Water Level Controller With Dry Pump Protection

Designed By:

Fahad Iftikhar

Case Study:

Two families sharing a same house in a society, family 1 (f1) Living on ground floor and family 2 is living on the first floor are facing some water problems. The house is being supplied by an overhead water tank (T1) located outside the house in a park managed by the housing society. The water is supplied to the houses located in the society for limited periods (P_i) of time and the schedule is not fix. The start and stop time of those intervals may vary according to season. Now during the time when water supply from the society through tank T1 is operational the family F1 living on the ground floor wants to use that water. The pressure is not appropriate to increase the water level up to the first story. So the family F2 living on the first story must use some mechanism to store water in an overhead tank (T2) placed on the roof top of the house. The electric motor which must be used to pump water in T2 should not be ON all the time and cannot be connected to the supply pipe which carries water from T1 to the house.

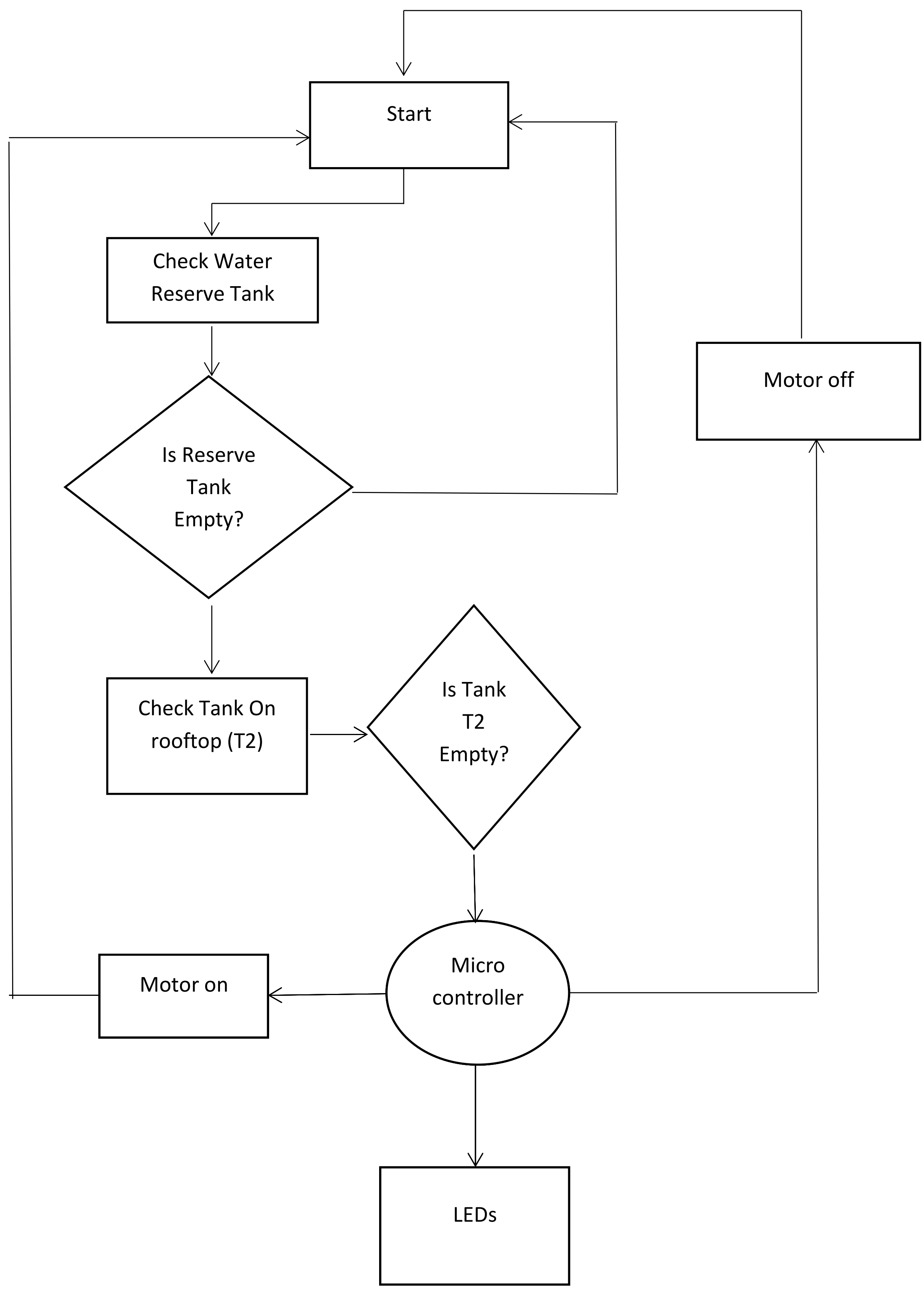
Problem:

The problem here is that water pressure is not suitable to increase the water level up to the first story to family (f2). As, the supply of water have no allotted time, not fixed with limited periods and is even vary on season so you cannot connect a water pump motor to the main water supply which carries water from t1 to the house. So, we have to raise the water to tank T2 without even disturbing the family on ground floor.

Solution:

The solution we provide to this problem is using a reserve tank which will store water for family 1 on the ground floor and by using sensors and microcontrollers it will sense the quantity of water in reserve tank if it is appropriate the motor will turn on and the water will raise to the tank t2. If water is not up to the level in reserve tank the motor will not turn on. The solution is affordable, autonomous and have a low running cost.

Flow Chart:


Start
Check Water
Reserve Tank
Is Reserve
Tank
Empty?
Motor off
Check Tank On
rooftop (T2)
Is Tank
T2
Empty?
Motor on
Micro
controller
LEDs

Explanation of Flowchart:

Initially, one sensor will check the water level of reserve tank. If the reserve tank is empty the sensors will not proceed and the motor will be remained off. If the reserve tank is not empty and filled up to the desired numbers the other four water sensors will start checking the Tank T2. The microcontroller will take the decision after sensing if t2 is empty, the microcontroller will turn on the motor and if t2 is full the microcontroller will turn off the motor.

Note: The motor is not attached to water supply of the society it is attached with the reserve tank.

Components:

Water Level Sensor:

Water Level sensors are used to detect the level of substances that can flow. Such substances include liquids, slurries, granular material and powders. Level measurements can be done inside containers or it can be the level of a river or lake.

We use four water level sensors for four different water level indication in Tank 2 (T2) and one sensor in reserve tank for the point where we can decide the controller work.

LEDs:

A light-emitting diode (***LED***) is a semiconductor light source that emits light when current flows through it.

LEDs are used for water level indication. By turning On/Off LEDs will show the level of water in the tank.

Water Pump:

The **water pump** can be defined as a **pump** which uses the principles like mechanical as well as hydraulic throughout a piping system and to make sufficient force for its future use.

The water pump is connected to reserve tank to produce an appropriate pressure for water to raise to the T2 tank n first floor. When reserve tank is not empty the water pump will turn on and pump the water to raise to the first floor.

Relay:

Relays are used for isolating a low voltage circuit from high voltage circuit. They are used for controlling multiple circuits. They are also used as automatic change over. Microprocessors use relays to control a heavy electrical load.

The electromagnetic relay is used as an electrical amplifier as the microcontroller, LEDs and other components work on low voltage and water pump will work on 220 AC voltages. So, a Relay is used as an interface device between microcontrollers and high voltage devices.

The Negative side of motor is connected to output side of relay and the positive side of the cable will be connected to 220V AC voltages.

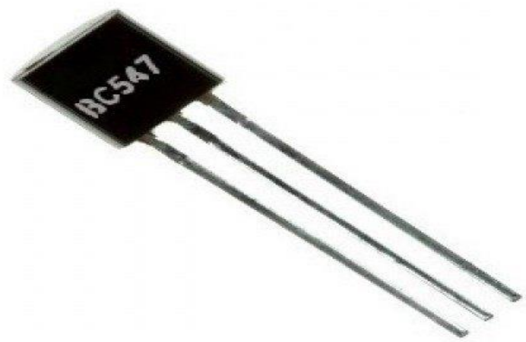
Resistors:

It is a device having resistance to the passage of an electric current.

The resistors are used to limit the current so no LEDs and water sensor burn.

Transistors:

A **transistor** is a miniature electronic component that can **do** two different jobs. It can **work** either as an amplifier or a switch: When it **works** as an amplifier, it takes in a tiny electric current at one end (an input current) and produces a much bigger electric current (an output current) at the other



Microcontrollers:

PIC16F84 Microcontroller

PIC16F84 Microcontroller is widely used electronic gadgets and machines. It seems like it is the ancestor of all the PIC micro-controllers. PIC16F84 is 18-pin Microcontroller chip. Furthermore, it is the “brain” of any project; the PIC can be accompanied with sensors and decides whether devices like motor and relays can be activated or not.

why pic16f84a microcontroller is better than other

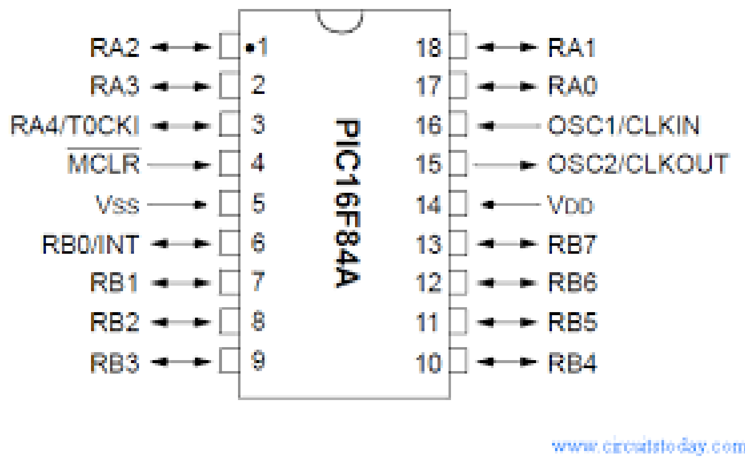
The performance of the **PIC microcontroller** is very fast because of using RISC architecture. When comparing to **other microcontrollers**, power consumption is very less and programming is also very easy. Interfacing of an analog device is easy without any extra circuitry.

Introduction to PIC16F84A microcontroller

- PIC16F84A is also known as the beginners' microcontroller.
- It contains high performance RISC CPU.
- This microcontroller from Microchip was presented in 1998 as a successor to the specific first sequentially programmable PIC, the PIC16C84.
- This is considered to be a starting pickup for learning PIC microcontrollers programming. because it only contains 35 assembly language instructions as well as it only costs less than $2 per piece.
- All instructions are takes one cycle to complete except branch instructions. Size of instruction is 8-bit.
- Maximum operating frequency is 20MHz. But it can be operated on lower frequency also to save power.
- It is an 8-bit microcontroller. Therefore, the size of data bus is 8 bits.
- Creating a serial programmer for this microcontroller also won't take a lot of time.
- The PIC16F84A is an 8-bit device which implies the majority of its registers are 8 bits wide.
- It has 1024 words of program memory. If you are working on simple applications then 1024 words of program memory is enough, and this is where PIC16F84A is frequently used.
- Single pin current sourcing and sinking limit is 25mA.

PIN Diagram of PIC16F84A

This is an 18 pin IC; the description of each pin is given below:



- It has thirteen I/O pins and each pin can be used either as digital input or digital output.
- These 13 I/O pins are shared with PORTA and PORTB.
- PORTA consists of 5 pins from RA0-RA4 and PORTB has 8 pins from RB0-RB7.

Circuit Diagram

Programming: -

For Water Level indication:

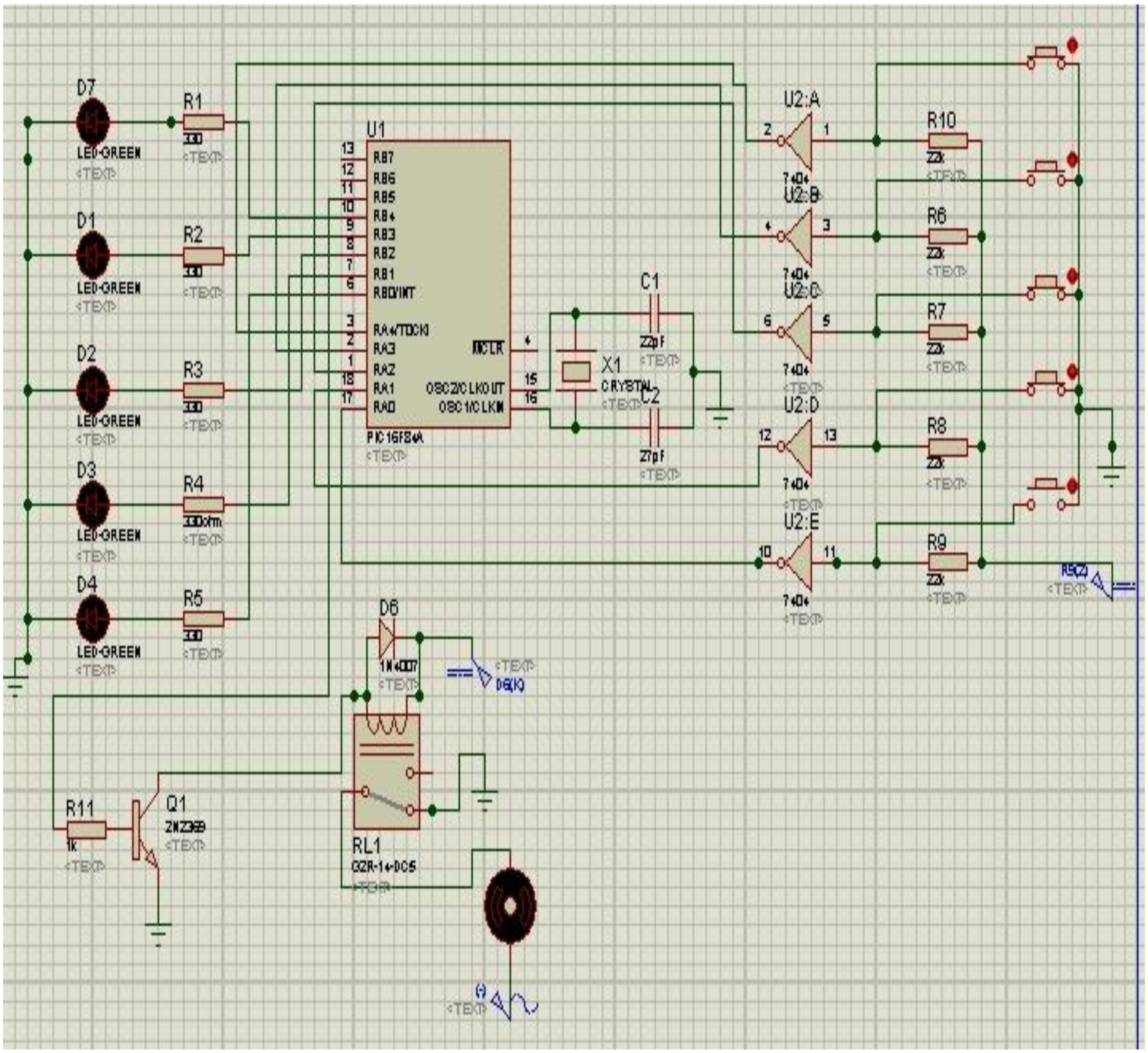
```
#include <iostream>
#include <windows.h>

void Start()
{   int RA0=0;          //Inialize 0 to port RA0
    int RA1=0;          //Inialize 0 to port RA1
    int RA2=0;          //Inialize 0 to port RA2
    int RA3=0;          //Inialize 0 to port RA3
    int RA4=0;          //Inialize 0 to port RA4
    int RB0=0;          //Inialize 0 to port RB0
    int RB1=0;          //Inialize 0 to port RB1
    int RB2=0;          //Inialize 0 to port RB2
    int RB3=0;          //Inialize 0 to port RB3
    int RB4=0;          //Inialize 0 to port RB4
    int RB5=0;          //Inialize 0 to port RB5
}
int main()
{
        int RA0;        //Inialize 0 to port RA0
        int RA1;        //Inialize 0 to port RA1
        int RA2;        //Inialize 0 to port RA2
        int RA3;        //Inialize 0 to port RA3
        int RA4;        //Inialize 0 to port RA4
        int RB0;        //Inialize 0 to port RB0
        int RB1;        //Inialize 0 to port RB1
        int RB2;        //Inialize 0 to port RB2
        int RB3;        //Inialize 0 to port RB3
        int RB4;        //Inialize 0 to port RB4
        int RB5;        //Inialize 0 to port RB5
        int RT;
        int T2;
        Start();
        if(RA4==1)              //If Reserve tank is not empty
            {
             RB0=1;         // LED 1 on
             if (RA01:40 AM 6/4/2020==0)       // Tank 2 is empty
                {
                 RB5=1;         // Motor on
                 if(RA0==1)       //Water reach sensor 1
                   {
                    RB1=1;      // LED 2 on
                   }
                 else
                   {
                    RB1=0;              // LED 2 off
                    Sleep(300000);  //Delay of 5 min
                   }
                 if(RA1==1)      //Water reach sensor 2
                   {
                    RB2=1;      // LED 3 on
                   }
```

```
            }
            else
              {
                RB2=0;          // LED 3 off
                Sleep(300000); //Delay of 5 min
              }
            if(RA2==1)     //Water reach sensor 3
              {
                RB3=1;     // LED 4 on
              }
            else
              {
                RB3=0;          // LED 4 off
                Sleep(300000); //Delay of 5 min
              }
            if(RA3==1)     //Water reach sensor 4
              {
                RB3=1;     // LED 5 on
              }
            else
              {
                RB3=0;         // LED 5 off
                Sleep(2000); //Delay of 2 seconds
              }
            Sleep(300000);      //Delay of 5 min
          }
        else                   //T2 is full
          {
            RB5=0;          //Motor off
            Sleep(300000);  //Delay of 5 min
            Start();       //Going to start
          }
      }
    else                     //Reserve Tank is empty
     {
      RB0=0;                 //LED 1 off
      Sleep(300000);        //Delay of 5 min
      Start();              //Going to start
      }
}
```

Explanation of Code:

In this code first we have initialized the ports port RA1, RA2, RA3, RA4, RB0, RB1, RB2, RB3, RB4, RB5. After that we have used If Else Statements for further coding. In this we have initialized RT equal to 1 to check whether the Reserve tank which is in basement is empty or not. If the reserve tank has water in it 1 Led will ON to that there is water in the reserve tank. And When it says 1 that the reserve tank is not empty then it will go to the next if statement. In that statement we have initialized T2 equal to 0 to check whether the tank 2 in empty or not. And for this purpose we have 4 led and 4 water sensors in the tank which will work with a coordination like if the T2 is equal to 0 it means tank 2 is empty it will send 1 to RB5 which will turn the motor on in Reserve tank to fill the tank 2 with water. When the water reach ¼ of the tank 2 the water sensor senses the water and turn Led 1 ON. After that there will a delay of 5

Simulation of Circuit



Microcontroller Programming

```
void main()
{
  TRISB = 0x00;            //  set direction to be output
  TRISA = 0xFF;            // set direction to be output
 do
{
      if(PORTA.F4==1)              //If Reserve tank is not empty
         {
          PORTB.F4=1;           // LED 1 on
          delay_ms(100);
          if  (PORTA.F0==0)       // Tank 2 is empty
            {
              PORTB.F5=1;       // Motor on
              if(PORTA.F0==1)      //Water reach sensor 1
                {
                  PORTB.F0=1;    // LED 2 on
                }
              else
                {
                  PORTB.F0=0;           // LED 2 off
                  delay_ms(300000); //Delay of 5 min
                }
              if(PORTA.F1==0)    //Water reach sensor 2
                {
                  PORTB.F1=1;   // LED 3 on
                }
              else
                {
                  PORTB.F0=0;         // LED 3 off
                  delay_ms(300000); //Delay of 5 min
                }
              if(PORTA.F2==0)    //Water reach sensor 3
                {
                  PORTB.F2=1;     // LED 4 on
                }
              else
                {
                  PORTB.F2=0; ;         // LED 4 off
                  delay_ms(300000); //Delay of 5 min
                }
```

```
              }
              if(PORT1.F3==1)    //Water reach sensor 4
                {
                  PORTB.F3=1;    // LED 5 on
                }
              else
                {
                  PORTB.F3=0;         // LED 5 off
                  delay_ms(2000); //Delay of 2 seconds
                }
              delay_ms(300000); //Delay of 5 min
             }
          else                //T2 is full
            {
              PORTB.F5=0;           //Motor off
              delay_ms(300000); //Delay of 5 min

             }
         }
       else                       //Reserve Tank is empty
        {
         PORTB.F4=0;                 //LED 1 off
         delay_ms(300000);       //Delay of 5 min

         }
}
}
  while(1);            // Endless loop
}
}
```

EXPERIMENTAL RESULT OF WATER LEVEL SENSING UNIT

Input From Water Sensor		Output						
Res Tank	Water Tank	LED 1	LED 2	LED 3	LED 4	Tank	Motor	Reserve Tank
0	0000	OFF	OFF	OFF	OFF	Empty	OFF	Empty
1	0000	ON	OFF	OFF	OFF	Empty	ON	Water Exist
1	1000	ON	ON	OFF	OFF	$\frac{1}{4}$	NO OP.	Water Exist
1	1100	ON	ON	ON	OFF	$\frac{2}{4}$	NO OP.	Water Exist
1	1110	ON	ON	ON	ON	$\frac{3}{4}$	NO OP.	Water Exist
1	1111	ON	ON	ON	ON	Full	ON	Water Exist