Ing. de mantenimiento de Computadores y Redes

Sistema embebido para control automatizado de bombas de agua con protección contra el funcionamiento en vacio.

D. José Vicente Lozano Copa
Universidad de Alicante
Jvlc3@alu.ua.es
Marzo de 2016

Descripción del problema

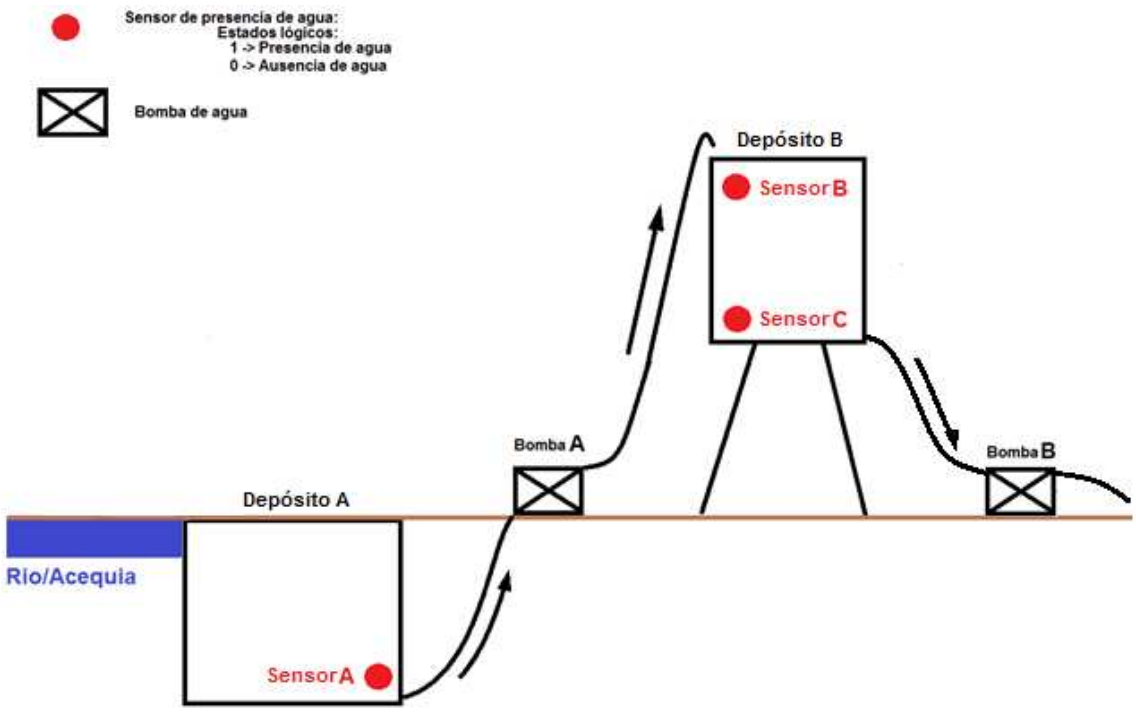
El actual proyecto consiste en el diseño e implementación de un circuito impreso en una placa PCB para el control de la lógica que gobierna las bombas de agua de un depósito elevado y la bomba de riego que utiliza dicha agua para un regadío por goteo.

En el escenario modelo existen dos bombas de agua y dos depósitos tal como se muestra en la figura siguiente. Cuando el depósito B alcanza el nivel mínimo de agua, la bomba correspondiente (A) debe activarse para llenar dicho depósito. La bomba no debe activarse por el hecho de bajar el nivel de agua por debajo del nivel máximo, sino cuando se alcance el mínimo.

La activación de la bomba B para comenzar el riego será manual por el operario del huerto.

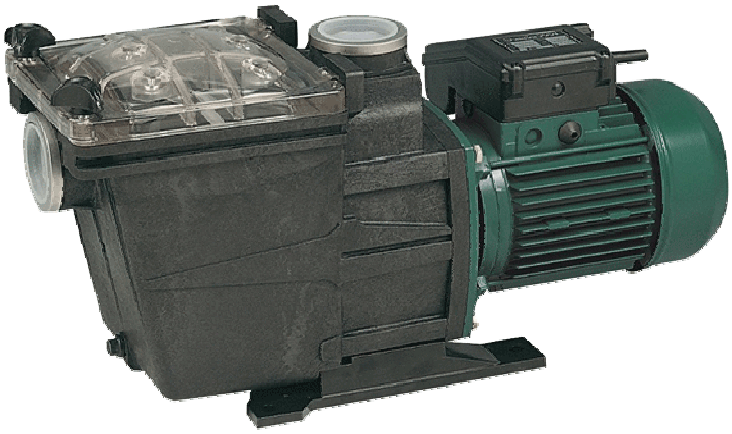
Las bombas no deben funcionar si el punto del cual toman agua está por debajo del nivel mínimo, pues en caso de funcionar en vacio sufrirán daños.

Representación del escenario



Bombas de agua

Las bombas de agua instaladas requieren una gran corriente de 8.6A cada una, por lo que en caso de estar activas ambas bombas la intensidad puede alcanzar los 17.2A. Por tanto las pistas por las que circula dicha corriente deberán tener un grosor considerable.



Voltaje 220 AC
Max 8.6 Amperios
2800 RPM

Composición del sistema.

- 1 Fuente de alimentación regulada 220V - 5V
- 1 Micro controlador AMTEL ATMega328-p
- 2 Relés 5V a 220v 10A (Relé A, Relé B)
- 3 Sensores de presencia de agua (Sensor A, Sensor B, Sensor C)
- 1 Interruptor de riego (Sensor R)

Requisitos funcionales:

- Cuando el nivel de agua del depósito B baje por debajo del sensor C, C=0, la bomba A se activará hasta que el nivel del agua del depósito alcance el sensor B, B=1;
- Si el nivel del agua en el depósito A está por debajo del sensor A, la bomba A no trabajará para evitar el funcionamiento en vacio.
- Cuando el operario active el botón de riego (R [1,0]) la bomba B se activará.
- Si el nivel de agua del depósito B está por debajo del sensor C, C=0, la bomba B no trabajará para evitar el funcionamiento de esta en vacio.

Diagrama de estados de la bomba A

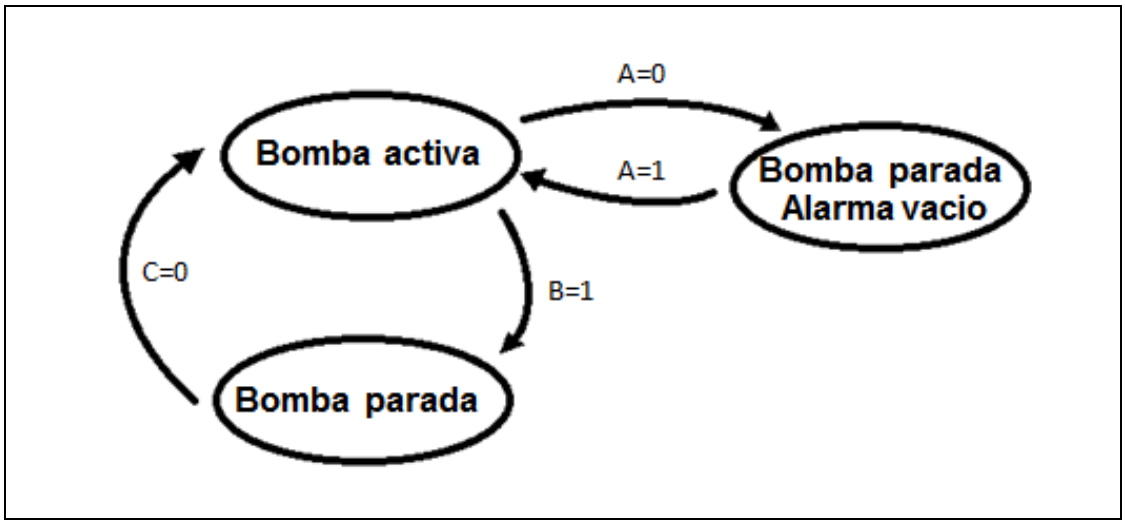
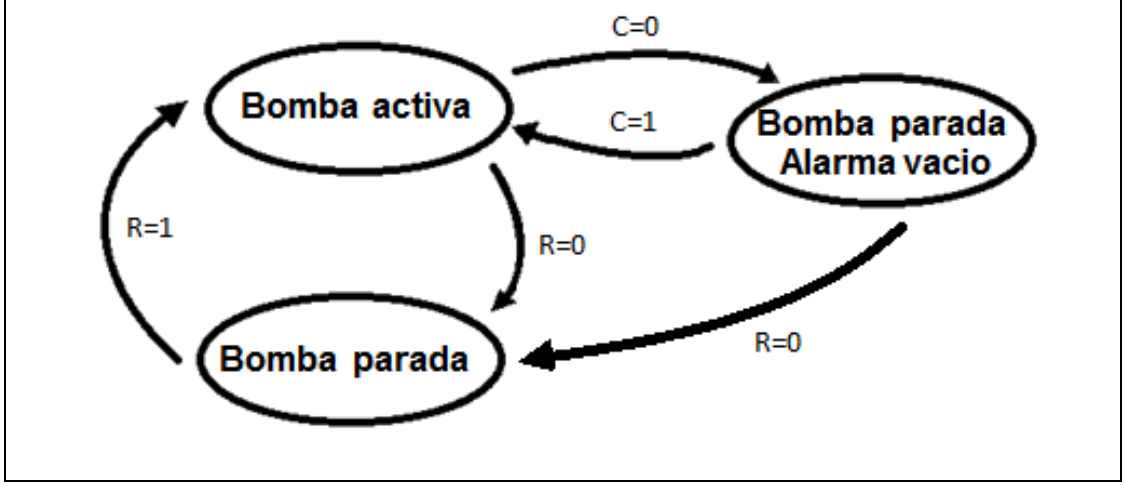
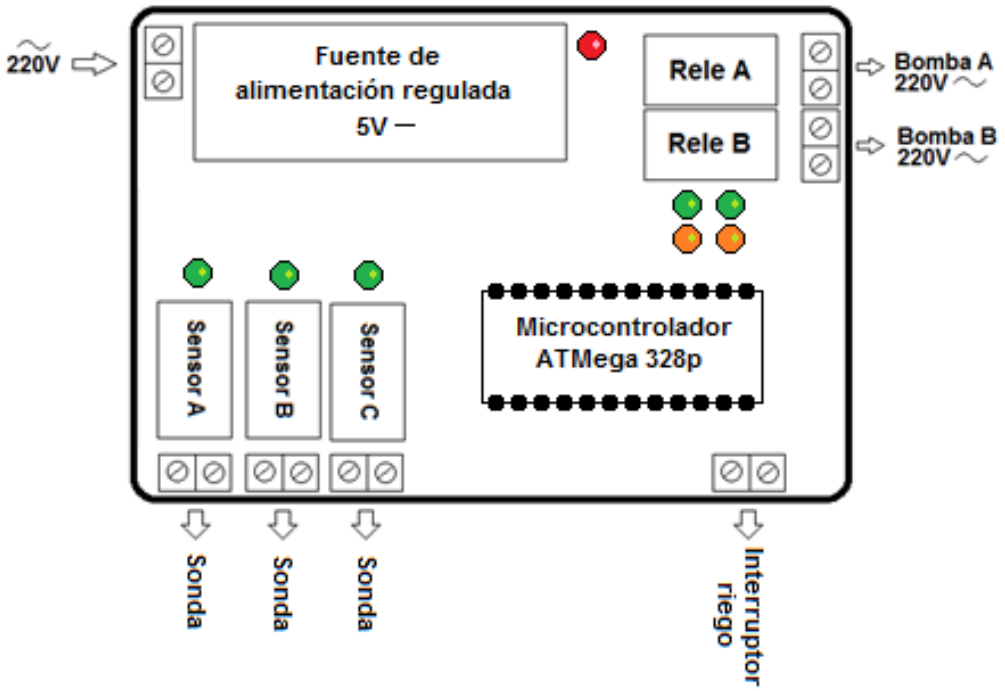


Diagrama de estados de la bomba B



Aproximación al diseño



La fuente de alimentación proporcionará 5V estabilizados para alimentar el resto de los elementos del circuito.

Los sensores A, B y C llevarán incorporada una sonda que se sumergirá en los depósitos a la altura deseada, indicando al circuito la presencia o ausencia de agua.

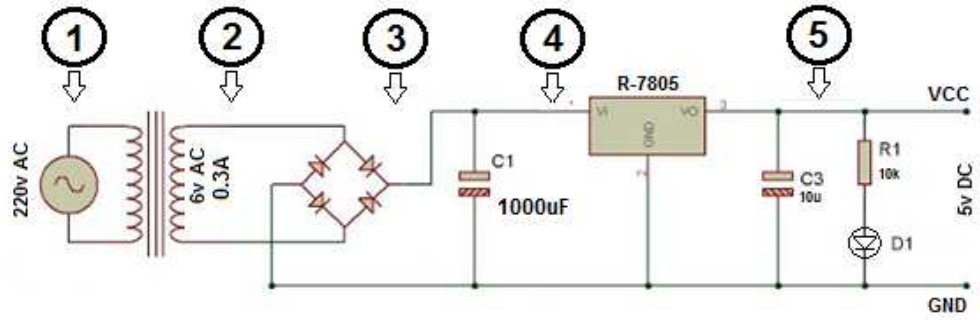
El interruptor de riego serán unos pines de conexión que al cerrarse indicarán al micro controlador la intención de riego del operario. Se implementa con pines y no con un interruptor para dar al operario la posibilidad de instalar un interruptor con temporizador.

Los relés A y B activarán o desactivarán las bombas de agua, permitiendo el flujo de 220V AC.

El micro controlador ATMega-328P implementará el firmware con la lógica que gobierna el resto de los componentes del sistema.

Diseño de la Fuente de alimentación

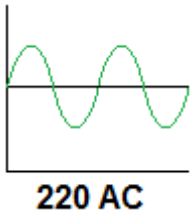
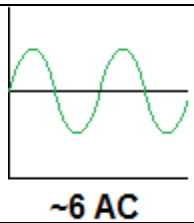
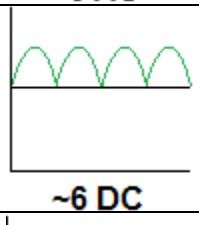
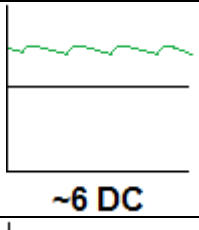
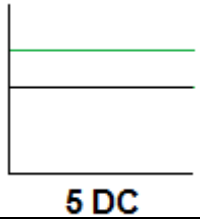
Esquema eléctrico



Calculo del condensador

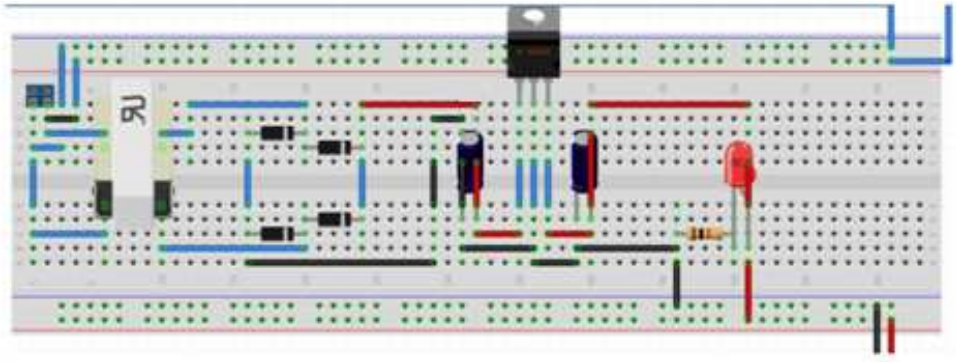
$$C = \frac{I_{max}\ T}{V_{max} - V_{min}} = \frac{0.3A \cdot 0.02S}{6V - 0V} = 1000uF$$

Fases de la transformación

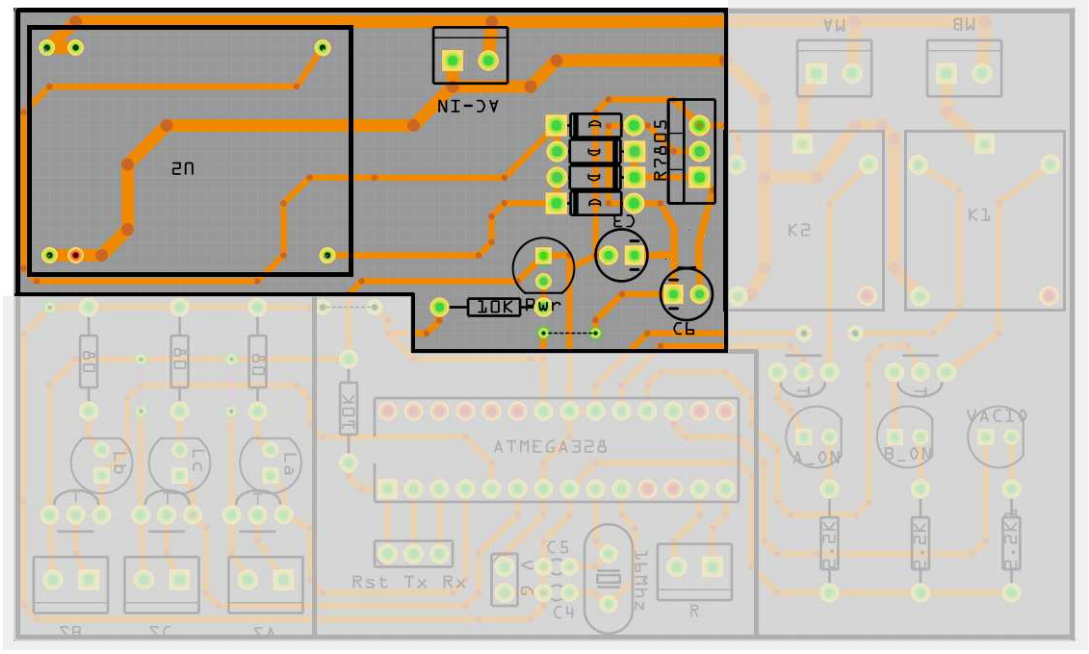
1		Inicialmente la corriente de entrada son 220V alterna (50-60Hz)
2		El transformador reduce el voltaje efectivo a 6V alterna. (50-60Hz).
3		El puente de diodos o puente rectificador encamina la corriente para que fluya en un único sentido eliminando la componente alterna por lo que pasa a DC. (corriente continua)
4		El condensador electrolítico mantiene el voltaje durante la parte baja del ciclo, aproximando más la forma de la onda a una corriente continua.
5		El regulador de voltaje 7805 asegura un voltaje continuo de 5V. Existen otras variantes de este regulador para diferentes voltajes. 7806 - 6V 7809 - 9V 7812 - 12V...

La fuente de alimentación estabilizada proporciona 5V estables al resto del circuito

Implementación de la fuente en protoboard



Implementación en la PCB



Materiales

1 Transformador
- 220v - 6V 0.3A

1 puente rectificador
- (4 Diodos 1A)

1 Condensador electrolítico
- 1000uF
- 16V

1 Condensador electrolítico
- 10uF
- 25V

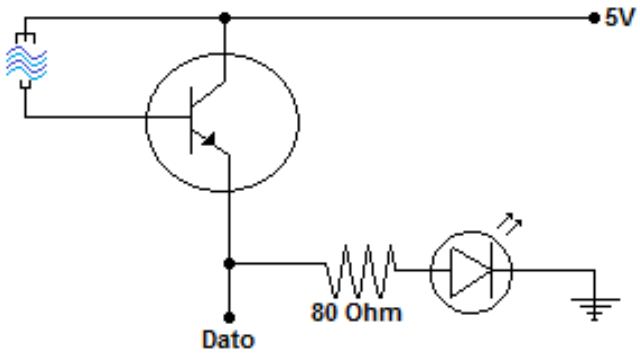
1 Regulador de voltaje 7805

1 Resistencia 10k ± 5%

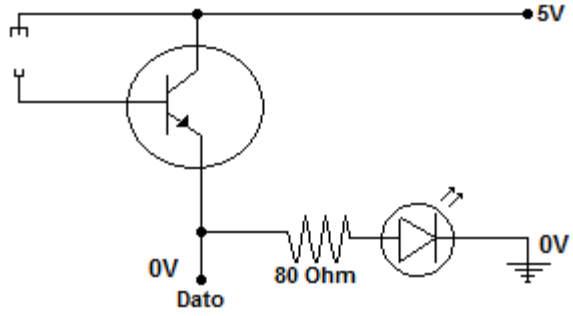
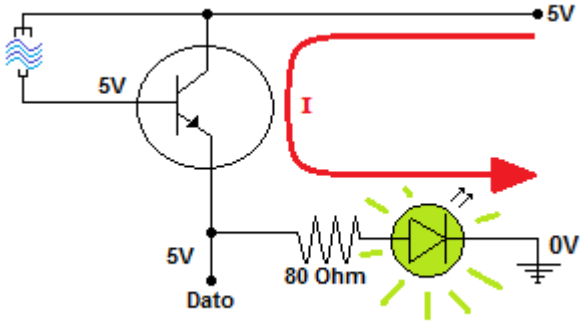
1 Diodo LED (Rojo)

Electrodo de presencia de agua

Esquema eléctrico



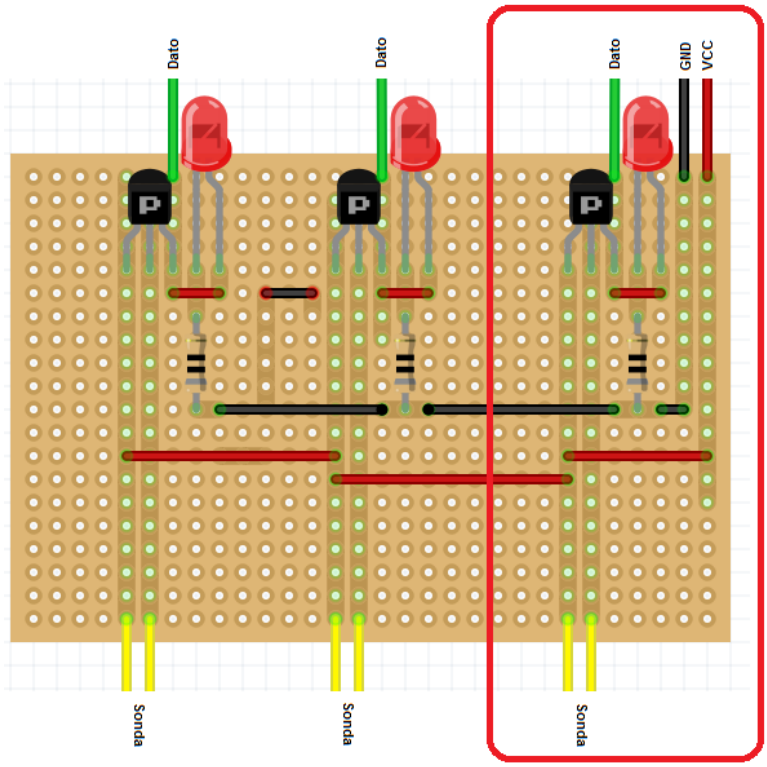
Funcionamiento del sensor de agua.

	En ausencia de agua, el transistor NPN permanece cerrado, por lo que el dato tiene el voltaje de tierra y el diodo LED permanece apagado al no haber flujo de corriente.
	Al estar el sensor en contacto con agua, la base del transistor NPN se polariza positivamente, lo que abre el transistor y permite el flujo de corriente desde el colector hacia el emisor. En este estado, el dato estará a 5V y el LED se iluminará al haber flujo de corriente desde 5V a tierra.

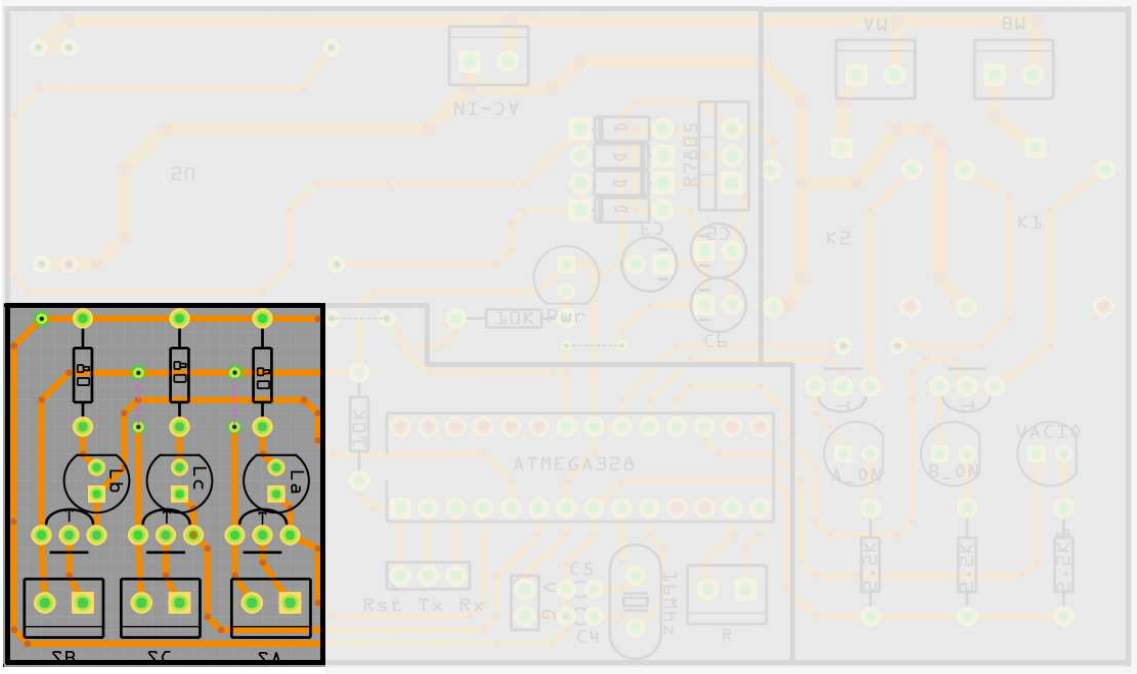
Es interesante tener en cuenta que el nodo de Dato no se encuentra a 5V cuando el sensor está en presencia de agua, el propio transistor tiene una resistencia que produce una caída de aproximadamente 0.7V, por lo que en realidad se encuentra a 4.3V aproximadamente. En cualquier caso, al estar por encima de (70% 5V) 3.5V, es interpretado como un 1 lógico por el micro controlador.

Existe un error en el diseño del sensor por el que resulta demasiado sensible, la implementación de la solución está explicada al final de este documento.

Implementación en protoboard

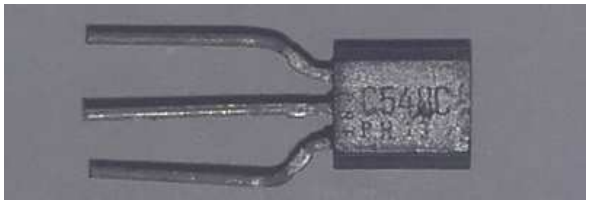


Implementación en la placa PCB



Materiales x3

1 Transistor NPN
- C548C



1 Resistencia 80 Ohm

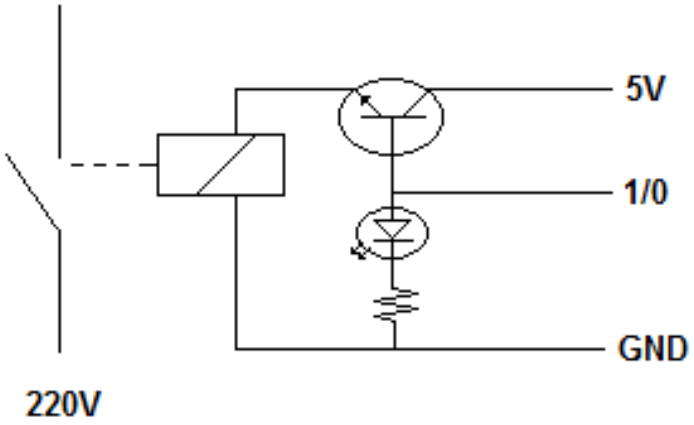
1 diodo LED

1 Sonda

Activación de las bombas (Relés)

La señal digital enviada por el ATMega abrirá un transistor NPN, permitiendo el flujo de 5V que activará el relé. Una vez activo el relé, la circuito de 220V quedará cerrado, poniendo en funcionamiento la bomba de agua.

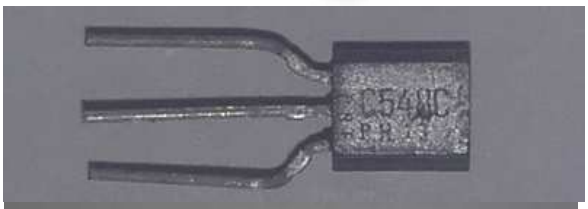
Esquema eléctrico



Componentes (x2)

1x Rele 5V -> 220V 10A
(Finder 36.11.9.005.4011) 70 Ohm

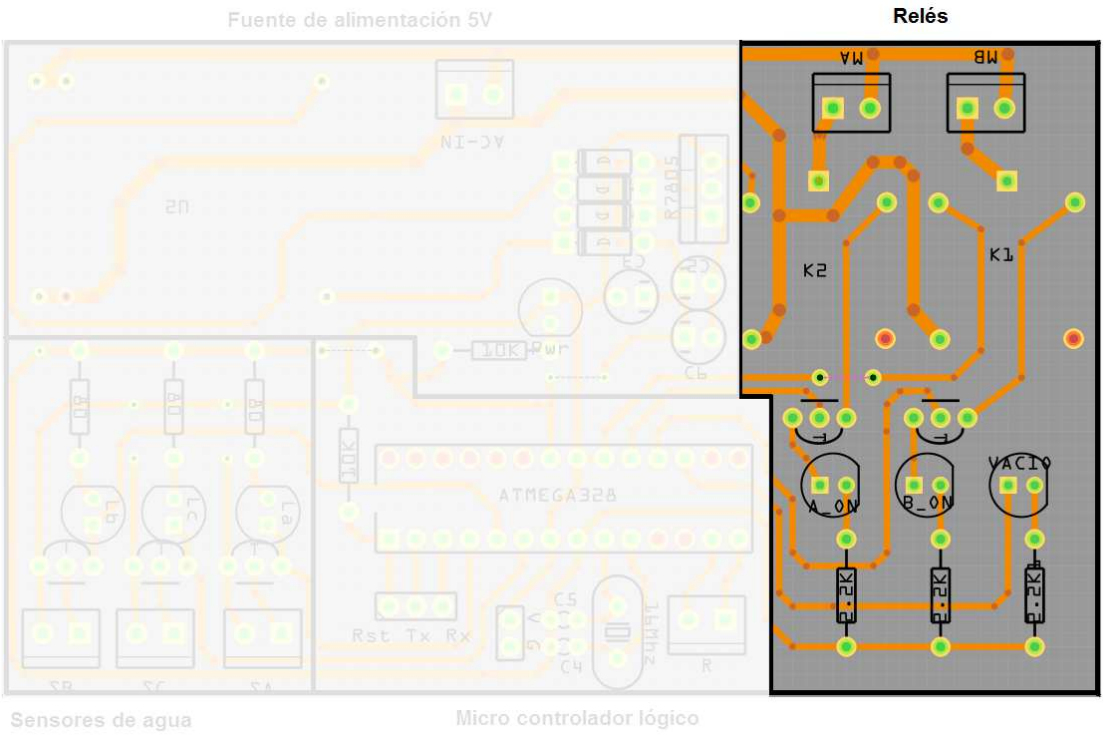
1x Transistor NPN



1x Diodo LED (Verde)

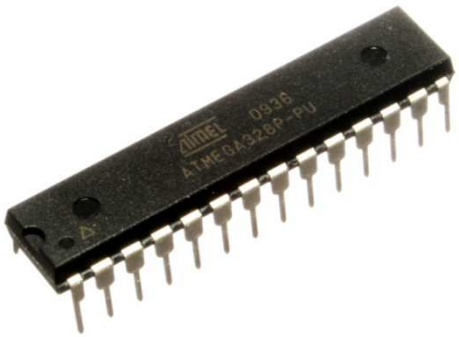
1x Resistencia 220 Ohm

Implementación



En esta parte del circuito se ha implementado también un diodo LED que indicará la alarma de vacío, este LED está conectado directamente a la salida digital del ATMega haciendo fluir una pequeña corriente a través de él.

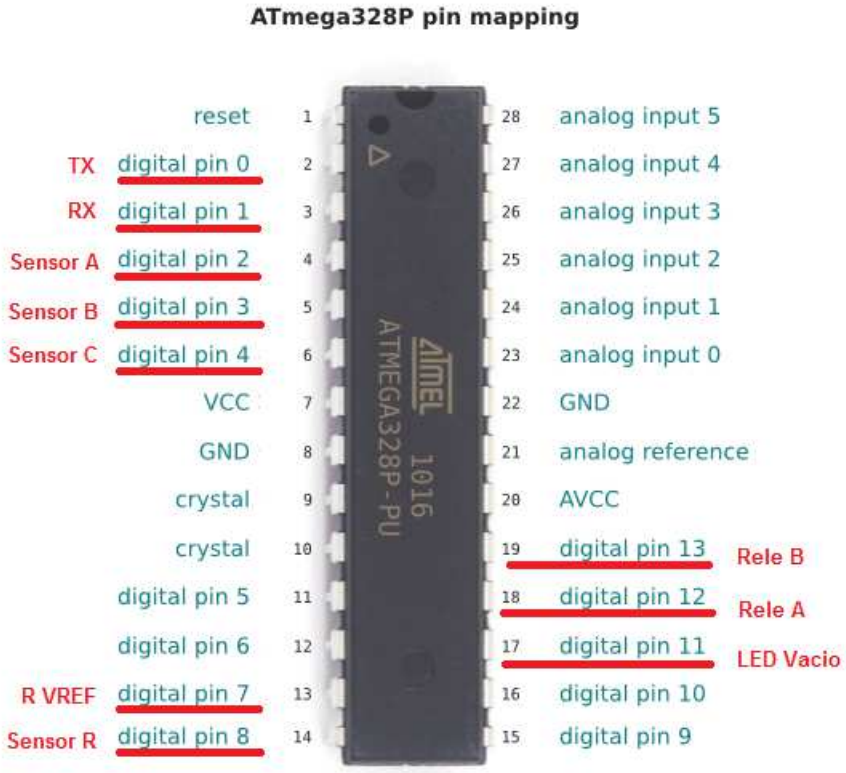
Micro Controlador Amtel ATMega 328p



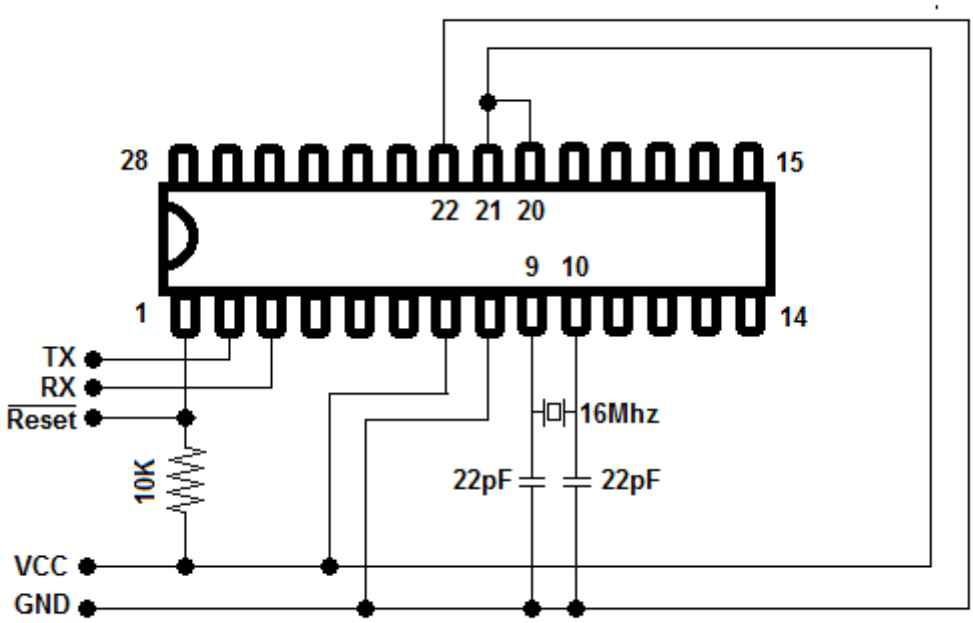
Micro controlador 8bits
1.8V - 5.5V
Arquitectura RISC
131 Instrucciones
32 Registros
Max 20MHz
32KBytes flash programables
23 Lineas I/O programables

Consumo a 1MHz 0.3mA

Datasheet



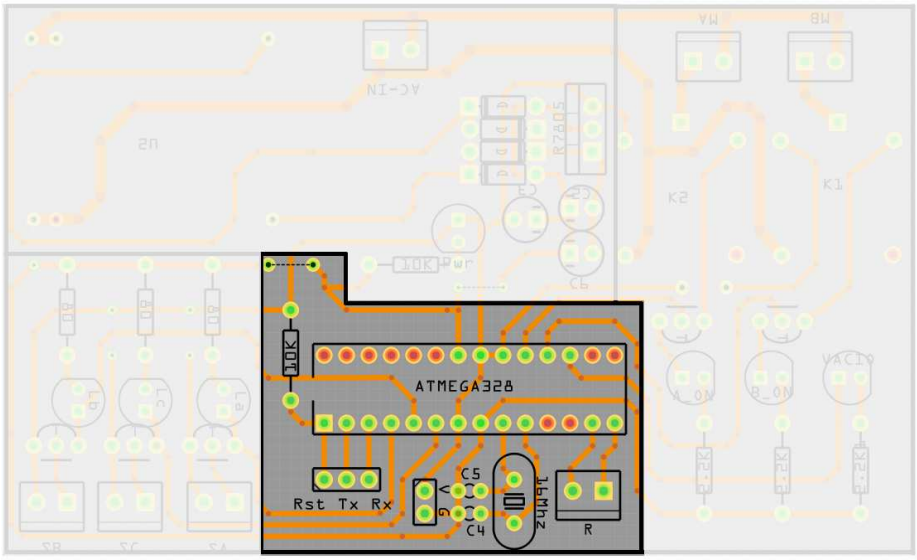
Esquema eléctrico de conexión del ATMega328P



Se debe conectar el pin 1 con VCC por medio de una resistencia de 10K, el micro controlador entra en modo programación cuando se recibe una señal digital 0 por dicho pin, durante el funcionamiento normal, la conexión a VCC mantendrá un 1 lógico.

El micro controlador ATMEga328P incorpora un oscilador de cuarzo interno de 8Mhz con el que funcionaría en caso de no conectar un oscilador externo. En este proyecto se ha incorporado un oscilador de 16Mhz conectado a tierra por medio de dos condensadores cerámicos de 22 pF tal y como aconsejan las especificaciones del fabricante.

En el anterior esquema se ha conectado el pin 20 con VCC aunque no es necesario para el desempeño de este trabajo. Este pin es la señal de referencia para la comparación de voltajes de los pines analógicos que no serán utilizados en este proyecto.



Componentes

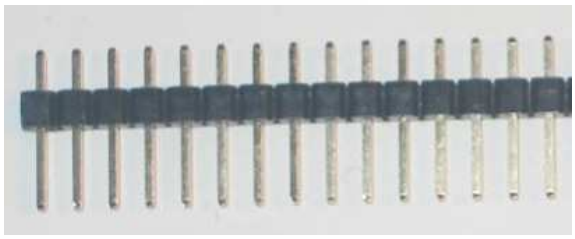
Micro controlador ATMega328P

Oscilador de cuarzo 16Mhz

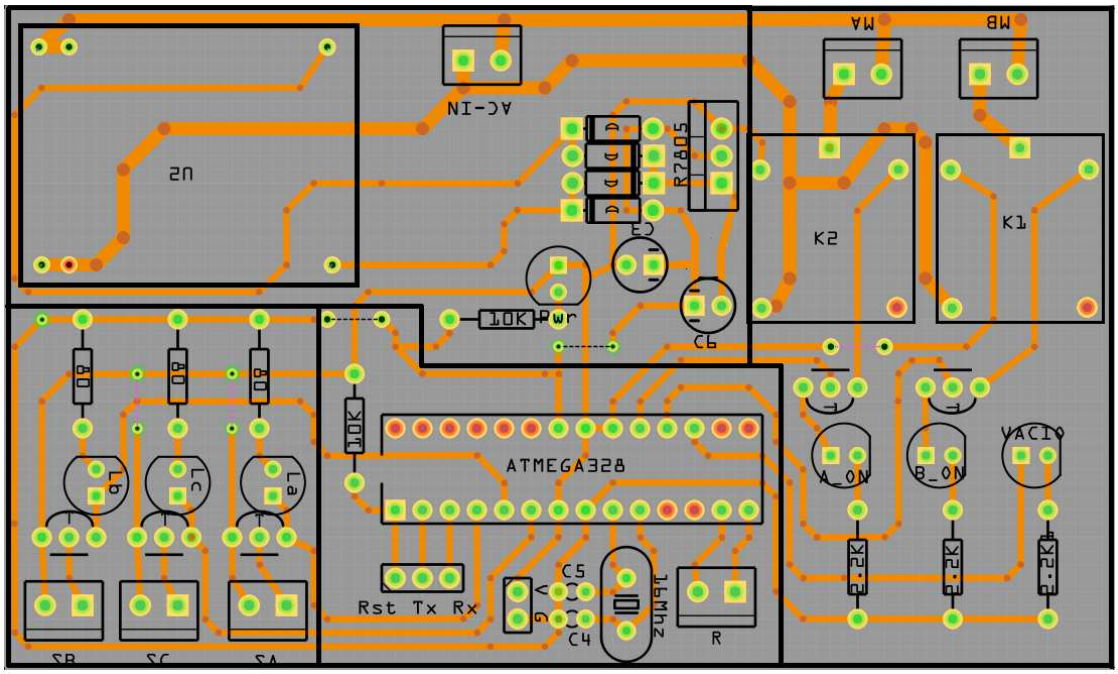
2x Condensador cerámico 22pF

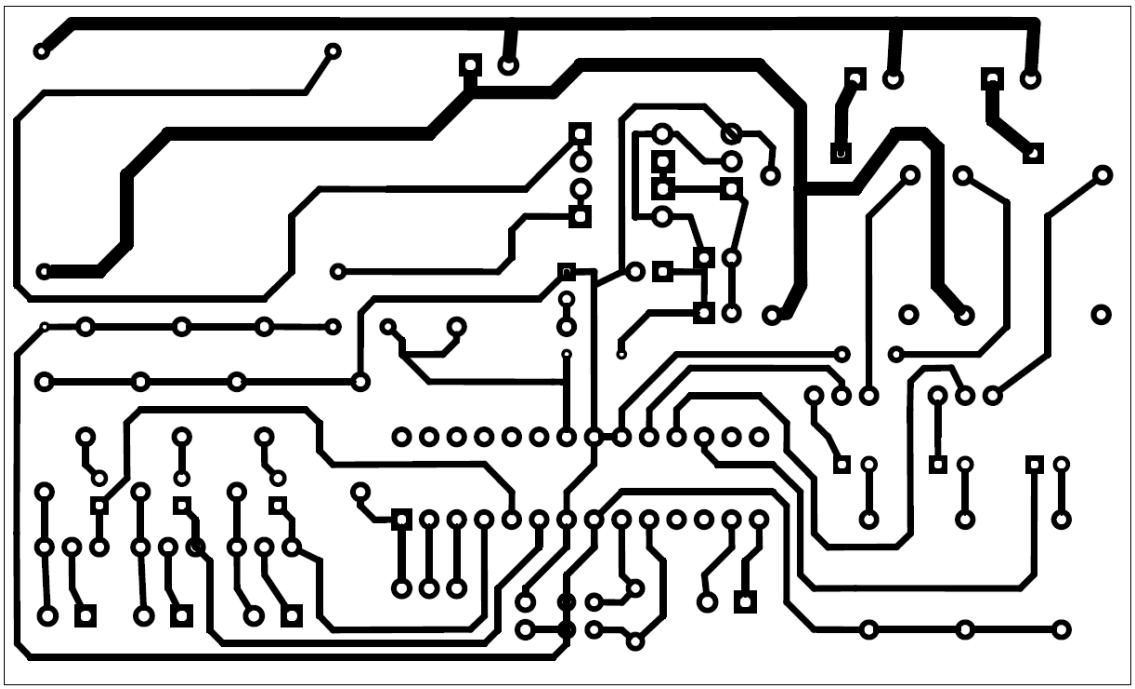
Resistencia 10Kohm

Pines de conexión (Macho)



Diseño final





Proceso de fabricación de la PCB

Materiales

Placa de cobre para PCB.

Existen diversas variantes, las más comunes son de "baquelita" y de "fibra de vidrio", siendo esta última la más adecuada para circuitos que van a sufrir inclemencias meteorológicas ya que son más resistentes.

Persulfato de Sodio

Podemos conseguir este compuesto químico en cualquier tienda de electrónica, es uno de los dos compuestos que formarán nuestro ácido.

Es un producto muy tóxico y debe ser tratado con mucha cautela.

También es posible utilizar cloruro férrico, el cual es mejor por lo rápidamente que actúa, pero cada vez más difícil de encontrar por su gran toxicidad.

Acido clorhídrico

Segundo compuesto de nuestro ácido, se puede encontrar en el salfumán domestico rebajado al 20%, idealmente debería estar en una concentración del 40%, pero aunque su actuación será más lenta, sirve para el propósito.

Alcohol

Etílico o isopropílico

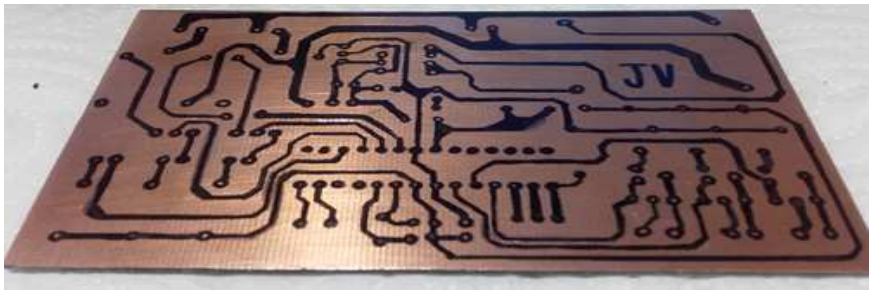
Rotulador permanente

El edding 3000 es el más utilizado, pero casi cualquier rotulador permanente será perfectamente válido.

Proceso de fabricación

1. Limpiar muy bien la placa de cobre con alcohol. Se debe eliminar cualquier residuo de las huellas de los dedos y/o partículas.

2. Dibujar el circuito con el rotulador permanente en la placa de cobre.

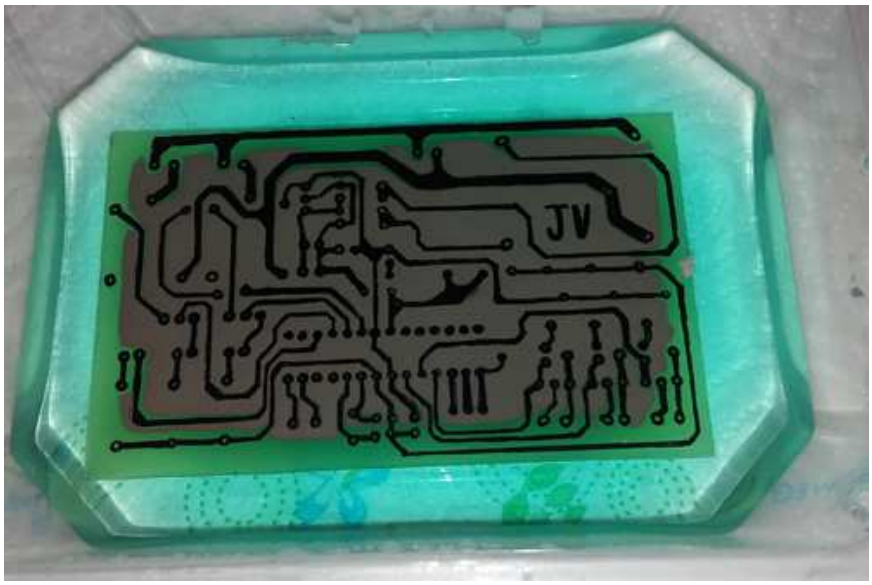


3. Preparación del ácido

- Imprescindible el uso de material de seguridad (GAFAS y GUANTES).
- Recuerde que el ácido produce vapores peligrosos, airee la habitación.

- 1 Vaso (250ml) de agua caliente en un recipiente PLÁSTICO, importante el uso de plástico, pues el ácido puede corroer otros materiales.
- 2 cucharadas grandes de Persulfato de sodio, remover hasta disolución.
- Acido clorhídrico 20%-40% (125ml).

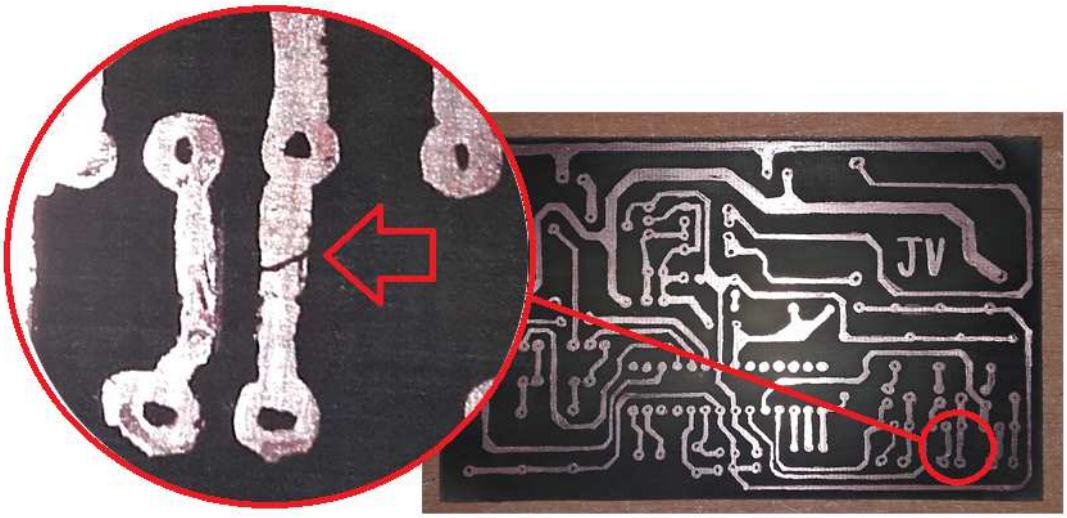
4. Sumergir la placa en el ácido y no dejar de moverla suavemente durante todo el proceso. Al cabo de unos minutos veremos como el ácido comienza a eliminar el cobre por los bordes de la placa.



5. Se debe poner atención a las pistas, especialmente a las más delgadas ya que el ácido puede corroer la tinta que la protege. Si esto sucede se debe sacar la placa, limpiarla en agua y volver a pintar las pistas que estén siendo corroídas antes de volver a sumergir la placa en el ácido.

6. Una vez que el cobre haya sido totalmente eliminado, retiraremos la placa del acido y la enjuagaremos en abundante agua. Después utilizaremos el alcohol para eliminar la tinta del rotulador.

7. Verificar las pistas usando un polímetro en modo de "continuidad" o "resistencia" para verificar que no hay cortes, en mi caso encontré una pequeña fisura que cortaba una pista y la inutilizaba. Esto se soluciona fácilmente con un poco de estaño y un soldador.



Soldadura de los componentes

Materiales necesario

Soldador de punta fina

(25W-40W)

Estaño para soldadura electrónica.

Existen muchas variantes del estaño incorporando diversas proporciones de estaño-plomo.

El estaño óptimo para soldadura electrónica tiene una proporción de estaño-plomo del 60/40 además de un núcleo de resina que facilita la soldadura.

Flux o pasta de soldadura.

Aunque no es totalmente necesario si es muy aconsejable para impregnar las superficies a soldar antes de aplicar el estaño.

Esto mejora la adherencia del estaño y facilita enormemente la soldadura.

Se puede apreciar en la imagen siguiente que no solo se han soldado los componentes, se han estañado al completo las pistas. Al ser el estaño un metal que se enlaza muy mal con el oxigeno, las pistas quedarán protegidas de la corrosión por oxidación.

Para estañar las pistas.

1. Precalentar la pista con el soldador.
2. Aplicar abundante flux o pasta de soldadura sobre la pista.
3. Aplicar un poco de estaño sobre la pista y esparcirla con la punta del soldador.

(Nota) Es sorprendente la poca cantidad de estaño necesaria para estañar una pista.

Para soldar los componentes:

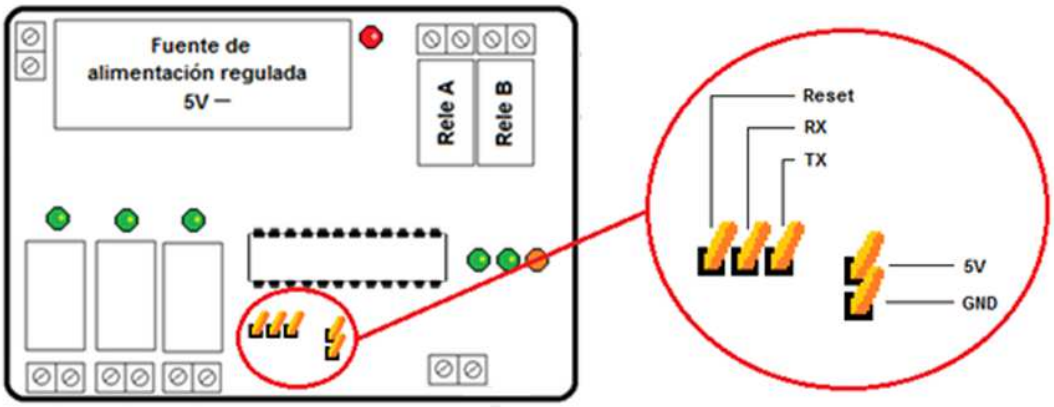
Perforar la placa con una broca del tamaño más adecuado al componente que se va a introducir, una broca de 0.8mm ó 1mm será la más adecuada a la mayoría de componentes. Estas brocas se pueden conseguir en tiendas e electrónica.

Se debe limitar al máximo el tiempo de contacto entre el soldador y el componente a no más de 2-3 segundos, para evitar dañarlo.

Resultado final

Programación del ATMega328P

Con intención de poder modificar el Firmware del ATMega328 después de instalado en la PCB se han dispuesto los siguientes pines en la placa.



Reset	Pone el ATMega en modo de programación
TX	Transmisión de datos serie
RX	Recepción de datos serie
5V	Alimentación externa durante el proceso de programación
GND	Tierra

Los terminales de 5V y GND proporcionan una salida de alimentación de 5V durante el funcionamiento normal del circuito cuando está conectado a 220V. Estos bornes pueden ser útiles para la instalación de un ventilador que refrigere el 7805 si fuese necesario, aunque la corriente que fluirá por él será inferior a los 300mA por lo que no debería ser necesario.

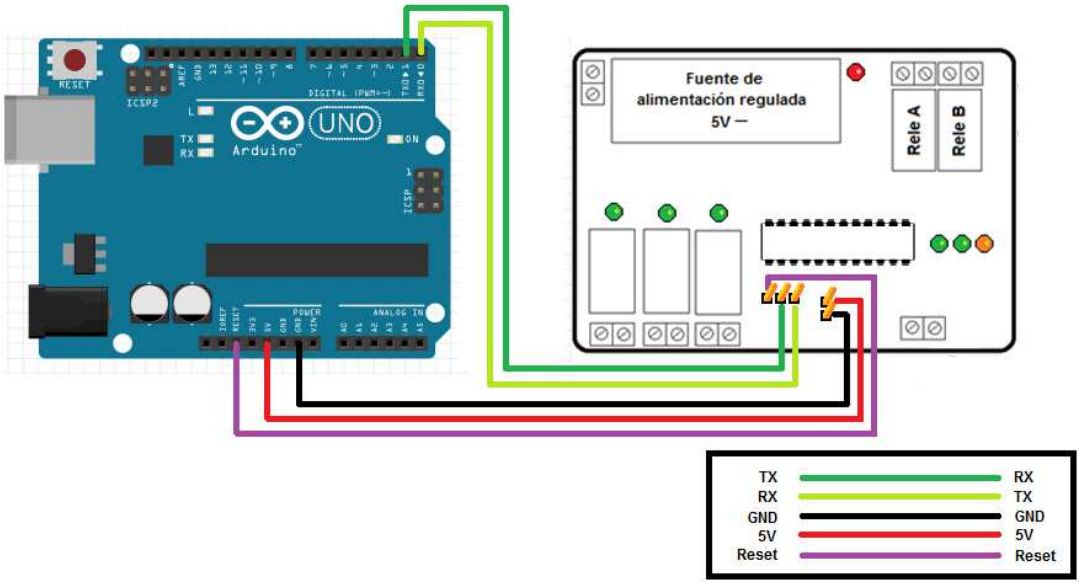
Para llevar a cabo la programación del ATMega nos serviremos de una placa "Arduino UNO" a la que **previamente se le ha retirado el micro controlador**. Esta tarea también se puede realizar con placas SMD aunque la configuración es bastante más compleja y no la contemplaré en este documento.

La placa Arduino cumple varias funciones.

- Alimentar el micro controlador con 5V durante el proceso de programación.
- Poner el micro controlador en modo de programación.
- Convertir la señal USB a una señal serial que el ATMega pueda comprender.

Conexión

La programación se realizará **con la placa desconectada de la toma de corriente**, esta recibirá la alimentación necesaria (5V) a través de una placa Arduino.



Se debe ser muy cauteloso en las conexiones, especialmente en la alimentación de 5V ya que la placa no incorpora ninguna protección en caso de conectarla al revés y el micro controlador podría resultar dañado.

Una vez conectada la placa con el Arduino, conectaremos este al ordenador por medio de un cable USB.

Todo el software necesario para realizar la carga del firmware se puede obtener gratuitamente en la URL. http://www.arduino.cc

Una característica añadida al firmware es la lectura "consistente" de los sensores de agua. Esto significa que cuando se realiza la lectura del estado de un sensor, no se hace en una única lectura, el micro controlador realizará hasta 10 lecturas seguidas y solo considerará la lectura como válida si todas las lecturas individuales coinciden. En caso de no ser válida, se considerará un posible fallo en el sensor y las bombas de agua se detendrán para evitar el funcionamiento en vacio.

Esta característica previene de posibles falsas lecturas producidas por interferencias externas.

Código fuente

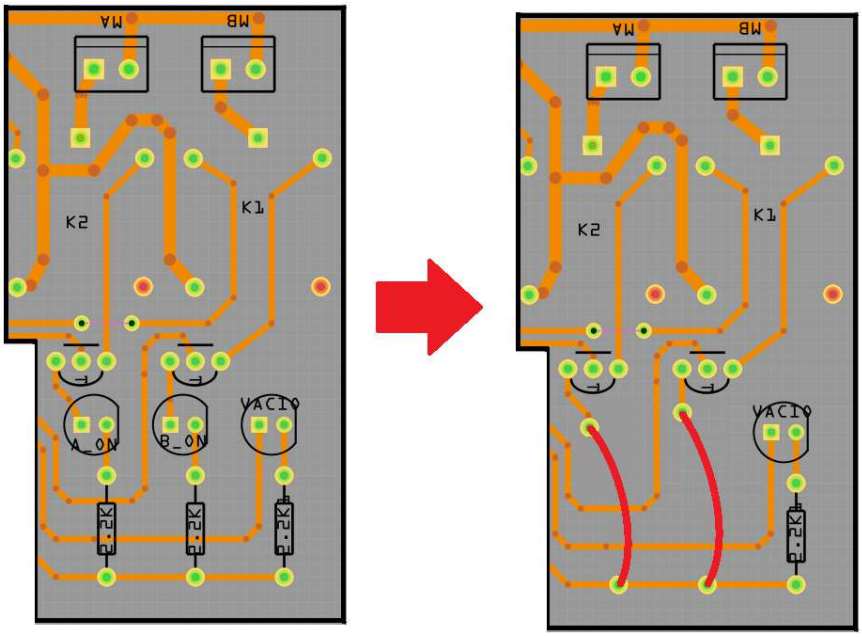
```
/*
  Control de bombas de agua
  Author Jose Vicente Lozano Copa
  Universidad de Alicante Marzo de 2016
*/

//Definición de los pins digitales para cada funcion
int tx = 0;
int rx = 1;
int sensorA = 2;
int sensorB = 3;
int sensorC = 4;
int bombaA = 13;
int bombaB = 12;
int ledVacio = 11;
int sensorR = 8;
int vrefR = 7;
boolean estadoError = false;

bool estadoLed = false;
bool estadoA = false;
bool estadoB = false;
bool llenando = false;
bool estadoBlink = true;


void setup() {
  /*Esto inicial de cada uno de los pines digitales*/
  pinMode(tx, OUTPUT);
  pinMode(rx, INPUT);
  pinMode(sensorA, INPUT);
  pinMode(sensorB, INPUT);
  pinMode(sensorC, INPUT);
  pinMode(bombaA, OUTPUT);
  pinMode(bombaB, OUTPUT);
  pinMode(ledVacio, OUTPUT);
  pinMode(sensorR, INPUT);
  pinMode(vrefR, OUTPUT);
  digitalWrite(vrefR, HIGH);
  Serial.begin(9600);
}

//Retorna True si despues de N lecturas, todas coinciden, el dato leído se retorna por referencia
boolean lectura(int sensor, int &dato){
  dato = digitalRead(sensor);
  for (int i=0;i<10;i++){
    int newDato = digitalRead(sensor);
    if (dato != newDato) {
       return false;
       Serial.println("Inconsistencia en el sensor!!");
    }
  }
  return true;
}
```

```
void loop() {
  delay(500);
  estadoError = false;
  int lecturaA = 0;
  int lecturaB = 0;
  int lecturaC = 0;
  int lecturaR = 0;

  //Lectura de los sensores
  //Si alguna de las lecturas no es consistente, me pongo en modo de ERROR
  if (!lectura(sensorA, lecturaA)) estadoError = true;
  if (!lectura(sensorB, lecturaB)) estadoError = true;
  if (!lectura(sensorC, lecturaC)) estadoError = true;
  if (!lectura(sensorR, lecturaR)) estadoError = true;

//Se activa el modo ERROR si se detecta agua en el sensor B pero no en el C, es imposible
if (lecturaB == 1 && lecturaC == 0) estadoError = true;

  if (estadoError){
    Serial.println("Estado inconsistente");
  }
  else
  {
    Serial.print("Sensor A = ");
    Serial.print(lecturaA + ",");
    Serial.print("Sensor B = ");
    Serial.print(lecturaB, ",");
    Serial.print("Sensor C = ");
    Serial.print(lecturaC, ",");
    Serial.print("Interruptor de riego = ");
    Serial.print(lecturaR);
    Serial.println("");
  }

    //Inicialización de los estados antes de verificaciones
      estadoA = false;  estadoB = false; estadoLed = false;

    /**Lógica Bomba A*/
      if (lecturaC == 0 && !llenando){//Al bajar el nivel de agua por debajo de B comienza el llenado
        llenando = true;
        Serial.println("Comienza el llenado!!");
      }
      else
      {
        if (llenando && lecturaB == 1){  //El llenado se detiene cuando la marca alcanza B
          llenando = false;
              Serial.println("Fin de llenado");
        }
      }

      if (llenando) Serial.println("Llenando!!");

      if (llenando){              //Si estamos en estado de llenado
        if (lecturaA == 1)        //Verifico el sensor de agua A, si hay agua activo la bomba
          estadoA = true;
        else {
          estadoA = false;        //Si no hay agua, paro la bomba y enciendo la alarma
          estadoLed = true;
        }
      }
      else
      {
        estadoA = false;        //Si no estamos en estado de llenado, la bomba A estara detenida
      }

    /**Lógica Bomba B*/
    //Si esta activo el interruptor de riego, activo la bomba B
      if (lecturaR==1)    estadoB = true;
      else                estadoB = false;

    //Si la bomba B esta activa pero el nivel de agua no alcanza el mínimo, detengo la bomba y
enciendo la alarma de vacio
      if (estadoB && lecturaC == 0){
          estadoB = false;
          estadoLed = true;
        }
```

```
//Escrituras digitales
    if (estadoError){

      //Paro las bombas
      digitalWrite(bombaA, LOW);
      digitalWrite(bombaB, LOW);

      //Y hago parpadear la luz

      if (estadoBlink == true){
      Serial.println("BLink!!! encendido");
        estadoBlink = false;
        digitalWrite(ledVacio, HIGH);
      }
      else
      {
      Serial.println("BLink!!! apagado");
        estadoBlink = true;
        digitalWrite(ledVacio, LOW);
      }
    }
    else{
      if (estadoLed){
        digitalWrite(ledVacio, HIGH);
        Serial.print("Alarma de Vacio = ON ");
      }
      else{
        digitalWrite(ledVacio, LOW);
        Serial.print("Alarma de vacio = OFF ");
      }

      if (estadoA){
        digitalWrite(bombaA, HIGH);
        Serial.print("BombaA ON ");
      }
      else{
        digitalWrite(bombaA, LOW);
        Serial.print("BombaA OFF ");
        }

      if (estadoB){
        digitalWrite(bombaB, HIGH);
        Serial.print("BombaB ON ");
        }
      else{
        digitalWrite(bombaB, LOW);
        Serial.print("BombaB OFF ");
        }
        Serial.println("");
    }
  }
```

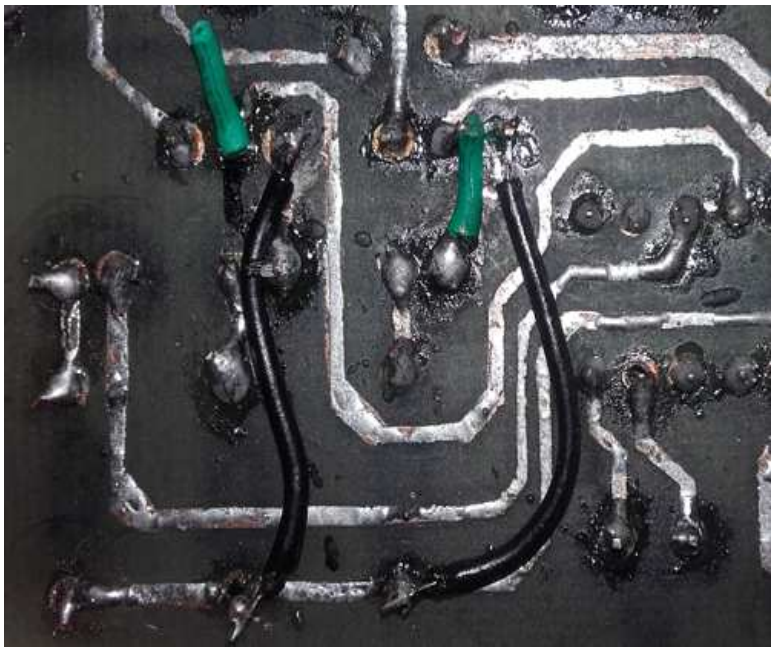
Errores de diseño detectados y solucionados a posteriori

Relés



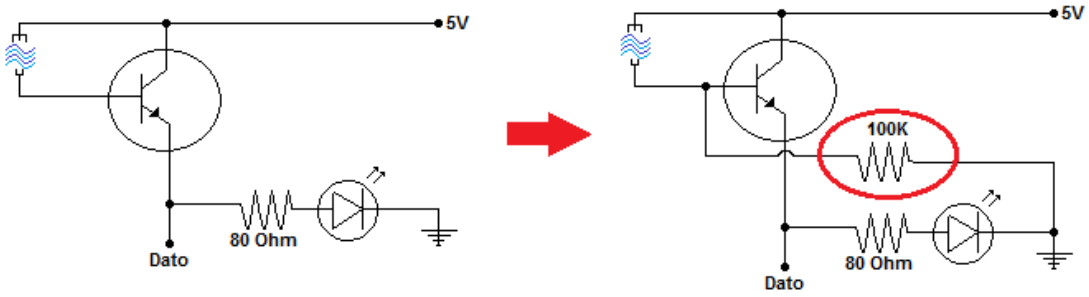
La caída de tensión que se produce en el diodo LED impide que el relé se active a pesar de estar abierto el transistor.

Este error se ha detectado durante las pruebas de funcionamiento con la placa ya producida por lo que la única forma de solventarlo sin tener que comenzar de nuevo ha sido anular el LED y la resistencia que lo protege.

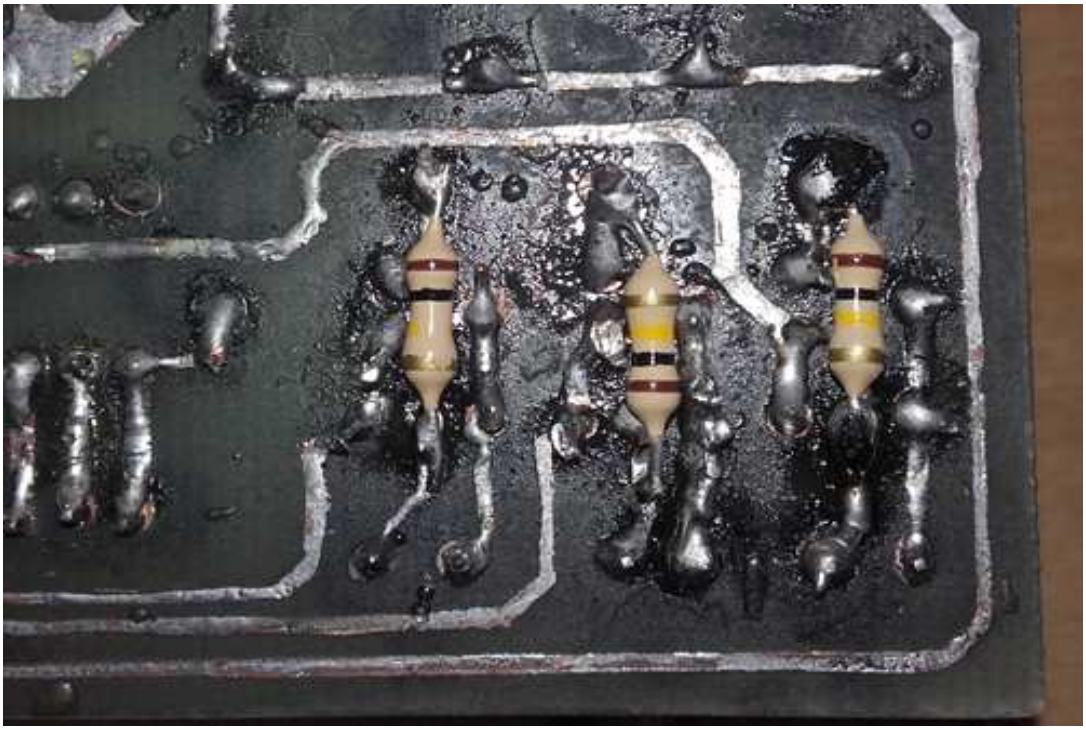


Sensor de presencia de agua

La ausencia de una conexión entre la base del transistor y tierra hacía que cualquier pequeña carga, incluso estática, activase el sensor. Por esta razón se ha introducido una resistencia de valor óhmico alto entre la base del transistor y tierra.



En la implementación final, se han colocado las nuevas resistencias en la cara inferior del circuito.



Pruebas de funcionamiento.

Leyenda

X = 1	- = 0	Blink = Parpadeo de 1 Segundo Indica un error en los sensores

Casos de prueba

Estado de los sensores					Comportamiento Deseado			
Sensor A	Sensor B	Sensor C	Sensor R	Llenando	Bomba A	Bomba B	Vacio	Ok?
-	-	-	-	False	-	-	-	Ok
-	-	-	X	False	-	-	X	Ok
-	-	X	-	False	-	-	-	Ok
-	-	X	X	False	-	X	-	Ok
-	X	-	-	False	-	-	Blink	Ok
-	X	-	X	False	-	-	Blink	Ok
-	X	X	-	False	-	-	-	Ok
-	X	X	X	False	-	X	-	Ok
X	-	-	-	False	X	-	-	Ok
X	-	-	X	False	X	-	X	Ok
X	-	X	-	False	-	-	-	Ok
X	-	X	X	False	-	X	-	Ok
X	X	-	-	False	-	-	Blink	Ok
X	X	-	X	False	-	-	Blink	Ok
X	X	X	-	False	-	-	-	Ok
X	X	X	X	False	-	X	-	Ok
-	-	-	-	True	-	-	X	Ok
-	-	-	X	True	-	-	X	Ok
-	-	X	-	True	-	-	X	Ok
-	-	X	X	True	-	X	X	Ok
-	X	-	-	True	-	-	Blink	Ok
-	X	-	X	True	-	-	Blink	Ok
-	X	X	-	True	-	-	-	Ok
-	X	X	X	True	-	X	X	Ok
X	-	-	-	True	X	-	-	Ok
X	-	-	X	True	X	-	X	Ok
X	-	X	-	True	X	-	-	Ok
X	-	X	X	True	X	X	-	Ok
X	X	-	-	True	-	-	Blink	Ok
X	X	-	X	True	-	-	Blink	Ok
X	X	X	-	True	X	-	-	Ok
X	X	X	X	True	X	X	-	Ok

Problemas en el uso de Fritzing

Por razones de la asignatura, para este proyecto se ha utilizado el software Fritzing para el diseño de la palca.

Fritzing es un software muy bonito y sencillo, pero tiene algunos fallos y limitaciones por los cuales no lo aconsejo para el diseño de placas PCB.

- No simula el comportamiento eléctrico.
- El condensador máximo es de 100uF y solo existe 1 diámetro disponible.
- No permite diseñar pistas de más de 4.8mm ni zonas libres.
- El auto ruteado se hace siempre a dos capas a pesar de estar la PCB configurada para una.
- Errores durante el guardado, de manera que al volver a abrir el proyecto las pistas se han descolocado de su posición.
- Librería de componentes muy limitada.

Bibliografía

Sobre el ATMega328P

Datasheet ATMega328p

http://pdf1.alldatasheet.com/datasheet-pdf/view/241077/ATMEL/ATMEGA328P.html

Conexión ATMega 328

https://www.youtube.com/watch?v=VLHg4UbP7MA

Programación del ATMega328P

http://www.arduino.cc

Información sobre Transistores

https://es.wikipedia.org/wiki/Transistor_de_uni%C3%B3n_bipolar

Quemado de placa PCB

https://www.youtube.com/watch?v=anadeSIFl_A

Fuente de alimentación

http://www.areatecnologia.com/electronica/fuente-alimentacion.html

https://www.youtube.com/watch?v=1D4_5KLFaug