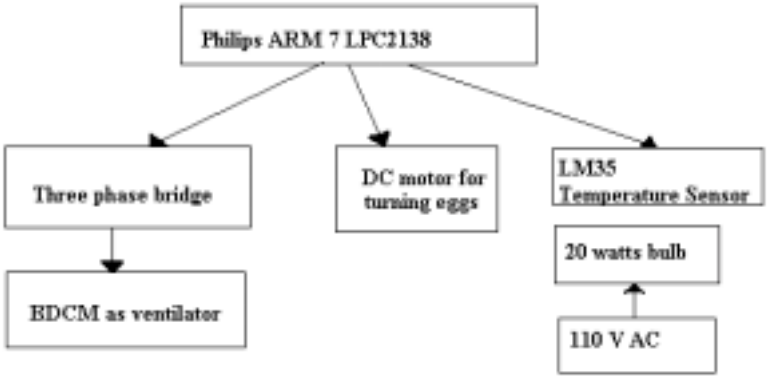
## Abstract

Having some eggs hatch is not a simple task. It requires controlling humidity, temperature and turning the eggs at least three or four times a day. The present project is about controlling the temperature and humidity of a six eggs incubator. For this, a brushless DC motor is controlled by the Philips ARM7 LPC2138, as shown in the next block diagram.



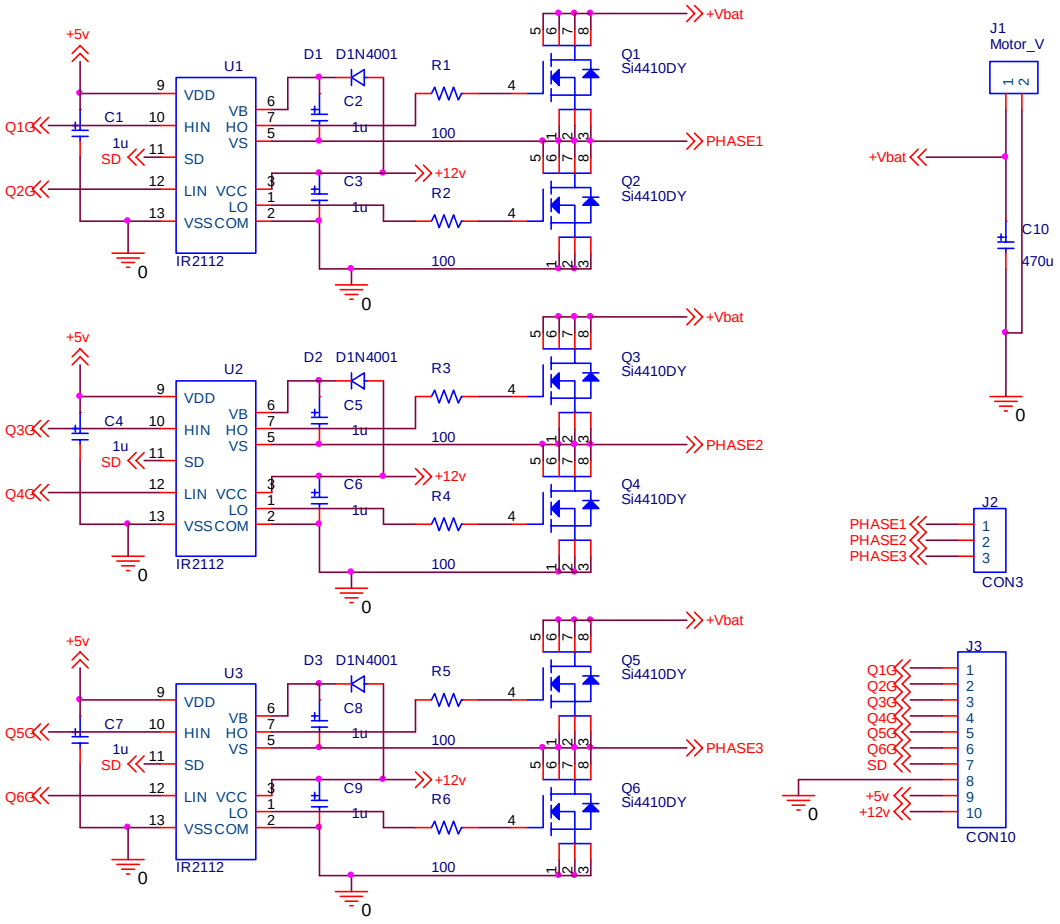

The main part of the project is to manage a sensored three phase Brushless DC motor. This motor has the advantages of being economical, no too heavy compared with a Permanent Magnet DC motor and also, it has a good torque.

The next program is for turning the BDCM and changing its speed with the rheostat that comes with the Philips ARM7 kit. The variable resistance produces a voltage that is measured and converted to a proportional duty cycle.

```
/*****************************************************************************/
/*                                                                           */
/*  MOTOR.C:  Motor control                                                  */
/*                                                                           */
/*****************************************************************************/

#include <stdio.h>                      /* standard I/O .h-file */
#include <LPC213x.H>                    /* LPC213x definitions  */

#define Q1 0x00000001
#define Q2 0x00000080
#define Q3 0x00000002
#define Q4 0x00000100
#define Q5 0x00200000
#define Q6 0x00000200

int main (void) {

  unsigned int m=0;
  unsigned int n;
  unsigned int val_adc;

  IODIR1  = 0x00FF0000;                     /* P1.16..23 defined as Outputs  */
  IODIR0  = 0x00200383;                     /* P0. defined as Inputs  */
  IOCLR0  = 0x00200383;
  AD0CR   = 0x00200402;                     /* Setup A/D: 10-bit AIN0 @ 3MHz */
  PINSEL0 = 0x00000000;
  PINSEL1 = 0x01000000;                     /* enable ADC */

  //PWM
  PWMMR0 = 1024;// period
  PWMMR1 = 600;
  PWMMR2 = 600;
  PWMMR3 = 600;
```

```
  PWMMR4 = 600;
  PWMMR5 = 600;
  PWMMR6 = 600;
  PWMPR  = 0000;// prescaler
  PWMMCR = 0x2; /* reset on MR0 */
  PWMPCR = 0x7E00;   /*0111 1110 0000 0000*/
  PWMLER = 0x7F;
  PWMTCR = 0x0b;
  PWMTCR = 0x09; /* enable PWM and start counting */


  while (1) {                                  /* Loop forever */
     n= IOPIN0 & 0x00000070;
     //IOSET1 = n<<12;
     if( m!= n ){
        m=n;
        IOCLR0  = 0x00200383;
              PINSEL0= 0x00000000;
        PINSEL1= 0x01000000;// clear all outputs

        AD0CR |= 0x01200000;                          /* Start A/D Conversion */
        do {
           val_adc = AD0DR;                           /* Read A/D Data Register */
        } while ((val_adc & 0x80000000) == 0);        /* Wait for end of A/D Conversion
*/
        AD0CR &= ~0x01000000;                         /* Stop A/D Conversion */
        val_adc = (val_adc >> 6) & 0x03FF;            /* Extract AIN0 Value */
              PWMMR1 = val_adc;
        PWMMR3 = val_adc;
        PWMMR5 = val_adc;
              PWMLER = 0x2A;  /* latch new values in match registers 1, 3, and 5 */
              IOCLR1 = 0x00FF0000;                     /* Turn off LEDs */
              IOSET1 = (val_adc << 12);
              /* based on the states of the sensors, select which wires must be powered
*/
           switch (m)  {

           case 0x00000010:
              PINSEL0= 0x00000002;    /* Q1 */
                        IOSET0 = Q6;
              break;

           case 0x00000020:
              PINSEL1= 0x01000400;    /* Q5 */
              IOSET0 = Q4;
              break;

           case 0x00000030:
              PINSEL0= 0x00000002;    /* Q1 */
                        IOSET0 = Q4;
              break;

           case 0x00000040:
              PINSEL0= 0x00000008;    /* Q3 */
                        IOSET0 = Q2;
              break;

           case 0x00000050:
              PINSEL0= 0x00000008;    /* Q3 */
                        IOSET0 = Q6;
              break;

           case 0x00000060:
              PINSEL1= 0x01000400;   /* Q5 */
                        IOSET0 = Q2;
              break;

           default:
           break;
        }
     }
  }
}
```

The next figure shows the schematic diagram of the power circuit. Three PWMs from the ARM 7 are used in the upper side of each inverter leg. They are enabled when needed based on the states of the Hall Effect sensors.




A picture of the project is presented next.