Growinator

Hydroponic Control System

Instruction Manaual

Table of contents:

Thank you for purchasing the Growinator hydroponic control system.

Please read this instruction manual for safety and proper use of this system.

Retain this manual for future reference.

Safety and Precautions

WARNING:

This unit is connected directly to the AC line of your home. Please insure that wiring is done properly to avoid the risk of electric shock or death.

Only use this unit is the manner it was designed for use in. Failing to use this unit properly or modifying the unit increases your risk of electric shock or death.

The control system is not waterproof. Insure that the unit is positioned in a place where it does not come in direct contact with water.

Do not disassemble the unit.

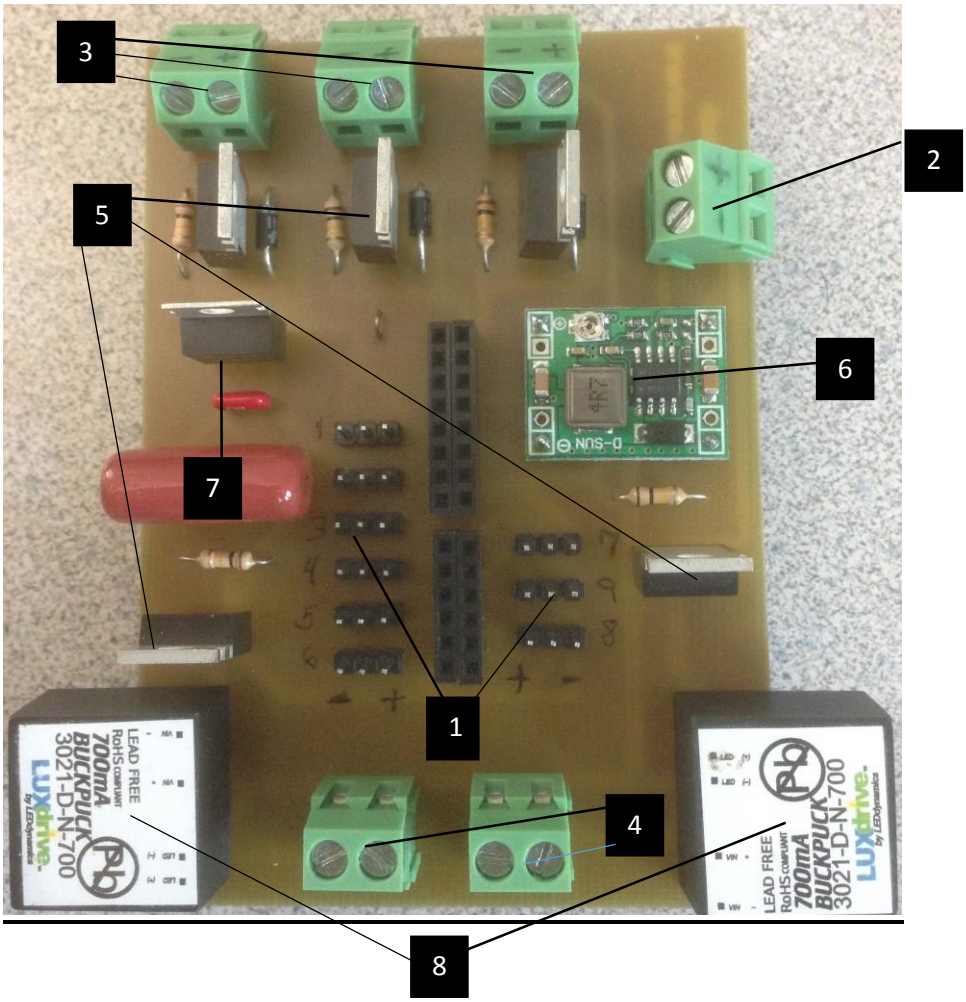
Do not store the unit in direct sunlight or high temperature areas.

Do not place tongue onto the unit.

Features

- Easy to use system that controls your hydroponic growing system for you
- Preset light timing schedule to eliminate the need for mechanical timers
- Moisture sensors which decide when it is time to water your plants for you
- Integrated water control
- 18/6 Hours on/off for lighting control
- All in one, plug and play unit
- Low power unit compared to always on systems

Components

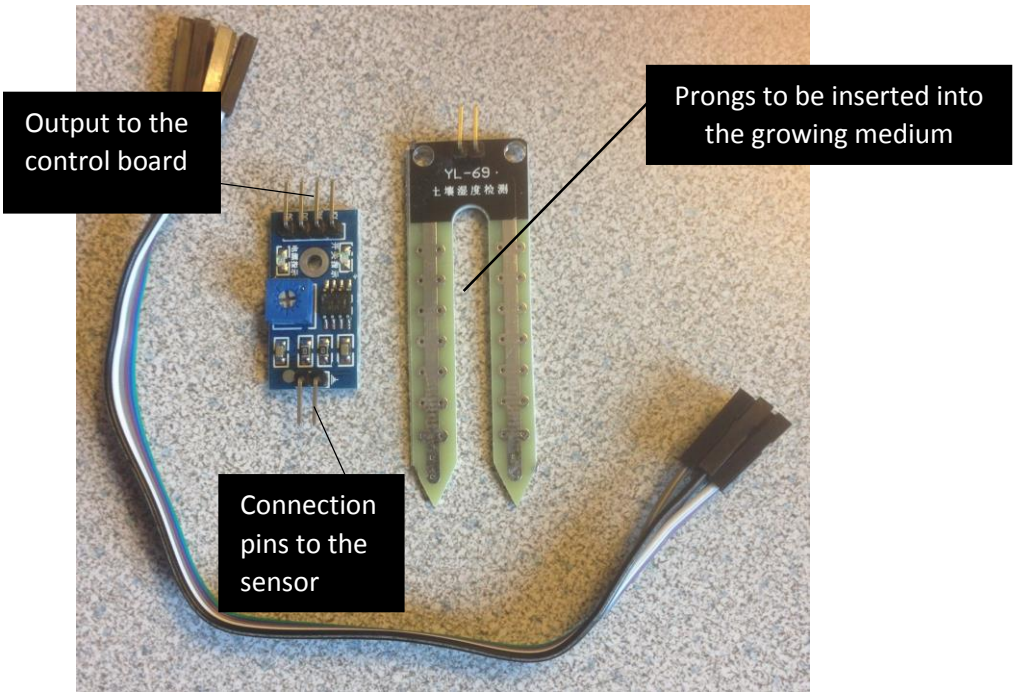


1. Header pins for moisture sensors
2. Screw terminal for input power
3. Screw terminals for power out to 12V water pumps
4. Screw terminals for 700 mA constant current out to LEDs
5. MOSFET switches
6. 12V DC-DC step down converter
7. LM7805 5V DC-DC step down converter

8. 700 mA constant current converters

How to use this unit

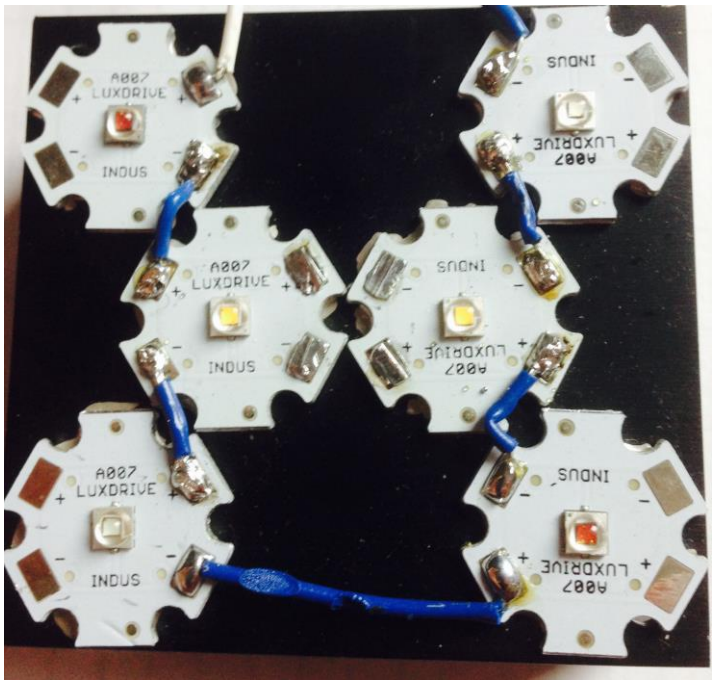
1. First position your moisture sensors with the prongs in the growing medium that you are using.



2. Connect the output pins of the sensor to the input of the sensor board. Then connect the analog out, VCC and ground pins from the sensor board to your choice of sensor inputs on the control board (choice of nine inputs shown as #1 on the component picture)
3. Connect your 12V water pump to the + and – (red for + and black for -) screw terminals of the control board to power them (shown as #3 on the component picture). Make sure your water pumps are 12V DC pumps. These can also be purchased through us if you are unsure.

12V DC water pump that we recommend

4. Connect your LED lighting to the + and - screw terminals on the control board to power them (shown as #4 on the component picture). Make sure your LED arrays do NOT exceed 24V forward voltage. These arrays can be purchased through us if you are unsure. For our arrays white is + and blue is -.



17V LED array

5. Locate the power supply you plan to use to power the unit. We recommend a 24V 5A supply as shown here.

6. Connect the V+ and the COM to the + and – input terminals on the board (Shown as #2 on the component picture).
7. Make sure that your input cord from the AC line of your home is connected to the proper terminals of the power supply. L (line), N (Neutral), and Ground.
8. Plug in the power supply and your system is now running!

Specifications

Max input:

24V DC maximum input voltage

Max outputs:

Water pump output: 12V DC maximum
LED output: 24V DC maximum @ 700 mA

Sensor output: 5V DC maximum

MOSFET switch characteristics:

Gate threshold: 1.2V minimum to 2.5V maximum

Max drain to source voltage: 60V maximum

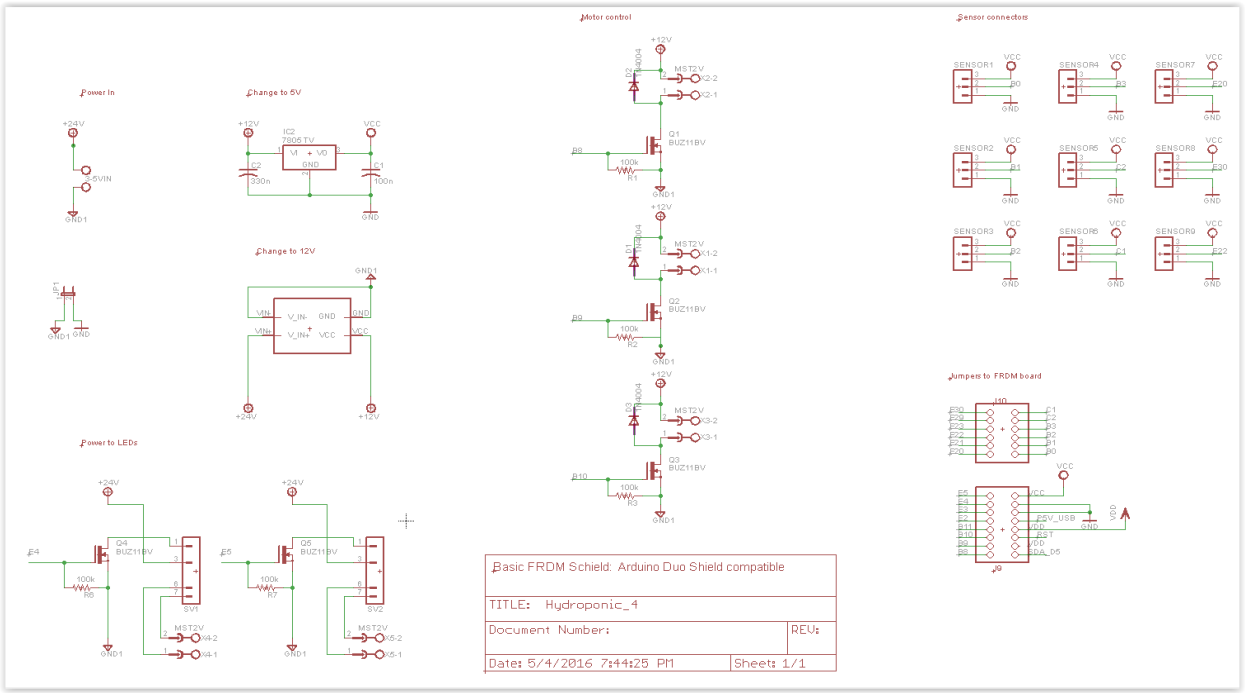
Qunqi DC-DC converter:

Adjustable from 3 to 12V. Preset to 12V in this system

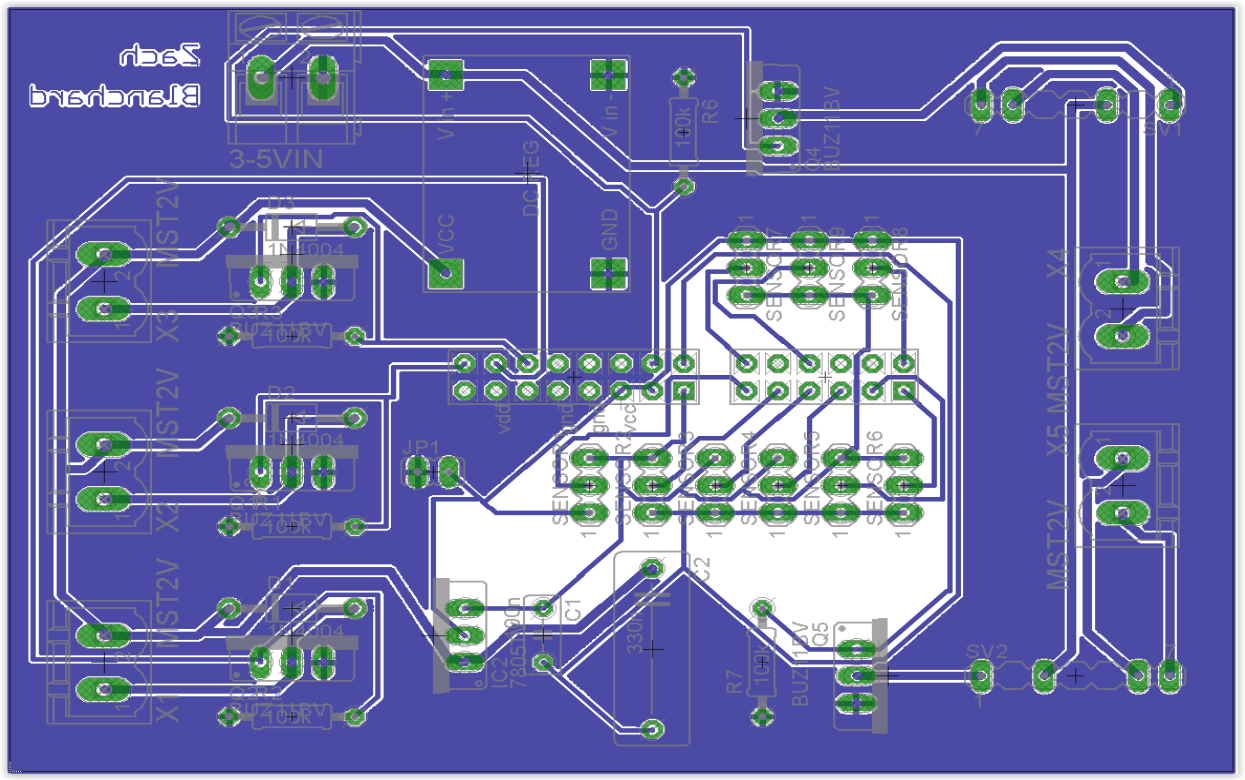
Max Power consumption:

70W maximum

Schematic:



Board Layout:



Programming

Hydro.c

```
/****************** Defines and Includes *****************/
#include "derivative.h"
#include "my_io_apps.c"
#include "my_interrupts.c"
#include "mcg.h"

#define TURN_OFF_LED_ARRAY1 gpo_e(4,0)
#define TURN_ON_LED_ARRAY1  gpo_e(4,1)
#define TURN_OFF_LED_ARRAY2 gpo_e(5,0)
#define TURN_ON_LED_ARRAY2  gpo_e(5,1)

#define TURN_OFF_PUMP1 gpo_b(8,0)
#define TURN_ON_PUMP1  gpo_b(8,1)
#define TURN_OFF_PUMP2 gpo_b(9,0)
#define TURN_ON_PUMP2  gpo_b(9,1)
#define TURN_OFF_PUMP3 gpo_b(10,0)
#define TURN_ON_PUMP3  gpo_b(10,1)


/**************Global Variables Here************************/

int x;

int ticks=0;       // Measurement for 1ms during program

int Moisture_Level[3][3];  // Lowest data on moisture data

int Percent;

int Hour_Timer=0; // Time variable for lighting

int main(void) {

          int pll_err=pll_init(8000000,0,1,4,24,1);     //Use the crystal oscillator to drive the PLL

           syst_tick_init();


//******Chip initializations**************************************

           adc0_init();

           init_gpio_b(8,1); // Pump switch pins
           init_gpio_b(9,1);
           init_gpio_b(10,1);

           init_gpio_e(4,1); // Light switch pins
           init_gpio_e(5,1);


//*************************************************************
```

Hydro.c

```
55
56      for(;;) {
57
58          if(Hour_Timer<=34){  // Turns LED arrays on for 18 hours.
59              TURN_ON_LED_ARRAY1;
60              TURN_ON_LED_ARRAY2;
61          }
62          if(Hour_Timer>34){   // Turns LED arrays of for 6 hours.
63              TURN_OFF_LED_ARRAY1;
64              TURN_OFF_LED_ARRAY2;
65          }
66          if(Hour_Timer>=47){Hour_Timer=0;}  // Resets the time for a day.
67          MOISTURE_PERCENT();
68
69  //
70          if(Moisture_Level[0][0]<60 || Moisture_Level[0][1]<60 || Moisture_Level[0][2]<60){ // Check moisture percentage. If lower then defined set point turn one pump on for defined amount of time.
71              TURN_ON_PUMP1;
72              delay_ms(300000);  //pump comes on for 5 minutes if moisture is to low
73              TURN_OFF_PUMP1;
74          }
75          if(Moisture_Level[1][0]<60 || Moisture_Level[1][1]<60 || Moisture_Level[1][2]<60){ // Check moisture percentage. If lower then defined set point turn one pump on for defined amount of time.
76              TURN_ON_PUMP2;
77              delay_ms(300000); //pump comes on for 5 minutes if moisture is to low
78              TURN_OFF_PUMP2;
79          }
80          if(Moisture_Level[2][0]<60 || Moisture_Level[2][1]<60 || Moisture_Level[2][2]<60){ // Check moisture percentage. If lower then defined set point turn one pump on for defined amount of time.
81              TURN_ON_PUMP3;
82              delay_ms(300000);  //pump comes on for 5 minutes if moisture is to low
83              TURN_OFF_PUMP3;
84          }
85
86
87          while(sys_ticks<1800000); // One half hour delay
88          Hour_Timer++;
89          sys_ticks=0;
90      }
91
92      return 0;
93      }
94
```

Hydro.c

```
93      }
94
95  void MOISTURE_PERCENT (void){
96
97  /****************** Row 1 *****************/
98
99      x=adc0_value(8);        //Sensor 1
100     Percent=(100-((((float)x-2313.0)/(1800.0))*100.0)); //calibrated formula for each sensor to give % moisture in the medium
101     Moisture_Level[0][0]=Percent;
102
103 //  Moisture_Level[0][0]=adc0_value(8);
104
105     x=adc0_value(9);        //Sensor 2
106     Percent=(100-((((float)x-2207.0)/(1900.0))*100.0));
107     Moisture_Level[0][1]=Percent;
108
109 //  Moisture_Level[0][1]=adc0_value(9);
110
111     x=adc0_value(12);       //Sensor 3
112     Percent=(100-((((float)x-2313.0)/(1800.0))*100.0));
113     Moisture_Level[0][2]=Percent;
114
115 //  Moisture_Level[0][2]=adc0_value(12);
116
117 /****************** Row 2 *****************/
118
119     x=adc0_value(13);       //Sensor 4
120     Percent=(100-((((float)x-2538.0)/(1600.0))*100.0));
121     Moisture_Level[1][0]=Percent;
122
123 //  Moisture_Level[1][0]=adc0_value(13);
124
125     x=adc0_value(11);       //Sensor 5
126     Percent=(100-((((float)x-2343.0)/(1800.0))*100.0));
127     Moisture_Level[1][1]=Percent;
128
129 //  Moisture_Level[1][1]=adc0_value(11);
130
131     x=adc0_value(15);       //Sensor 6
132     Percent=(100-((((float)x-2334.0)/(1800.0))*100.0));
133     Moisture_Level[1][2]=Percent;
134
135 //  Moisture_Level[1][2]=adc0_value(15);
```

```
136
137 /****************** Row 3 *****************/
138
139     x=adc0_value(0);        //Sensor 7
140     Percent=(100-((((float)x-2278.0)/(1900.0))*100.0));
141     Moisture_Level[2][0]=Percent;
142
143 //  Moisture_Level[2][0]=adc0_value(0);
144
145     x=adc0_value(3);        //Sensor 8
146     Percent=(100-((((float)x-2250.0)/(1900.0))*100.0));
147     Moisture_Level[2][1]=Percent;
148
149 //  Moisture_Level[2][1]=adc0_value(3);
150
151     x=adc0_value(23);       //Sensor 9
152     Percent=(100-((((float)x-2306.0)/(1800.0))*100.0));
153     Moisture_Level[2][2]=Percent;
154
155 //  Moisture_Level[2][2]=adc0_value(23);
156     return(0);
157 }
158
```

Troubleshooting

- Is the power plugged in? If not connect the power
- Insure all screw terminals are tighten and that the wires are in the terminals
- Check LED indicators on the moisture sensors to make sure they are powered. If not check the connection pins on the control board. The +/- pins are labeled and should read 5V
- Check 12V DC-DC converter with a volt meter to make sure that it was on adjusted in transit. VCC out to ground should read 12V.
- Unplug and re-plug in the board to reset the programming
- Connect the Freescale board to your computer via USB and check the MOISTURE_LEVEL[3][3] array to make sure readings are taking place. Test in both air and water. There is also a commented out section for each sensor should you want to see the direct reading from the ADC converter and make any calibrations from that
- Contact us for any further questions or for a replacement