หอสมุดมหาวิทยาลัยราชภัฏธนบุรี

ภาคผนวก ก

คำสั่งโปรแกรมไมโครคอนโทรลเลอร์

```
CCS PCM C Compiler, Version 4.013, 28193               23-Sep-14 19:57

               Filename: D:\My Office\Sim_ISIS\16F877\MPPT\MPPT.lst

               ROM used: 291 words (4%)
                     Largest free fragment is 2048
               RAM used: 18 (5%) at main() level
                     19 (5%) worst case
               Stack:    2 locations

*
0000:  MOVLW  00
0001:  MOVWF  0A
0002:  GOTO   07C
0003:  NOP
.................... ////////////////////////////////////////////////////////////////////////////
.................... //////////////////////////          Project       MPPT@DRU.sp
////////////////////////////
.................... //////////////////////////    ID : 5349600013    ////////////////////////////
.................... ////////////////////////////////////////////////////////////////////////////
....................
.................... #include<16F877.h>
.................... //////// Standard Header file for the PIC16F877 device ////////////////
.................... #device PIC16F877
.................... #list
....................
.................... #device ADC=10      /// use 10-bit resolution for ADC of 16F877 //////
.................... #FUSES HS,NOPROTECT,NOPUT,NOLVP              /// Function setting of
Controller
.................... #use delay(clock=20000000)              /// Frequency Cystal Value 20MHz
////
0066:  MOVLW  28
0067:  MOVWF  04
0068:  BCF    03.7
0069:  MOVF   00,W
006A:  BTFSC  03.2
```

```
006B:  GOTO   079
006C:  MOVLW  06
006D:  MOVWF  78
006E:  CLRF   77
006F:  DECFSZ 77,F
0070:  GOTO   06F
0071:  DECFSZ 78,F
0072:  GOTO   06E
0073:  MOVLW  7B
0074:  MOVWF  77
0075:  DECFSZ 77,F
0076:  GOTO   075
0077:  DECFSZ 00,F
0078:  GOTO   06C
0079:  BCF    0A.3
007A:  BCF    0A.4
007B:  GOTO   121 (RETURN)
..... #use rs232(baud=115200, parity=N, xmit=PIN_C6, rcv=PIN_C7, bits=8)
..... #define pi=3.1414  //// pi=360/2 ////
..... ////////////////////////////////////////////////////////////////////////////
.....
..... ////////////////////////////////////Timer 2////////////////////////////////////
..... // Timer 2 Functions: SETUP_TIMER_2, GET_TIMER2, SET_TIMER2//////////////////
..... // Constants used for SETUP_TIMER_2() are:////////////////////////////////////
..... #define ADC_CLOCK_INTERNAL 0x00c0          // Internal 2-6us
..... // Constants used in READ_ADC() are:
..... #define ADC_START_AND_READ    7   // This is the default if nothing is specified
..... #define ADC_START_ONLY        1
..... #define ADC_READ_ONLY         6
..... ////////////////////////////////////////////////////////////////////////////
.....
..... ////////////////////////////////////////////////////////////////// CCP/////////
```

..... ////////////////// CCP Functions: SETUP_CCPx, SET_PWMx_DUTY//////////////////

..... //////////////////CCP Variables: CCP_x, CCP_x_LOW, CCP_x_HIGH///////////////

..... ////////////////// Constants used for SETUP_CCPx() are:////////////////////

..... #define CCP_OFF 0

..... #define CCP_CAPTURE_FE 4

..... #define CCP_CAPTURE_RE 5

..... #define CCP_COMPARE_CLR_ON_MATCH 9

..... #define CCP_COMPARE_INT 0xA

..... #define CCP_COMPARE_RESET_TIMER 0xB

..... #define CCP_PWM 0xC

..... #define CCP_PWM_PLUS_1 0x1c

..... #define CCP_PWM_PLUS_2 0x2c

..... #define CCP_PWM_PLUS_3 0x3c

..... long CCP;

..... #byte CCP_1 = 0x15

..... #byte CCP_1_LOW= 0x15

..... #byte CCP_1_HIGH= 0x16

..... //

.....

..... // INT ///

..... // Interrupt Functions: ENABLE_INTERRUPTS(), DISABLE_INTERRUPTS(), ////

..... // EXT_INT_EDGE() ////

..... // ////

..... // Constants used in EXT_INT_EDGE() are: ////

..... //

..... #define GLOBAL 0x0BC0

..... #define INT_RTCC 0x0B20

..... #define INT_RB 0xFF0B08

..... #define INT_EXT 0x0B10

..... #define INT_AD 0x8C40

..... #define INT_TBE 0x8C10

..... #define INT_RDA 0x8C20

```
..... #define INT_TIMER1              0x8C01
..... #define INT_TIMER2              0x8C02
..... #define INT_CCP1               0x8C04
..... #define INT_CCP2               0x8D01
..... #define INT_SSP               0x8C08
..... #define INT_PSP               0x8C80
..... #define INT_BUSCOL             0x8D08
..... #define INT_EEPROM             0x8D10
..... #define INT_TIMER0             0x0B20
..... #define INT_COMP               0x8D40
.....
..... //////////////////////////// SPI//////////////////////////////////////
..... ///////SPI Functions: SETUP_SPI, SPI_WRITE, SPI_READ, SPI_DATA_IN//////
..... ///////////////// Constants used in SETUP_SPI() are://////////////////////
..... ////////////////////////////////////////////////////////////////////////
..... #define SPI_MASTER      0x20
..... #define SPI_SLAVE       0x24
..... #define SPI_L_TO_H      0
..... #define SPI_H_TO_L      0x10
..... #define SPI_CLK_DIV_4   0
..... #define SPI_CLK_DIV_16  1
..... #define SPI_CLK_DIV_64  2
..... #define SPI_CLK_T2      3
..... #define SPI_SS_DISABLED 1
.....
..... #define SPI_SAMPLE_AT_END 0x8000
..... #define SPI_XMIT_L_TO_H  0x4000
..... ////////////////////////////////////////////////////////////////////////////
.....
..... ////////////////////////////// UART//////////////////////////////////////
..... // Constants used in setup_uart()
are:////////////////////////////////////////
..... // FALSE - Turn UART off                          ////////////
..... // TRUE  - Turn UART on                           ////////////
..... #define UART_ADDRESS        2                     ////////////
..... #define UART_DATA           4                     ////////////
```

```
..... ////////////////////////////////////////////////////////////////////////////
..... ////////////////////////////////////////////////////////////////////////////
.....
..... void main (void)
..... {
007C: CLRF  04
007D: BCF   03.7
007E: MOVLW 1F
007F: ANDWF 03,F
0080: MOVLW 0A
0081: BSF   03.5
0082: MOVWF 19
0083: MOVLW A6
0084: MOVWF 18
0085: MOVLW 90
0086: BCF   03.5
0087: MOVWF 18
0088: BSF   03.5
0089: BSF   1F.0
008A: BSF   1F.1
008B: BSF   1F.2
008C: BCF   1F.3
..... unsigned int16 adc_val;
..... unsigned int8 period,Vsen,Isen;
.....
..... ////////////////////////////////////////////////////////////////////////////
.....
..... ////////////////////////////////////////////////////////////////////////////
..... ////////// calculate the PWM frequency at (20MHz/4)/4=1250000Hz
//////////////
..... ////////////////////////////////////////////////////////////////////////////
.....
..... period = 50;          //// frequency value (1250000/25kHz)=50 unit //////
*
0090: MOVLW 32
0091: MOVWF 25
```

```
.....
.....        ////////////////////////////////////////////////////////////
..... ////////////////////////// set timer at CCP1 by prescal 1:4 //////////////////
.....        ////////////////////////////////////////////////////////////
..... setup_timer_2(T2_DIV_BY_4,period-1,1);
0092:  MOVLW  01
0093:  SUBWF  25,W
0094:  MOVWF  28
0095:  MOVLW  00
0096:  MOVWF  78
0097:  IORLW  05
0098:  MOVWF  12
0099:  MOVF   28,W
009A:  BSF    03.5
009B:  MOVWF  12
..... setup_ccp1(CCP_PWM);              /// use CCP1 or RC0 is PWM mode
switching MPPT///
*
008D:  MOVLW  FF
008E:  BCF    03.5
008F:  MOVWF  22
*
009C:  BCF    03.5
009D:  BCF    22.2
009E:  MOVF   22,W
009F:  BSF    03.5
00A0:  MOVWF  07
00A1:  BCF    03.5
00A2:  BCF    07.2
00A3:  MOVLW  0C
00A4:  MOVWF  17
..... setup_ccp2(CCP_PWM);              /// use CCP2 or RC1 is PWM mode
switching 25kHz///
00A5:  BCF    22.1
00A6:  MOVF   22,W
00A7:  BSF    03.5
```

```
00A8:  MOVWF  07
00A9:  BCF    03.5
00AA:  BCF    07.1
00AB:  MOVLW  0C
00AC:  MOVWF  1D
.....
.....        ////////////////////////////////////////////////////////////
..... ////////////////////////// Analog input at AN0,AN1 =5Vdc /////////////////
.....        ////////////////////////////////////////////////////////////
..... Vsen=0;
00AD:  CLRF   26
..... Isen=0;
00AE:  CLRF   27
.....
..... set_pwm1_duty(Vsen);     //// Intitial PWM at Vsen will Varies Analog Input
////
00AF:  MOVF   26,W
00B0:  MOVWF  15
..... set_pwm2_duty(Isen);
00B1:  MOVF   27,W
00B2:  MOVWF  1B
.....
..... setup_adc_ports(ALL_ANALOG);              /// Analog signal Input AN0 ///
00B3:  BSF    03.5
00B4:  BCF    1F.0
00B5:  BCF    1F.1
00B6:  BCF    1F.2
00B7:  BCF    1F.3
..... setup_adc(ADC_CLOCK_INTERNAL);
00B8:  BCF    03.5
00B9:  BSF    1F.6
00BA:  BSF    1F.7
00BB:  BSF    03.5
00BC:  BSF    1F.7
00BD:  BCF    03.5
00BE:  BSF    1F.0
```

```
..... setup_adc(ADC_CLOCK_DIV_32);
00BF:  BCF    1F.6
00C0:  BSF    1F.7
00C1:  BSF    03.5
00C2:  BSF    1F.7
00C3:  BCF    03.5
00C4:  BSF    1F.0
.....
.....        //////////////////////////////////////////////////////////////
..... ////////////////////////// Intitial Varie duty cycle at 0% //////////////////////
.....        //////////////////////////////////////////////////////////////
..... while(1)
..... {
..... set_adc_channel(0);              //// read AN0 is Vsen ////
00C5:  MOVLW  00
00C6:  MOVWF  78
00C7:  MOVF   1F,W
00C8:  ANDLW  C7
00C9:  IORWF  78,W
00CA:  MOVWF  1F
..... adc_val=read_adc();
00CB:  BSF    1F.2
00CC:  BTFSC  1F.2
00CD:  GOTO   0CC
00CE:  BSF    03.5
00CF:  MOVF   1E,W
00D0:  BCF    03.5
00D1:  MOVWF  23
00D2:  MOVF   1E,W
00D3:  MOVWF  24
..... Vsen=((adc_val*period)-180)>>8;
00D4:  MOVF   24,W
00D5:  MOVWF  29
00D6:  MOVF   23,W
00D7:  MOVWF  28
00D8:  CLRF   2B
```

```
00D9:  MOVF   25,W
00DA:  MOVWF  2A
00DB:  CALL   004
00DC:  MOVLW  B4
00DD:  SUBWF  78,W
00DE:  MOVWF  2A
00DF:  MOVF   79,W
00E0:  MOVWF  2B
00E1:  MOVLW  00
00E2:  BTFSS  03.0
00E3:  MOVLW  01
00E4:  SUBWF  2B,F
00E5:  MOVF   2B,W
00E6:  MOVWF  26
.....
..... set_adc_channel(1);                  //// read A1 is Isen ////
00E7:  MOVLW  08
00E8:  MOVWF  78
00E9:  MOVF   1F,W
00EA:  ANDLW  C7
00EB:  IORWF  78,W
00EC:  MOVWF  1F
..... adc_val=read_adc();
00ED:  BSF    1F.2
00EE:  BTFSC  1F.2
00EF:  GOTO   0EE
00F0:  BSF    03.5
00F1:  MOVF   1E,W
00F2:  BCF    03.5
00F3:  MOVWF  23
00F4:  MOVF   1E,W
00F5:  MOVWF  24
..... Isen=((adc_val*period))>>8;
00F6:  MOVF   24,W
00F7:  MOVWF  29
00F8:  MOVF   23,W
```

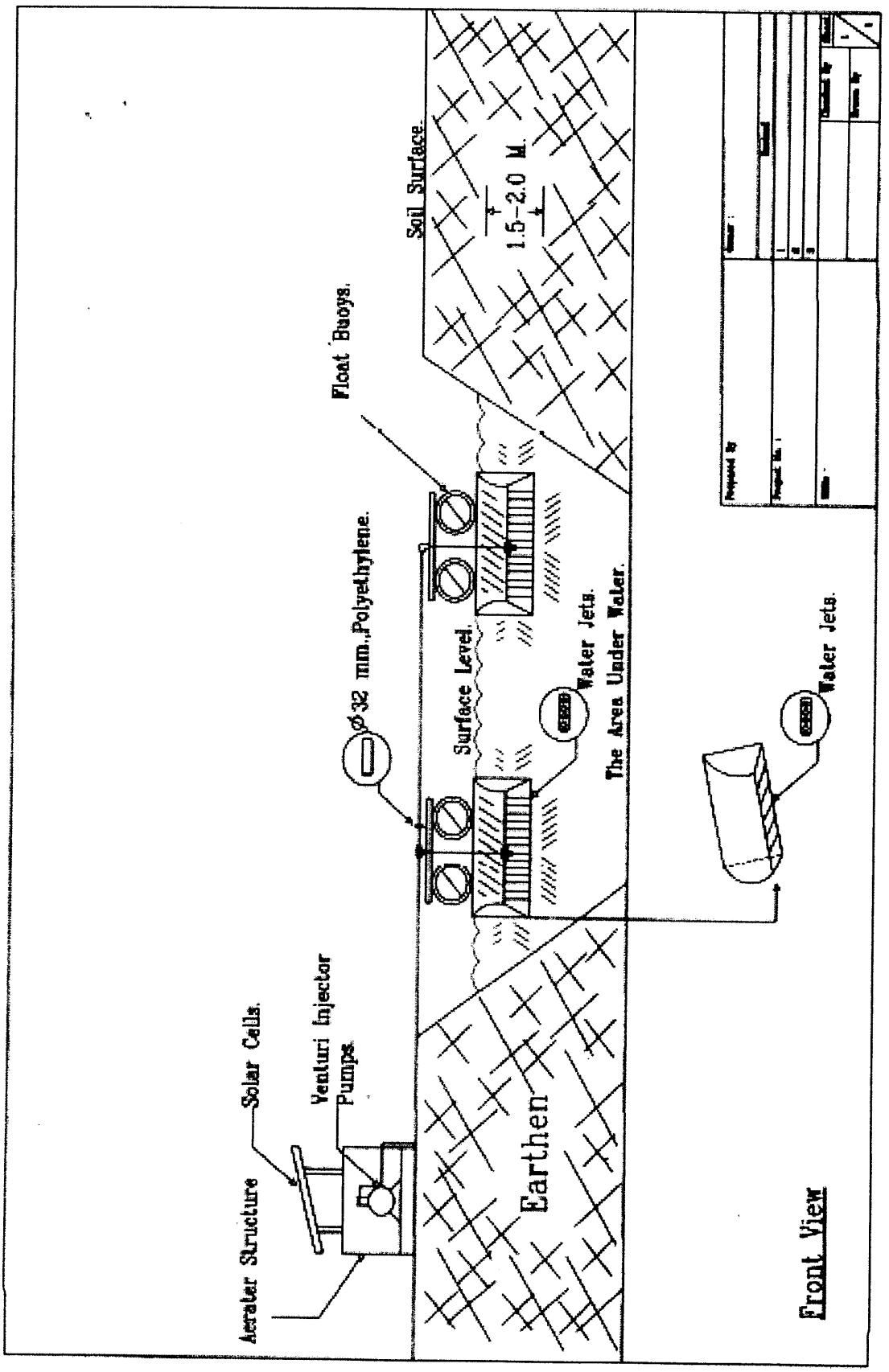
```
00F9:  MOVWF  28
00FA:  CLRF   2B
00FB:  MOVF   25,W
00FC:  MOVWF  2A
00FD:  CALL   004
00FE:  MOVF   79,W
00FF:  MOVWF  27
.....
..... set_pwm1_duty(Vsen);              ///// update duty of pwm1 respective at
ccp1 /////
0100:  MOVF   26,W
0101:  MOVWF  15
..... set_pwm2_duty(Isen);
0102:  MOVF   27,W
0103:  MOVWF  1B
.....
..... printf("%u %u \r\n",Vsen,Isen);    //// display the duty of pwm1 respective
////
0104:  MOVF   26,W
0105:  MOVWF  28
0106:  MOVLW  1B
0107:  MOVWF  29
0108:  CALL   02E
0109:  MOVLW  20
010A:  BTFSS  0C.4
010B:  GOTO   10A
010C:  MOVWF  19
010D:  MOVF   27,W
010E:  MOVWF  28
010F:  MOVLW  1B
0110:  MOVWF  29
0111:  CALL   02E
0112:  MOVLW  20
0113:  BTFSS  0C.4
0114:  GOTO   113
0115:  MOVWF  19
```

```
0116:  MOVLW  0D
0117:  BTFSS  0C.4
0118:  GOTO   117
0119:  MOVWF  19
011A:  MOVLW  0A
011B:  BTFSS  0C.4
011C:  GOTO   11B
011D:  MOVWF  19
..................... delay_ms(100);
011E:  MOVLW  64
011F:  MOVWF  28
0120:  GOTO   066
.....................
.....................
..................... }
0121:  GOTO   0C5
..................... }
.....................  ////////////////////////////////////////////////////////////////////////
.....................        ////////////////////////////////  end  process  MPPT
////////////////////////////
.....................      ////////////////////////////////////////////////////////////////////
.....................
.....................
0122:  SLEEP

Configuration Fuses:
  Word  1:  3F7A    HS NOWDT NOPUT NOPROTECT BROWNOUT NOLVP NOCPD
NOWRT NODEBUG
```

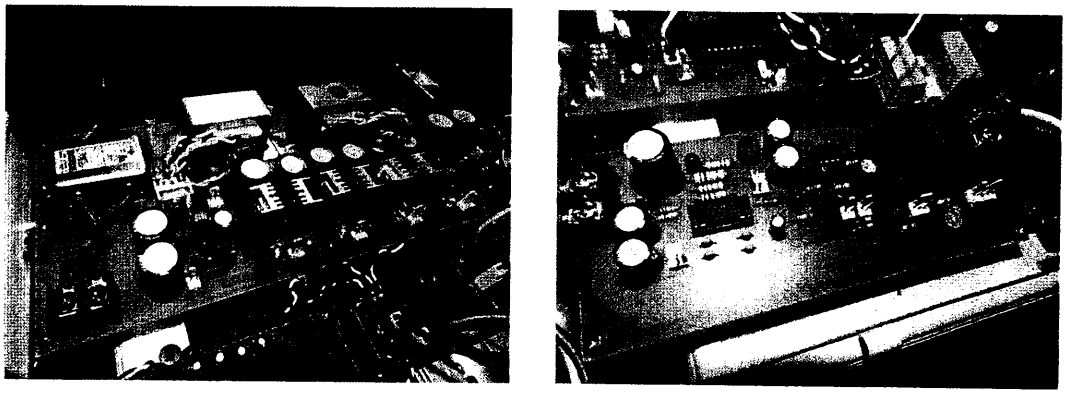
ภาคผนวก ข

แบบลักษณะโครงสร้างและงานติดตั้งในบ่อเพาะเลี้ยงกุ้ง

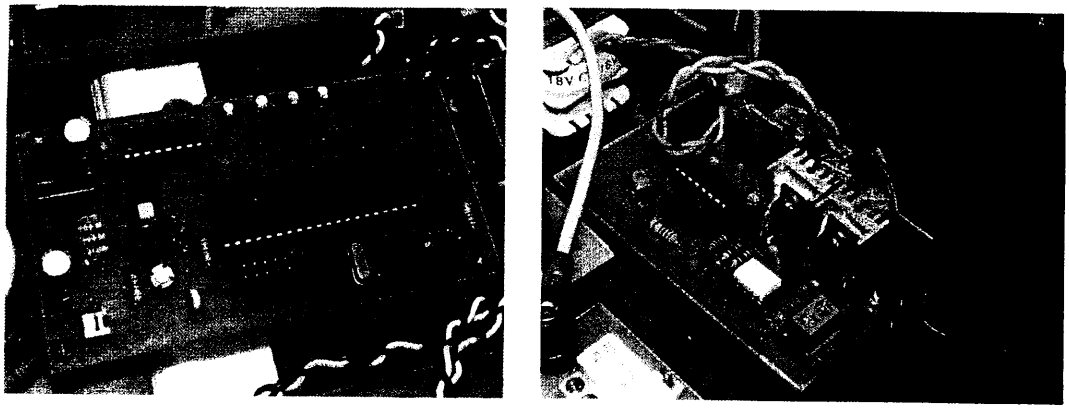

Soil Surface.
1.5–2.0 M.
Float Buoys.
Ø 32 mm. Polyethylene.
Surface Level.
Water Jets.
The Area Under Water.
Water Jets.
Solar Cells.
Venturi Injector
Pumps.
Aerator Structure
Earthen
Front View

ภาคผนวก ค

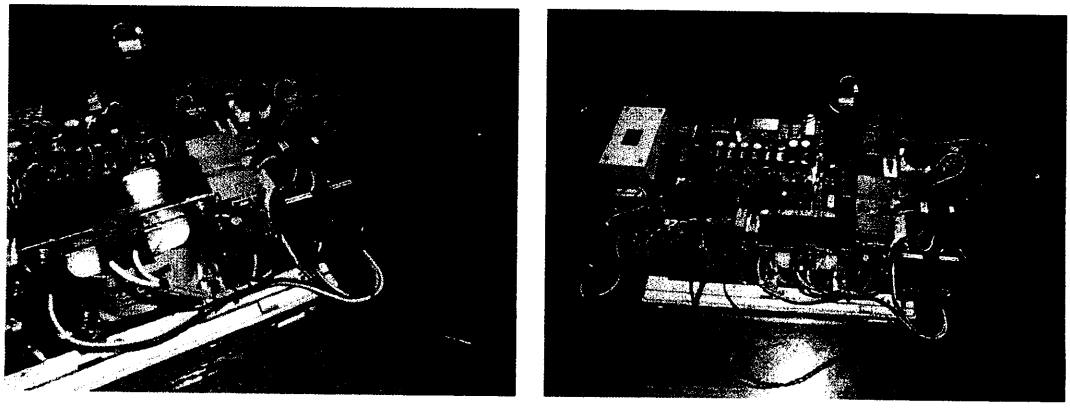
ภาพส่วนประกอบชิ้นงาน



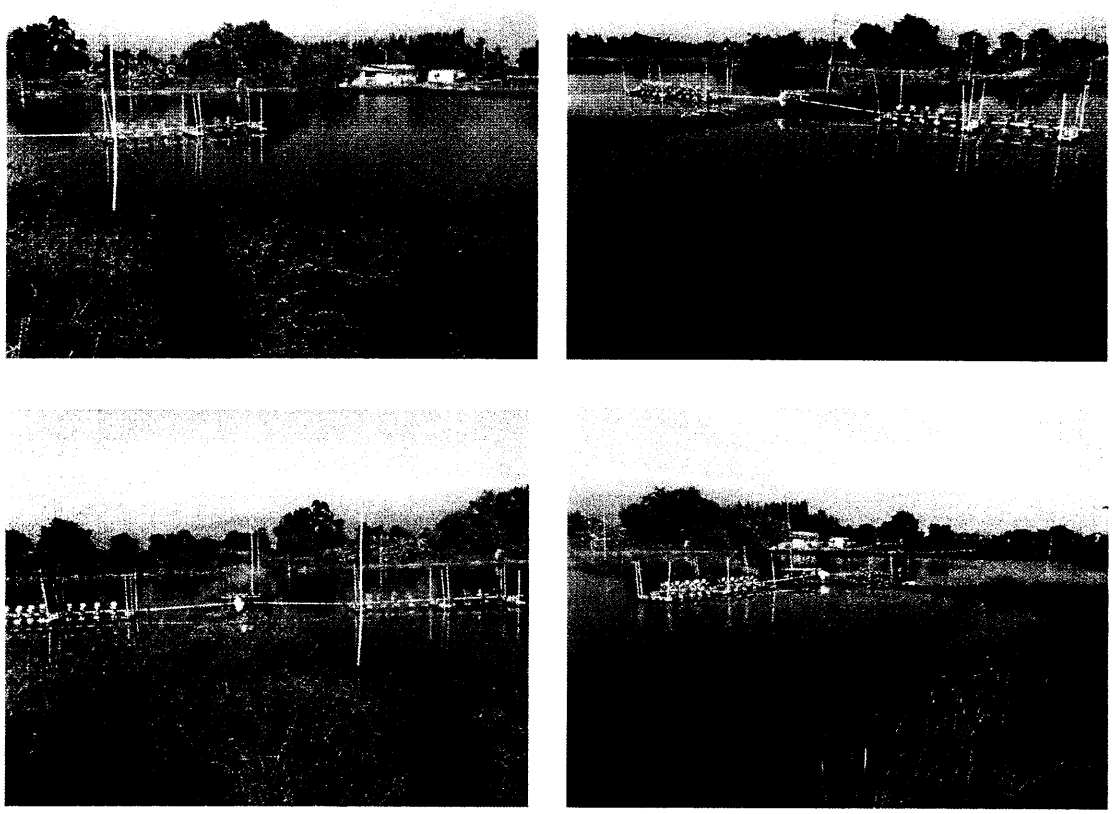
ภาพ วงจรไฟเลี้ยงไอซีในวงจรควบคุม และวงจรตรวจจับสัญญาณแรงดันและกระแส



ภาพ วงจรประมวลผลไมโครคอนโทรลเลอร์PIC16F877 และ วงจรไอซีขับเกต TLP250

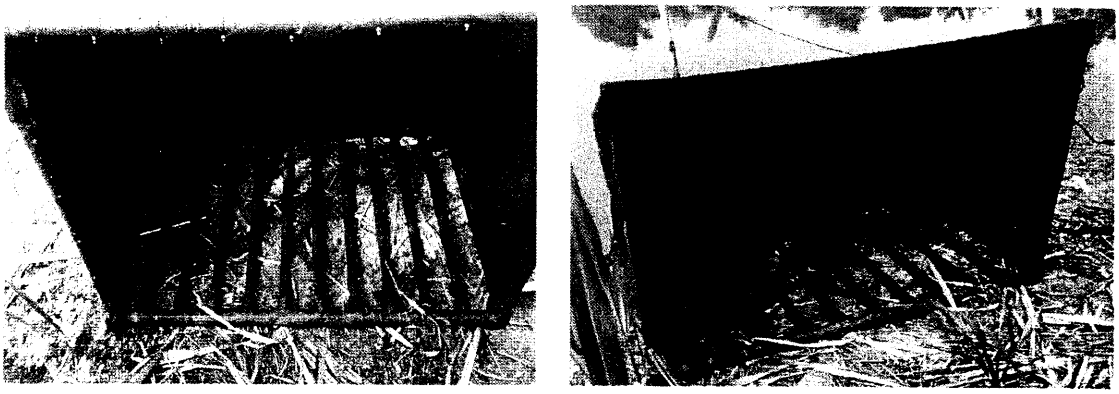


ภาพ วงจรแปรผันแรงดันไฟฟ้ากระแสตรงเป็นแรงดันไฟฟ้ากระแสตรง

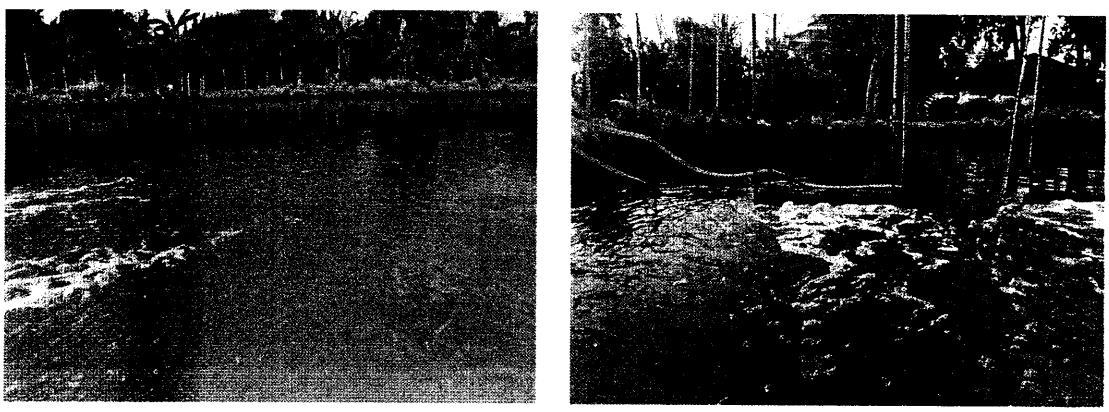


ภาพ การติดตั้งกังหันตีน้ำเพื่อเติมอากาศของเกษตรกรที่เพาะเลี้ยงกุ้งแบบเดิม

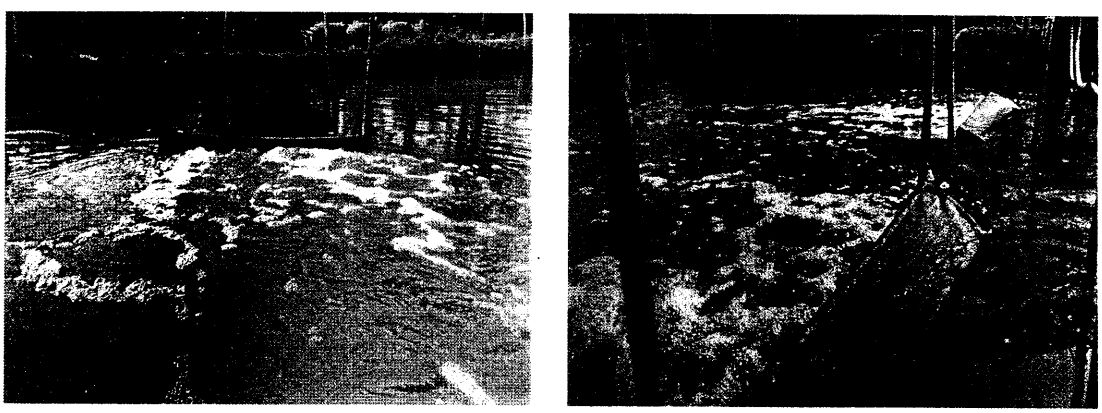
ภาพ การติดตั้งแผงเซลล์แสงอาทิตย์ร่วมกับเครื่องเติมอากาศบริเวณบ่อของเกษตรกรที่เพาะเลี้ยงกุ้ง



ภาพ การติดตั้ง Water Jets เพื่อต่อเชื่อมกับเครื่องเติมอากาศลงในบ่อกุ้งของเกษตรกรที่เพาะเลี้ยง



ภาพ การทำงานของ Water Jets เมื่อเชื่อมต่อกับเครื่องเติมอากาศในบ่อกุ้งของเกษตรกรที่เพาะเลี้ยง



ภาพการทำงานของ Water Jets เมื่อเชื่อมต่อกับเครื่องเติมอากาศในบ่อเพาะเลี้ยงกุ้ง

ภาคผนวก ง
แบบสอบถามความพึงพอใจ

แบบสอบถาม

เรื่อง ความพึงพอใจที่มีต่อการพัฒนาระบบเติมอากาศสำหรับบ่อกุ้งโดยใช้พลังงานแสงอาทิตย์

ส่วนที่ 1 แบบสอบถามเกี่ยวกับข้อมูลส่วนบุคคล
ส่วนที่ 2 แบบสอบถามเกี่ยวกับความพึงพอใจต่อการนำพลังงานแสงอาทิตย์มาใช้สำหรับเครื่องเติมอากาศเพื่อการหมุนเวียนน้ำในบ่อกุ้งของเกษตรกร
ส่วนที่ 3 แบบสอบถามเกี่ยวกับความพึงพอใจต่อการนำพลังงานแสงอาทิตย์มาใช้สำหรับเครื่องตีน้ำเพื่อการหมุนเวียนน้ำในบ่อกุ้งของเกษตรกร
ส่วนที่ 4 ข้อเสนอแนะเพิ่มเติม

ส่วนที่ 1 ข้อมูลทั่วไปของผู้ตอบแบบสอบถาม
โปรดทำเครื่องหมายลงใน ☑ หน้าข้อความที่ตรงกับตัวท่านมากที่สุด

1. เพศ
☐ ชาย ☐ หญิง
2. อายุ
☐ 17-24 ปี ☐ 25-34 ปี ☐ 35 ปีขึ้นไป
3. สถานภาพ
☐ โสด ☐ สมรส
4. การศึกษาสูงสุด
☐ ประถมศึกษา ☐ มัธยมศึกษา ☐ อนุปริญญา
☐ ปริญญาตรี ☐ ปริญญาโท ☐ ปริญญาเอก
5. ลักษณะพื้นที่ของบ่อเลี้ยง
☐ พื้นที่ของตนเอง ☐ พื้นที่เช่า ☐ พื้นที่ของตนเองและเช่าพื้นที่บางส่วน
6. จำนวนผู้ดูแลบ่อเลี้ยง
☐ 2 คน ☐ 3 คน ☐ 4 คน
☐ มากกว่า 5 ขึ้นไป
7. รายได้ต่อเดือน
☐ ต่ำกว่า 5,000 บาท ☐ 5,001 ถึง 10,000 บาท
☐ 10,001 ถึง 20,000 บาท ☐ 20,000 บาทขึ้นไป

ตอนที่ 2 ความพึงพอใจต่อการนำพลังงานแสงอาทิตย์มาใช้สำหรับเครื่องเติมอากาศเพื่อการหมุนเวียนน้ำในบ่อกุ้งของเกษตรกร

คำชี้แจง โปรดทำเครื่องหมาย ✓ ในช่องว่างหลังข้อความแต่ละช่อง ที่ท่านมีความพึงพอใจต่อการนำพลังงานแสงอาทิตย์มาใช้สำหรับเครื่องเติมอากาศเพื่อการหมุนเวียนน้ำในบ่อกุ้งของเกษตรกร

เกณฑ์ระดับความพึงพอใจ

5 หมายถึง ระดับความพึงพอใจมากที่สุด
4 หมายถึง ระดับความพึงพอใจมาก
3 หมายถึง ระดับความพึงพอใจปลานกลาง
2 หมายถึง ระดับความพึงพอใจน้อย
1 หมายถึง ระดับความพึงพอใจน้อยที่สุด

ความพึงพอใจ	ระดับความพึงพอใจ				
	5	4	3	2	1
ประสิทธิภาพและสมรรถนะ					
1. คุณภาพความคงทนของอุปกรณ์					
2. ความเหมาะสมของรูปแบบการติดตั้ง					
3. การได้รับพลังงานไฟฟ้าจากเซลล์แสงอาทิตย์					
4. ขนาดของแผงเซลล์แสงอาทิตย์เหมาะสม					
5. การประจุพลังงานที่ได้รับ					
6. ความสะดวกในการติดตั้งใช้งาน					
7. ความปลอดภัยในการใช้งาน					
8. ลดขั้นตอนและระยะเวลาการติดตั้ง					
9. ความซับซ้อนในการติดตั้งใช้งาน					
10. ราคามีความเหมาะสม					
คุณภาพและความปลอดภัย					
11. ปริมาณออกซิเจนในน้ำ					
12. สภาพภาพน้ำในบ่อ					
13. ระยะเวลาในการปรับสภาพภาพน้ำ					
14. ความเชื่อมั่นของอุปกรณ์ที่ติดตั้ง					
15. ความมั่นใจด้านความปลอดภัยของสัตว์น้ำ					

ตอนที่ 3 ความพึงพอใจที่มีต่อนำพลังงานแสงอาทิตย์มาใช้สำหรับเครื่องตีน้ำเพื่อการหมุนเวียนน้ำในบ่อกุ้งของเกษตรกร

คำชี้แจง โปรดทำเครื่องหมาย ✓ ในช่องว่างหลังข้อความแต่ละช่อง ที่ท่านมีความพึงพอใจต่อการนำพลังงานแสงอาทิตย์มาใช้สำหรับเครื่องเติมอากาศเพื่อการหมุนเวียนน้ำในบ่อกุ้งของเกษตรกร

เกณฑ์ระดับความพึงพอใจ

5	หมายถึง	ระดับความพึงพอใจมากที่สุด
4	หมายถึง	ระดับความพึงพอใจมาก
3	หมายถึง	ระดับความพึงพอใจปลานกลาง
2	หมายถึง	ระดับความพึงพอใจน้อย
1	หมายถึง	ระดับความพึงพอใจน้อยที่สุด

ความพึงพอใจ	ระดับความพึงพอใจ				
	5	4	3	2	1
ประสิทธิภาพและสมรรถนะ					
1. คุณภาพความคงทนของอุปกรณ์					
2. ความเหมาะสมของรูปแบบการติดตั้ง					
3. การได้รับพลังงานไฟฟ้าจากเซลล์แสงอาทิตย์					
4. ขนาดของแผงเซลล์แสงอาทิตย์เหมาะสม					
5. การประจุพลังงานที่ได้รับ					
6. ความสะดวกในการติดตั้งใช้งาน					
7. ความปลอดภัยในการใช้งาน					
8. ลดขั้นตอนและระยะเวลาการติดตั้ง					
9. ความซับซ้อนในการติดตั้งใช้งาน					
10. ราคามีความเหมาะสม					
คุณภาพและความปลอดภัย					
11. ปริมาณออกซิเจนในน้ำ					
12. สภาพภาพน้ำในบ่อ					
13. ระยะเวลาในการปรับสภาพภาพน้ำ					
14. ความเชื่อมั่นของอุปกรณ์ที่ติดตั้ง					
15. ความมั่นใจด้านความปลอดภัยของสัตว์น้ำ					

ตอนที่ 4 ข้อเสนอแนะเพิ่มเติม

คำชี้แจง โปรดให้ข้อเสนอแนะเพิ่มเติมเกี่ยวกับแนวทางการพัฒนาระบบเติมอากาศสำหรับบ่อกุ้งโดยใช้พลังงานแสงอาทิตย์ ที่ท่านคิดว่าน่าจะเป็นประโยชน์ต่อการพัฒนาต่อไป

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....