

Sammanfattning

Denna rapport behandlar vårt projekt "Blue Key". "Blue Key" är tänkt att vara själva avlarmningsfunktionen i t.ex. en larmapplikation. Systemet är uppbyggt av ett antal olika enheter varav de tre mest väsentliga är en enchipsdator, en bluetoothmodul och en mobiltelefon med bluetooth. Enchipsdatorn använder bluetoothmodulen för att söka efter användare som finns inom en förutbestämd radie. När väl en användare är inom avsökningsområdet jämför datorn telefonens ID med sparade ID och om dessa stämmer överens skickar datorn "larma av". Det finns stora möjligheter till att utveckla och utöka "Blue Key" funktionerna. Man kan införa systemet i "smarta hem" så att då användaren närmar sig hemmet låses dörren upp samtidigt som de lampor och den radiokanalen användaren har angett går igång. Det finns i princip inga begränsningar för konceptet "Blue Key", endast fantasin sätter gränser.

Innehållsförteckning

1	Presentation av problem	3
2	Bakgrund	3
3	Teori	4
3.1	Larm	4
3.2	Enchipsdator	4
3.3	Bluetoothmodul	5
3.4	Mobiltelefon	5
3.5	Systemet som helhet	5
4	Genomförande	6
5	Funktionsbeskrivning	8
6	Resultat och diskussion	9
7	Källförteckning	10
8	Appendix	11

1 Presentation av problem

Tanken är att byta ut nyckeln i ett larmsystem till något som de flesta redan bär med sig, mobiltelefonen. Att använda en mobiltelefons Bluetooth för att automatiskt larma/larma av när man närmar sig eller avlägsnar sig från systemet.

2 Bakgrund

Dagens mobiltelefoner får fler och fler funktioner, men det är inte alla som utnyttjas till sin fulla potential. En del tillverkares telefoner är nu mer utrustade med en radiomodul, en så kallad Bluetoothmodul. Denna modul kan användas för att överföra data upp till 700kBps eller för att fjärrstyra telefonen. Det finns två effektklasser. Det som skiljer dessa är räckvidden, 10m eller 100m. Varje Bluetoothmodul har en unik adress precis som en IP-adress på Internet. Det är detta som utnyttjas i denna applikation.

3 Teori

Som bild 3.1 visar består applikationen av ett antal olika block:

- Larm
- Enchipsdator (MC)
- Bluetoothmodul
- Mobiltelefon

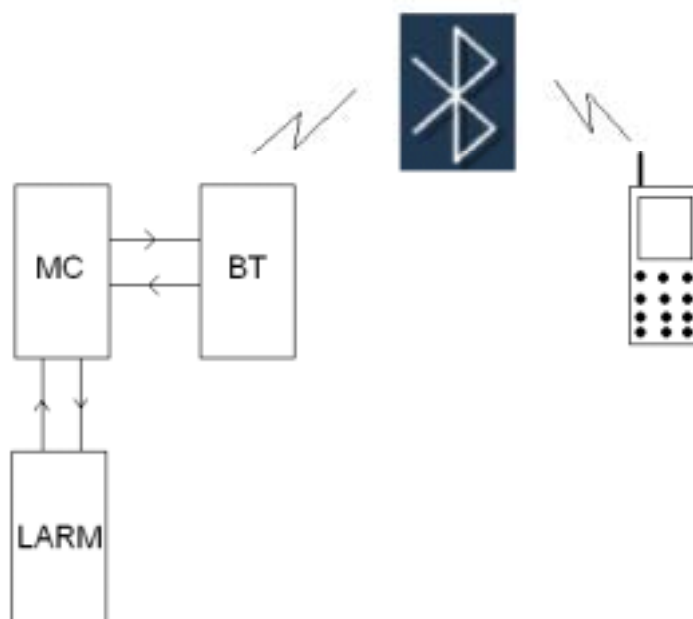


Bild 3.1 Funktionsbeskrivning.

3.1 Larm

Detta block består av i stort sett vilket kommersiellt larm som helst. Inkoppling av olika larm måste naturligtvis ske på olika sätt men möjligheten finns. Billarm är lite svårare att anpassa medan villalarm har mer flexibilitet då det gäller val av nyckelsystem.

3.2 Enchipsdator

Enchipsdatorn är en ATmega16. Denna har till uppgift att sköta kommunikationen med bluetoothmodulen och larmet. Kommunikationen med modulen sker via en serieport. ATmega16 har en inbyggd USART för RS-232 kommunikation. För närvarande sker larmning med en reläutgång men endast mjukvaran sätter begränsningar.

3.3 Bluetoothmodul

Modulen som används är en TDK "Blue2i serialmodule". Anledningen till att vi valde denna var att den kommunicerar med AT-kommandon och har ett RS-232 interface. Enda nackdelen med denna modul är att den måste kopplas in med en 40-pinnars "board to board" kontakt av typ Hirose DF12C med 0,5 mm delning se bild 3.2.

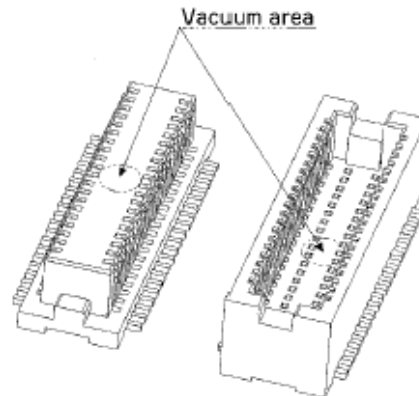


Bild 3.2 Board to Board kontakt DF12C

3.4 Mobiltelefon

Alla telefoner som har bluetooth kan användas som nycklar.

3.5 Systemet som helhet

Atmega16 skickar kommandon till bluetoothmodulen som då börjar söka av om det finns några BT enheter i närheten. Om det finns det skickar den dessa BT-adresser till Atmega16. Enchipsdatorn kollar mot sitt minne för att se om några av dessa BT-adresser är giltiga nycklar. Om så är fallet av aktiveras larmet. När sedan BT-enheten kommer för långt bort från systemet kommer denna inte hittas av BT-modulen vilket innebär att enchipsdatorn kommer att aktivera larmet.

4 Genomförande

Eftersom bluetooth delen var helt ny för oss började vi med att koppla upp modulen mot en PC via comporten. Det vi inte hade räknat med var att man måste koppla in sig via den ovan nämnda "board to board" kontakten. Detta visade sig vara den verkliga utmaningen i detta projekt. Vi fick lov att rita ett kretskort i ett CAD program för att omvandla 0,5 mm delning till 2,54 mm delning som vi hade på labkortet. Att löda dit DF12C var inte det lättaste men efter åtskilliga försök lyckades vi att få ett fungerande kort. Sedan var det bara att koppla in modulen enligt bild 4.1 och skicka AT-kommandon via ett terminalfönster. Genom att göra detta kunde vi komma fram till vilka inställningar och kommandon som måste utföras.

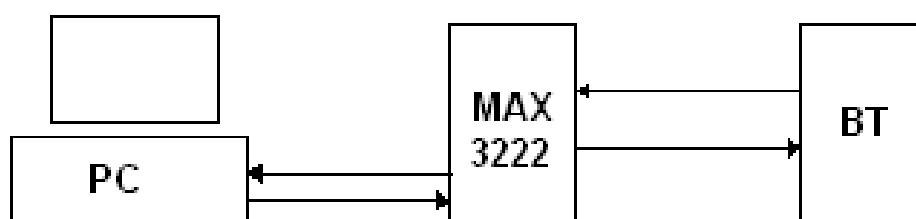


Bild 4.1 Inkoppling PC till bluetoothmodul.

Anledningen till att MAX3222 används är för att omvandla RS-232 standardens ± 12 V till bluetooth modulens 3,3 V TTL.

Man kan genom att ändra olika register i modulen ställa in en hel del olika egenskaper. Man kan välja sändareffekt, söktid, kryptering osv.

När vi väl fått modulen att fungera som vi ville så började vi att koppla upp Atmega16 med JTAG ICE. Även denna kopplade vi mot PC:ns comport för att underlätta utvecklingen av programmet i enchipdatoren. Här behövdes också spänningsomvandling från ± 12 V till 5 V, detta gjordes med kretsen MAX232.

Vi kopplade även in en LCD till enchipdatoren, svårigheten med att koppla in displayen var att få timingen på de olika signalerna att bli rätt.

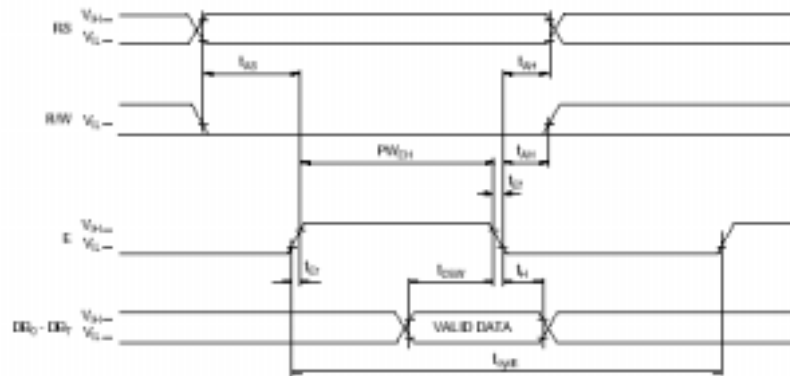


Bild 4.2 Timingdiagram för skrivoperation till LCD

Som synes i bild 4.2 måste man sätta RS och R/W innan E sätts. E måste sedan ligga hög då data skickas, samt minst 0,5 ms.

5 Funktionsbeskrivning

Programmet består av två delar:

- Lägg till BT-enheter som nycklar
- Normalläge, leta efter enheter

Den delen som lägger till nycklar söker först efter enheter i närheten och visar sedan enheternas smeknamn på systemets display. Användaren har då möjlighet att välja Skip eller Confirm för varje enhet som BT modulen hittat se bild 5.1. Om Confirm valts läggs BT adressen för vald enhet till som nyckel. Om man däremot trycker på Skip kommer nästa smeknamn upp i displayen. När man är färdig med att lägga till nycklar trycker man på Mode knappen och programmet går över i normalläge.

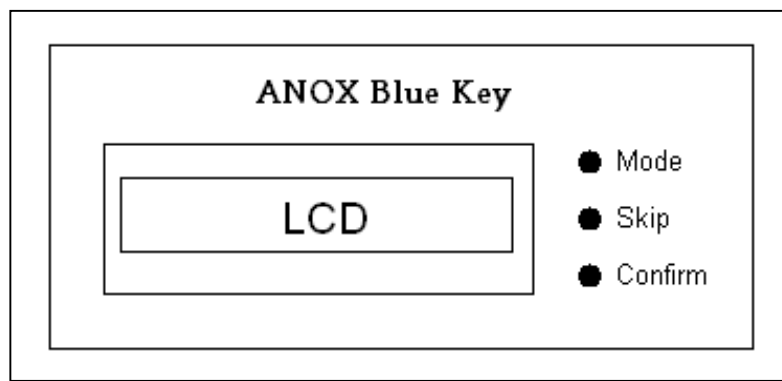


Bild 5.1 Användargränssnitt.

6 Resultat och diskussion

Vid rapportens skrivande finns en fungerande prototyp dock med begränsad prestanda. Under projektets gång har vi fått lära oss C-programmering för att kunna göra programmet till ATmega16. För att bli klara i tid valde vi att endast tillåta en användare, vilket förenklade programmeringen. Detta är inget större problem då man med lite mer tid ganska lätt kan utöka programmet till att stödja fler användare. Ett annat problem som vi stött på är att "board to board" kontaktingen är relativt instabil. Detta leder till att enchipdatorn inte alltid får kontakt med BT modulen. En lösning på detta problem är att beställa ett färdigt kretskort med DF12C monterad från en kretskortstillverkare.

"Blue Key" är byggt så att man kan bestämma avståndet för vilket systemet skall upptäcka nycklarna. Detta gör det möjligt att anpassa applikationen till användaren. Då vi pratar om nycklar menar vi inte nödvändigtvis en mobiltelefon. Man kan använda i stort sett alla enheter med Bluetooth som nyckel t.ex. en handdator. Tanken med "Blue Key" är att man ska få en sak mindre att bära med sig, alltså väljer man en apparat med Bluetooth som man använder dagligen.

7 Källförteckning

ATmega16 datablad (2004-05-12),

http://www.atmel.com/dyn/resources/prod_documents/doc2466.pdf

Brian W. Kernighan - Dennis M Ritchie (1988), The C programming language 2nd edition, Prentice Hall

Sharp LCD datablad (2004-05-12),

<http://www.it.lth.se/datablad/display/LCD.pdf>

MAX232 datablad (2004-05-12),

<http://www.it.lth.se/datablad/periphery/communication/max232-233.pdf>

MAX3222 datablad (2004-05-12),

<http://www.elfa.se/pdf/73/732/07322845.pdf>

Bluetooth Intelligent Serial Module AT Command Set (2004-05-12),

<http://www.blu2i.com/download/TDK%20Bluetooth%20AT%20Command%20set.pdf>

Blu2i module user guide (2004-05-12),

<http://www.blu2i.com/download/blu2i%20Module%20-%20User%20Guide%282%29.pdf>

8 Appendix