## Appendix A ;

Microprocessor program for the motor controller ( PIC 16f628)

```
;*****************************************************************
;                                                                *
;   Filename:      Fan controller pwm.asm                        *
;   Date:          23/8/2010                                     *
;   File Version:  5                                             *
;                                                                *
;   Author:        M.D.A.K.Wijerathna                            *
;   Project:       Development of a remote control hum less fan dimmer *
;                                                                *
;*****************************************************************
;                                                                *
;   Files required:                                              *
;                                                                *
;*****************************************************************
;                                                                *
;   Notes:  The program is developed for 72%, 76%, 80% percentage line *
;           voltages.                                            *
;                                                                *
;*****************************************************************

        list    p=16f628A        ; list directive to define processor
        #include <p16F628A.inc>      ; processor specific variable definitions

        errorlevel  -302         ; suppress message 302 from list file

__CONFIG   _CP_OFF & _DATA_CP_OFF & _LVP_OFF & _BOREN_OFF &
_MCLRE_OFF & _WDT_OFF & _PWRTE_ON & _INTOSC_OSC_NOCLKOUT

; '__CONFIG' directive is used to embed configuration word within .asm file.
; The lables following the directive are located in the respective .inc file.
; See data sheet for additional information on configuration word settings.


;***** VARIABLE DEFINITIONS*****************************************
w_temp      EQU    0x71      ; variable used for context saving
status_temp  EQU    0x72      ; variable used for context saving

;*****************************************************************
        ORG    0x000          ; processor reset vector
        goto   main           ; go to beginning of program
```

```
        ORG    0x004                ; interrupt vector location
        movwf  w_temp               ; save off current W register contents
        movf   STATUS,w             ; move status register into W register
        movwf status_temp           ; save off contents of STATUS register

; isr code can go here or be located as a call subroutine elsewhere

        nop
        nop

        movf   status_temp,w        ; retrieve copy of STATUS register
        movwf STATUS                ; restore pre-isr STATUS register contents
        swapf  w_temp,f
        swapf  w_temp,w             ; restore pre-isr W register contents
        retfie                      ; return from interrupt

main
                bsf             STATUS,RP0
                movlw b'10000000'
                movwf OPTION_REG
                clrf    INTCON
                clrf    TRISB
                clrf    TRISA
;*****************************************************
                movlw b'00000010'
                movwf PIE1

                movlw d'96'
                movwf PR2

;*****************************************************

                bcf             STATUS,RP0
                clrf    PORTB
                movlw 0x07
                movwf CMCON
                clrf    24
;*****************************************************
                movlw b'00000000'
                movwf PIR1

                clrf    TMR2
                movlw b'00000000'
                movwf T2CON
;*****************************************************
                clrf    25  ;
```

```
        clrf    26
        clrf    27
        clrf    28
        clrf    29
        clrf    30
        clrf    31
        clrf    32
        clrf    34
        clrf    35

        clrf    40
        clrf    41
        clrf    45


;*********************************************************
;               Pulse Generation
;************************************** 1 **************
@_1

        bsf             PORTA,03    ;SD pin
        bsf             PORTA,04    ;SD pin


        bsf             PORTA,00    ;1.2ms timer
        bcf             PORTA,01
        call    time_1


;*****************************************2****************

        bcf             PORTA,00    ;0.6ms timer
        bsf             PORTA,01
        call    time_2

;************************************** 3 ***************

        bsf             PORTA,00    ;1.7ms timer
        bcf             PORTA,01
        call    time_3

;************************************** 4 ***************

        bcf             PORTA,00    ;0.2ms timer
        bsf             PORTA,01
        call    time_4
```

```
;************************************ 5 ***************

        bSf         PORTA,00    ;5.0ms timer
        bcf         PORTA,01
        call    time_5

;************************************ 6 **************

        bCf         PORTA,00    ;0.2ms timer
        bSf         PORTA,01
        call    time_6

;************************************ 7 ****************

        bSf         PORTA,00    ;1.7ms timer
        bcf         PORTA,01
        call    time_7

;************************************ 8 ***************

        bCf         PORTA,00    ;0.6ms timer
        bSf         PORTA,01
        call    time_8

;************************************ 9 *****************

        bSf         PORTA,00    ;1.2ms timer
        bcf         PORTA,01
        call    time_9

        bsf         PORTA,03

;*****************************************************
;       Timer Calculation

;************************************ 1 **************

        bsf         PORTA,02    ;

        bsf         PORTA,04    ;SD pin

        bcf         PORTA,00    ;1.2ms timer
        bsf         PORTA,01
        call    time_1
```

```
;*****************************************2**************

            bsf         PORTA,00    ;0.6ms timer
            bcf         PORTA,01
            call    time_2

;************************************** 3 ****************

            bcf         PORTA,00    ;1.7ms timer
            bsf         PORTA,01
            call    time_3

;************************************** 4 ****************

            bsf         PORTA,00    ;0.2ms timer
            bcf         PORTA,01
            call    time_4

;************************************** 5 ****************

            bcf         PORTA,00    ;5.0ms timer
            bsf         PORTA,01
            call    time_5

;************************************** 6 ***************

            bsf         PORTA,00    ;0.2ms timer
            bcf         PORTA,01
            call    time_6

;************************************** 7 *****************

            bcf         PORTA,00    ;1.7ms timer
            bsf         PORTA,01
            call    time_7

;************************************** 8 ****************

            bsf         PORTA,00    ;0.6ms timer
            bcf         PORTA,01
            call    time_8

;************************************** 9 ******************
```

```
                bcf             PORTA,00    ;1.2ms timer
                bsf             PORTA,01
                call    time_9

                bsf             PORTA,03    ;

                goto    @_1

;*************************************************************
;               timer sub program ; Speed at 40Hz
;********************************** 1 ********************

time_1

                bsf             T2CON,02

                btfsC           PORTB,04
                movlw d'12'                 ;1.2ms counter

                btfsc           PORTB,05
                movlw d'17'                 ;1.7ms counter

                btfsc           PORTB,06
                movlw d'21'                 ;2.1ms counter

                movwf 45

aaa             bsf             T2CON,02
                btfss   PIR1,01
                goto    aaa
                nop
                nop
                nop

                bcf             PIR1,01
                bcf             T2CON,02
                decfsz  45,f
                goto    aaa
                nop
                nop
                return

;*************************************************************


time_2
```

```
        bsf     T2CON,02

        btfsC   PORTB,04
        movlw d'6'              ;0.6ms counter

        btfsc   PORTB,05
        movlw d'9'              ;0.9ms counter

        btfsc   PORTB,06
        movlw d'13'             ;1.3ms counter

        movwf 45

bbb     bsf     T2CON,02
        btfss   PIR1,01
        goto    bbb
        nop
        nop
        nop

        bcf     PIR1,01
        bcf     T2CON,02
        decfsz  45,f
        goto    bbb
        nop
        nop
        return

;*********************************************************

time_3

        bsf     T2CON,02

        btfsC   PORTB,04
        movlw d'17'             ;1.7ms counter

        btfsc   PORTB,05
        movlw d'26'             ;2.6ms counter

        btfsc   PORTB,06
        movlw d'30'             ;3.0ms counter

        movwf 45
```

```
ccc         bsf         T2CON,02
            btfss   PIR1,01
            goto    ccc
            nop
            nop
            nop

            bcf         PIR1,01
            bcf         T2CON,02
            decfsz  45,f
            goto    ccc
            nop
            nop
            return


;*****************************************************

time_4

            bsf         T2CON,02

            btfsC       PORTB,04
            movlw d'2'                  ;0.2ms counter

            btfsc       PORTB,05
            movlw d'4'                  ;0.4ms counter

            btfsc       PORTB,06
            movlw d'7'                  ;0.7ms counter

            movwf 45

ddd         bsf         T2CON,02
            btfss   PIR1,01
            goto    ddd
            nop
            nop
            nop

            bcf         PIR1,01
            bcf         T2CON,02
            decfsz  45,f
            goto    ddd
            nop
            nop
```

```
                return

;*************************************************

time_5

                bsf             T2CON,02

                btfsC           PORTB,04
                movlw d'50'                     ;5ms counter

                btfsc           PORTB,05
                movlw d'36'                     ;3.6ms counter

                btfsc           PORTB,06
                movlw d'37'                     ;3.7ms counter

                movwf 45

eee             bsf             T2CON,02
                btfss   PIR1,01
                goto    eee
                nop
                nop
                nop

                bcf             PIR1,01
                bcf             T2CON,02
                decfsz  45,f
                goto    eee
                nop
                nop
                return

;*************************************************

time_6

                bsf             T2CON,02

                btfsC           PORTB,04
                movlw d'2'                      ;0.2ms counter

                btfsc           PORTB,05
                movlw d'4'                      ;0.4ms counter
```

```
                btfsc           PORTB,06
                movlw d'7'                      ;0.7ms counter

                movwf45

fff             bsf             T2CON,02
                btfss   PIR1,01
                goto    fff
                nop
                nop
                nop

                bcf             PIR1,01
                bcf             T2CON,02
                decfsz  45,f
                goto    fff
                nop
                nop
                return

;**************************************************

time_7

                bsf             T2CON,02

                btfsC           PORTB,04
                movlw d'17'                     ;1.7ms counter

                btfsc           PORTB,05
                movlw d'26'                     ;2.6ms counter

                btfsc           PORTB,06
                movlw d'30'                     ;3.0ms counter

                movwf45

ggg             bsf             T2CON,02
                btfss   PIR1,01
                goto    ggg
                nop
                nop
                nop

                bcf             PIR1,01
                bcf             T2CON,02
```

```
            decfsz  45,f
            goto    ggg
            nop
            nop
            return
;*************************************************

time_8

            bsf             T2CON,02

            btfsC           PORTB,04
            movlw d'6'                  ;0.6ms counter

            btfsc           PORTB,05
            movlw d'9'                  ;0.9ms counter

            btfsc           PORTB,06
            movlw d'13'                 ;1.3ms counter

            movwf 45

hhh         bsf             T2CON,02
            btfss   PIR1,01
            goto    hhh
            nop
            nop
            nop

            bcf             PIR1,01
            bcf             T2CON,02
            decfsz  45,f
            goto    hhh
            nop
            nop
            return

;*************************************************

time_9

            bsf             T2CON,02

            btfsC           PORTB,04    ;1.2ms counter
            movlw d'12'
```

```
        btfsc       PORTB,05
        movlw d'17'                 ;1.7ms counter

        btfsc       PORTB,06
        movlw d'21'                 ;2.1ms counter

        movwf 45

iii     bsf         T2CON,02
        btfss  PIR1,01
        goto   iii
        nop
        nop
        nop

        bcf         PIR1,01
        bcf         T2CON,02
        decfsz 45,f
        goto   iii
        nop
        nop
        return
END                             ; directive 'end of program'
```

Appendix B ;

Microprocessor program for the Remote Controller Receiver (PIC 16f628)

```
;*****************************************************************
;  Filename:      Receiiver.asm                                     *
;  Date:          15/6/2010                                         *
;  File Version:  6                                                 *
;                                                                   *
;  Author:        M.D.A.K.Wijerathna                                *
;  Project:       Development of a remote control hum less fan dimmer *
;                                                                   *
;*****************************************************************
;                                                                   *
;  Files required:                                                  *
;                                                                   *
;*****************************************************************
;                                                                   *
;  Notes:  The program is developed for 72%, 76%, 80% percentage line *
;          voltages.                                                *
;                                                                   *
;*****************************************************************

        list    p=16f628A        ; list directive to define processor
        #include <p16F628A.inc>      ; processor specific variable definitions

        errorlevel  -302         ; suppress message 302 from list file

;       __CONFIG  _CP_OFF & _DATA_CP_OFF & _LVP_OFF & _BOREN_OFF &
_MCLRE_OFF & _WDT_OFF & _PWRTE_ON & _INTOSC_OSC_NOCLKOUT

; '__CONFIG' directive is used to embed configuration word within .asm file.
```

```
; The lables following the directive are located in the respective .inc file.
; See data sheet for additional information on configuration word settings.

;*****************************************************************

        list    p=16f628A         ; list directive to define processor
        #include <p16F628A.inc>      ; processor specific variable definitions
        errorlevel  -302          ; suppress message 302 from list file

__CONFIG  _CP_ON & _DATA_CP_OFF & _LVP_OFF & _BOREN_OFF &
_MCLRE_ON & _WDT_OFF & _PWRTE_ON & _XT_OSC;_NOCLKOUT

; '__CONFIG' directive is used to embed configuration word within .asm file.
; The lables following the directive are located in the respective .inc file.
; See data sheet for additional information on configuration word settings.

;***** VARIABLE DEFINITIONS*****************************************

w_temp      EQU    0x7E      ; variable used for context saving
status_temp EQU    0x7F      ; variable used for context saving

time1_bit1          EQU         0x21
time1_bit2          EQU         0x22
time1_bit3          EQU         0x23
time2_bit1          EQU         0x24

;*****************************************************************
        ORG    0x000          ; processor reset vector
        goto   main           ; go to beginning of program
```

```
        ORG    0x004          ; interrupt vector location
        movwf  w_temp         ; save off current W register contents
        movf   STATUS,w       ;  move status register into W register
        movwf status_temp     ; save off contents of STATUS register

test_1  call   time_t
        btfss  PORTB,00
        goto   reset
        call   time_t
        btfss  PORTB,00
        goto   reset
        call   time_t
        btfss  PORTB,00
        goto   reset
        call   time_t

        call   time_t
        btssc  PORTB,00
        goto   reset
        call   time_t
        call   time_t
        btssc  PORTB,00
        goto   reset
        call   time_t

        call   time_t
        btfss  PORTB,00

reset
reset
```

```
; isr code can go here or be located as a call subroutine elsewhere

        movf   status_temp,w        ; retrieve copy of STATUS register
        movwf STATUS                ; restore pre-isr STATUS register contents
        swapf  w_temp,f
        swapf  w_temp,w             ; restore pre-isr W register contents
        retfie                      ; return from interrupt

main
; remaining code goes here

;       goto   main                 ;loop forever, remove this instruction, for test only

ini     clrf   STATUS
        bsf   STATUS,05             ;Change to Bank1
     movlw  b'00000000'             ;GIE=1,TOIE=1
     movwf   OPTION_REG

     bcf              STATUS,05     ;Change to Bank0

main_1bsf     PORTA,00              ;test indicator
        call   time_d               ;1s time
        bcf    PORTA,00
        call   time_d               ;1s time
        goto   main_1
time
        return

time_a call    time_t
        call   time_t
```

```
        call    time_t
        call    time_t
                return

time_b  call    time_t
        call    time_t
        return

time_c                          ;250mS
        return
time_t
        return

        END                     ; directive 'end of program'
```

Appendix C ;

Microprocessor program for the Remote Controller Emitter (PIC 16f628)

```
;*****************************************************************
;  Filename:      Emmiter.asm                                     *
;  Date:          20/5/2010                                       *
;  File Version:  6                                               *
;                                                                 *
;  Author:        M.D.A.K.Wijerathna                              *
;  Project:       Development of a remote control hum less fan dimmer *
;                                                                 *
;*****************************************************************
;  Files required:                                                *
;                                                                 *
;*****************************************************************
;                                                                 *
;  Notes:  The program is developed to emit the signals from      *
;          the remote  controller.                                *
;                                                                 *
;*****************************************************************

        list    p=16f628A        ; list directive to define processor
        #include <p16F628A.inc>      ; processor specific variable definitions

        errorlevel  -302         ; suppress message 302 from list file

        __CONFIG  _CP_ON & _DATA_CP_OFF & _LVP_OFF & _BOREN_OFF &
        _MCLRE_ON & _WDT_OFF & _PWRTE_ON & _XT_OSC;_NOCLKOUT

; '__CONFIG' directive is used to embed configuration word within .asm file.
; The lables following the directive are located in the respective .inc file.
```

```
; See data sheet for additional information on configuration word settings.

;***** VARIABLE DEFINITIONS************************************

w_temp      EQU   0x7E     ; variable used for context saving
status_temp EQU   0x7F     ; variable used for context saving

time1_bit1          EQU         0x21
time1_bit2          EQU         0x22
time1_bit3          EQU         0x23
time2_bit1          EQU         0x24

;*****************************************************************************
        ORG    0x000        ; processor reset vector
        goto   main         ; go to beginning of program

        ORG    0x004                ; interrupt vector location
        movwf  w_temp               ; save off current W register contents
        movf   STATUS,w             ; move status register into W register
        movwf status_temp           ; save off contents of STATUS register

; isr code can go here or be located as a call subroutine elsewhere

        movf   status_temp,w        ; retrieve copy of STATUS register
        movwf STATUS                ; restore pre-isr STATUS register contents
        swapf  w_temp,f
        swapf  w_temp,w             ; restore pre-isr W register contents
        retfie                      ; return from interrupt

main
```

```
; remaining code goes here

;       goto    main                ;loop forever, remove this instruction, for test only

ini     clrf    STATUS
        bsf     STATUS,05           ;Change to Bank1
      movlw   b'00000000'           ;GIE=1,TOIE=1
      movwf   OPTION_REG

     bcf              STATUS,05          ;Change to Bank0
main_1
        btfss   PORTB,05            ;on
        call    on
        btfss   PORTB,06            ;off
        call    off
        btfss   PORTB,07            ;speed
        call    speed
        call    time
        goto    main_1
on      movlw b'00000011'
        movwf 28
on_1    decfss 28
        goto    $1
        return
        bsf     PORTB,00            ;Hedar1
        call    time_a
        bcf     PORTB,00
        call    time_a
        call    time_b              ;Data
        call    time_b              ;Data
        bsf     PORTB,00            ;Data
```

```
        call    time_b
        bcf     PORTB,00
        call    time_a
        bsf     PORTB,00        ;End bit
        call    time_a
        bcf     PORTB,00
        call    time_c          ;250mS timer
        goto    on_1

off     movlw b'00000011'
        movwf 29
off_1   decfss  29
        goto    $1
        return
        bsf     PORTB,00        ;Hedar1
        call    time_a
        bcf     PORTB,00
        call    time_a
        call    time_b          ;Data
        bsf     PORTB,00        ;Data
        call    time_b          ;Data
        bcf     PORTB,00
        call    time_b          ;Data
        call    time_a
        bsf     PORTB,00        ;End bit
        call    time_a
        bcf     PORTB,00
        call    time_c          ;250mS timer
        goto    off_1

speed   movlw b'00000011'
```

```
        movwf 30
speed_1     decfss  30
        goto    $1
        return
        bsf     PORTB,00        ;Hedar1
        call    time_a
        bcf     PORTB,00
        call    time_a
        call    time_b          ;Data
        bsf     PORTB,00        ;Data
        call    time_b
        call    time_b          ;Data
        bcf     PORTB,00
        call    time_a
        bsf     PORTB,00        ;End bit
        call    time_a
        bcf     PORTB,00
        call    time_c          ;250mS timer
        goto    speed_1
time
        return
time_a
        time_t
        time_t
        time_t
        time_t
                return
time_b
        time_t
        time_t
```

```
        return
time_c                          ;250mS
        return

time_t
        loop_11     movlw b'01000111'
                    movwf time2_bit1

        loop_12     decfsz  time2_bit1
                    goto    loop_12
                    return

        END                         ; directive 'end of program
```