Expanding digital output pins on the Arduino using a shift register

Alex Zivanovic
March 2009

You may sometimes need more outputs than there are pins on the Arduino. One solution is to use a shift register. There is a good tutorial here: http://www.arduino.cc/en/Tutorial/ShiftOut.

You can use an alternative chip, the 4094, which is sometimes easier to get. It is available here:
http://www.rapidonline.com/Electronic-Components/Integrated-Circuits/Logic/4000-Series-CMOS-logic-family/77018/kw/hcf4094
order code: 83-0422, cost £0.46
Datasheet: http://www.rapidonline.com/netalogue/specs/83-0422.pdf

The following shows how you can use two of them to get 16 outputs using 3 digital pins on the Arduino.
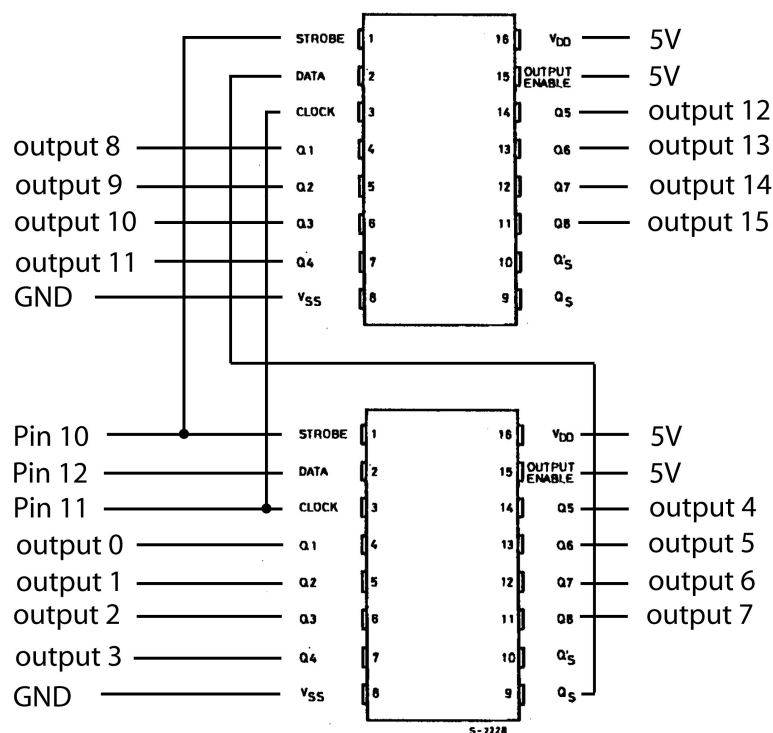
It uses pins 10,11,12 on the Arduino, and the outputs are labelled 0-15. Can you see that pin 9 of the lower chip is the output data stream that is fed to the input data pin on the upper chip? For data to get to the upper chip, it is shifted through the lower chip. This is called daisy-chaining. You can keep adding chips in this way, still only using three Arduino pins.

The program on the next page can be used to test this circuit. It is best to attach an LED to each output (with an appropriate resistor) to see what is happening.

```
// program to test using two 4094 shift registers
//
// if everything is correct, it will light up one LED at a time
//

int latchPin = 10;
int clockPin = 11;
int dataPin = 12;

void setup() {
  pinMode(latchPin, OUTPUT);
  pinMode(clockPin, OUTPUT);
  pinMode(dataPin, OUTPUT);
  digitalWrite(latchPin, 0);       //make sure data isn't latched
}

void loop() {

  unsigned int outputpattern =1;    //stores the output pattern as an unsigned int, that is, two bytes
  int pattern_LSB;              //the least significant byte (LSB) of the pattern
  int pattern_MSB;               //the most significant byte (MSB) of the pattern

  for (int i=0 ; i< 16; i++){
    pattern_MSB = outputpattern >>8;         //extract the MSB of the pattern by shifting all the bits over by 8
    pattern_LSB = outputpattern & B11111111;   //extract the LSB of the pattern by bitwise AND


    shiftOut(dataPin, clockPin, MSBFIRST, (byte) pattern_MSB);
    shiftOut(dataPin, clockPin, MSBFIRST, (byte) pattern_LSB);
    digitalWrite(latchPin, 1);            // flick the latch to put the data on the output pins
    delay(1);
    digitalWrite(latchPin, 0);
    delay(100);
    outputpattern = outputpattern <<1;      // shift the outputpattern left by one bit
  }
}
```