R9	SMD 1206	1k	1,81 kr	0,38 kr	ELFA	60-194-91
S1	Strömställare	B3F-1000	8,97 kr	7,51 kr	Farnell	176432
			45,28 kr	18,17 kr		

Figur 26 Dotterkort för statusdioder och knapp.

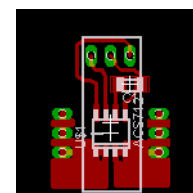
Beskrivning	Pris
Styrenhet	107,16 kr
Dotterkort	18,17 kr
Alarmenhet	23,43 kr
ZM2RA moduler	263,16 kr
	411,92 kr

Figur 27 Total komponentkostnad.

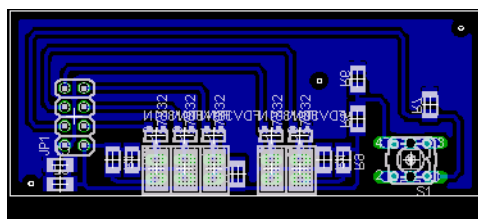
9 PCB



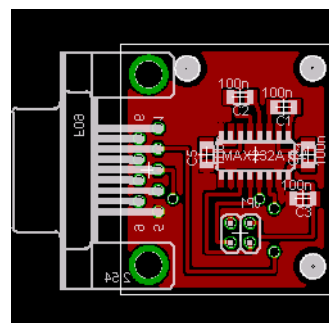
Figur 28 PCB Styrenhet



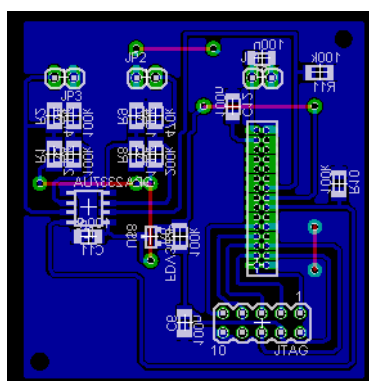
Figur 29 PCB ACS712-20



Figur 30 PCB dotterkort



Figur 31 PCB RS232kort



Figur 32 PCB nytt alarmkort

10 Referenser

- [1] Allan R.Hambley. Electronics Second Edition ISBN 0-13-085765-3
- [2] <http://www.ti.com/lit/gpn/cc2420>
- [3] http://www.atmel.com/dyn/resources/prod_documents/doc2467.pdf
- [4] <http://www.elfa.se/pdf/37/03707145.pdf>
- [5] <http://www.elfa.se/pdf/37/03721404.pdf>
- [6] <http://www.elfa.se/pdf/71/07109648.pdf>
- [7] <http://www.elfa.se/pdf/73/730/07309206.pdf>
- [8] <http://www.elfa.se/pdf/72/07200520.pdf>
- [9] http://www.nxp.com/acrobat_download/applicationnotes/AN10384_1.pdf
- [10] <http://standards.ieee.org/getieee802/download/802.15.4-2003.pdf>
- [11] <http://focus.ti.com/docs/prod/folders/print/reg102-3.html>
- [12] <http://www.antenova.com/?id=408>
- [13] http://www.allegromicro.com/en/Products/Part_Numbers/0712/0712.pdf
- [14] http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=ZRP-1&nodeId=01J4Fs25657103
- [15] www.elfa.se
- [16] www.digikey.com
- [17] www.farnell.com
- [18] www.mouser.com
- [19] www.zigbee.org
- [20] www.epcos.com
- [21] <http://focus.ti.com/docs/prod/folders/print/cc2550.html>
- [22] <http://www.nordicsemi.no/index.cfm?obj=product&act=display&pro=64>
- [23] <http://standards.ieee.org/faqs/OUI.html#q12>
- [24] <http://www.zigbee.org/en/join/benefits.asp>
- [25] <http://www.elfa.se/pdf/60/06029573.pdf>
- [26] <http://www.epcos.com/web/generator/Web/Sections/ProductCatalog/NonlinearResistors/Varistors/MonolithicAndLeadedVaristors/Page,templateId=render,locale=en.html> (equation overview)

11 Förkortningar

SMD	Surface mount Devices
UART	Universal Asynchronous Receiver Transmitter
ADC	Analog to Digital Converter
EUI-64	64-bits Extended Unique Identifier
MAC	Medium Access Layer
PHY	Physical Layer
IEEE	Institute of Electrical and Electronics Engineers
JTAG	Joint Test Action Group
SOC	System On Chip
WDT	Watchdog Timer
RFD	Reduced Functional Device
FFD	Full Function Device
OUI	Organizationally Unique Identifier
CSMA-CA	Carrier Sense Multiple Access with Collision Avoidance
TRIAC	Triode for Alternating Current
PCB	Printed Circuit Board
PANID	Personal Area Network Identifier
API	Application Program Interface
SRAM	Static Random Access Memory
EEPROM	Electrical Erasable Programmable Readonly Memory

12 Använd hård och mjukvara

Cadsoft EAGLE 4.16r2 Light

<http://www.cadsoft.de>

Atmel AVRStudio

http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725

WinAVR

<http://winavr.sourceforge.net/>

Olimex AVR-JTAG-USB

<http://www.olimex.com/dev/pdf/avr-usb-jtag.pdf>