

Appendix B: The transmitter source code

```
/******  
Chip type      : ATmega8535L  
Program type   : Application  
Clock frequency : 8.000000 MHz  
Memory model   : Small  
External SRAM size : 0  
Data Stack size : 128  
*****/  
#include <mega8535.h>  
#include <delay.h>  
// Declare your global variables here  
int x;  
void main(void)  
{  
// Declare your local variables here  
// Input/Output Ports initialization  
// Port A initialization  
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In  
Func0=Out  
// State7=P State6=P State5=P State4=P State3=P State2=P State1=P  
State0=0  
PORTA=0xFE;  
DDRA=0x01;  
// Port B initialization  
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In  
Func0=In  
// State7=P State6=P State5=P State4=P State3=P State2=P State1=P  
State0=T  
PORTB=0xFE;  
DDRB=0x00;  
// Port C initialization  
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In  
Func0=In  
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T  
State0=T  
PORTC=0x00;  
DDRC=0x00;  
// Port D initialization
```

```
// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out
Func1=Out Func0=Out
// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0
State0=0
PORTD=0x00;
DDRD=0xFF;
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer 1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;
// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer 2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
```

```

OCR2=0x00;
// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;
// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;
// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
// Analog Comparator Output: Off
ACSR=0x80;
SFIOR=0x00;

while (1)
{
    x=PINB.0;
    PORTD.0=PINC.0;
    PORTD.1=PINC.1;
    PORTD.2=PINC.2;
    PORTD.3=PINC.3;
    PORTD.4=PINC.4;
    PORTD.5=PINC.5;
    PORTD.6=PINC.6;
    PORTD.7=PINC.7;
    if (x==0x01)
    {
        PORTA.0=0x01;
        delay_ms(25);
        PORTA.0=PORTD.1;
        delay_ms(25);
        PORTA.0=PORTD.2;
        delay_ms(25);
        PORTA.0=PORTD.3;
        delay_ms(25);
        PORTA.0=PORTD.4;
        delay_ms(25);
        PORTA.0=PORTD.5;
    }
}

```

```
        delay_ms(25);
        PORTA.0=PORTD.6;
        delay_ms(25);
        PORTA.0=PORTD.7;
        delay_ms(25);
    }
    PORTA.0=0x00;
};
}
```

Appendix C: The receiver source code

```
*****/
Chip type      : ATmega8535L
Program type   : Application
Clock frequency : 8.000000 MHz
Memory model   : Small
External SRAM size : 0
Data Stack size : 128
/*****
#include <mega8535.h>
#include <delay.h>
//Declare your global variables here
int x;
void main(void(
}
//Declare your local variables here
//Input/Output Ports initialization
//Port A initialization
//Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out
Func1=Out Func0=Out
//State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0
State0=0
PORTA=0x00;
DDRA=0xFF;
//Port B initialization
//Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
Func0=In
//State7=P State6=P State5=P State4=T State3=P State2=P State1=P
State0=P
PORTB=0xEF;
DDRB=0x00;
//Port C initialization
//Func7=In Func6=In Func5=In Func4=In Func3=Out Func2=In Func1=In
Func0=In
//State7=P State6=P State5=P State4=P State3=0 State2=P State1=P
State0=P
PORTC=0xF7;
DDRC=0x08;
//Port D initialization
```

```
//Func7=In Func6=In Func5=In Func4=In Func3=In Func2=Out Func1=In
Func0=In
//State7=P State6=P State5=P State4=P State3=P State2=0 State1=P
State0=P
PORTD=0xFB;
DDRD=0x04;
//Timer/Counter 0 initialization
//Clock source: System Clock
//Clock value: Timer 0 Stopped
//Mode: Normal top=FFh
//OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;
//Timer/Counter 1 initialization
//Clock source: System Clock
//Clock value: Timer 1 Stopped
//Mode: Normal top=FFFFh
//OC1A output: Discon.
//OC1B output: Discon.
//Noise Canceler: Off
//Input Capture on Falling Edge
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;
//Timer/Counter 2 initialization
//Clock source: System Clock
//Clock value: Timer 2 Stopped
//Mode: Normal top=FFh
//OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
```

```

OCR2=0x00;
//External Interrupt(s) initialization
//INT0: Off
//INT1: Off
//INT2: Off
MCUCR=0x00;
MCUCSR=0x00;

//Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;
//Analog Comparator initialization
//Analog Comparator: Off
//Analog Comparator Input Capture by Timer/Counter 1: Off
//Analog Comparator Output: Off
ACSR=0x80;
SFIOR=0x00;

while (1)
{
    x=PINB.4;
    if (x==0x01)
    {
        PORTA.0=0x01
        delay_ms(25);
        x=PINB.4;
        PORTA.1=x;
        delay_ms(25);
        x=PINB.4
        PORTA.2=x;
        delay_ms(25);
        x=PINB.4;
        PORTA.3=x;
        delay_ms(25);
        x=PINB.4;
        PORTA.4=x;
        delay_ms(25);
        x=PINB.4;
        PORTA.5=x;
        delay_ms(25);
        x=PINB.4;
    }
}

```

```
        PORTA.6=x;
        delay_ms(25);
        x=PINB.4;
        PORTA.7=x;
        delay_ms(25);
    }
};
}
```