



ADC với LM35

Bởi:

DKS Group

Lý thuyết

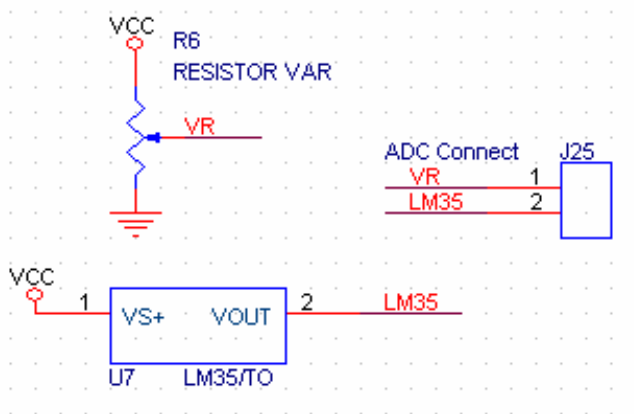
Đối với ATMEGA 16L: 8 chân của PORTA sử dụng làm 8 kênh đầu vào ADC. Để sử dụng tính năng ADC của Atmega 16L chúng ta cần phải thiết kế phần cứng của Vi điều khiển như sau :

* Chân AVCC chân này bình thường khi thiết kế mạch chúng ta đưa lên Vcc(5V) nhưng khi trong mạch có sử dụng các kênh ADC của phần cứng thì chúng ta phải nối chân này lên Vcc qua 1 cuộn cảm nhằm mục đích cấp nguồn ổn định cho các kênh (đầu vào) của bộ biến đổi.

* Chân AREF chân này cần cấp 1 giá trị điện áp ổn định được sử dụng làm điện áp tham chiếu, chính vì vậy điện áp cấp vào chân này cần ổn định vì khi nó thay đổi làm giá trị ADC ở các kênh thu được bị trôi (thay đổi) không ổn định với 1 giá trị đầu vào chúng ta có công thức tính như sau: $ADCx = (V_INT * 1024) / AREF$ chỉ dựa vào công thức chúng ta cũng có thể thấy giá trị ADCx tỉ lệ thuận với điện áp vào V_INT. Giá trị ADC thu được từ các kênh được lưu vào 2 thanh ghi ADCH và ADCL khi sử dụng chúng ta phải đọc giá trị từ các thanh ghi này, khi sử dụng ở chế độ 8 bit thì chỉ lưu vào thanh ghi ADCL.

Mô tả

Đầu ra của LM35 và chân 2 biến trở 1K trên Kit được nối với 2 jump chờ. Với AMEGA16L có 8 kênh ADC là chức năng thứ 2 của PORTA. Do đó để ADC ta dùng dây nối 2 chân đó với 2 bit của PORTA là bit 0 và bit 1..

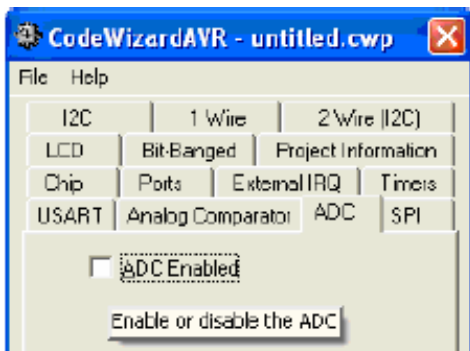


Theo datasheet LM35 thì cứ 10mV tương ứng với 10C, ở 00C điện áp ra là 0V, tương ứng với giá trị ADC là 0. Với $V_{ref}=5V$, giá trị của ADC từ 0 đến 256, lấy tròn 250 mức. Mỗi giá trị ADC ứng với $5V/250=20\text{ mV}$. Vậy 1 giá trị ADC ứng với 20C. Muốn tăng độ phân giải ADC ta giảm V_{ref} .

Thực hành

Các bước khởi tạo code như sau:

Trong tab ADC check vào ADC enable:



Ta check vào Use 8 bit, để ADC trả về giá trị 8 bit, và ta ADC dùng ngắt check vào Interrupt, về điện áp tham khảo AREF thì lấy điện áp của chân AREF của AVR được nối với 5V. Tần số ADC tùy các bạn thích nhanh hoặc chậm chọn giá trị phù hợp. Trong box Automatically Scan Inputs các bạn check vào Enabled. Vì chúng ta cần ADC 2 kênh, 1 kênh dùng biến trở để test ADC, một kênh từ LM35 đấu với 2 bit 0 và 1 của PORTA do đó chọn First 0, Last 1.

Khởi tạo cho LCD vào PORTB như hình bên cạnh.

Chọn Generate, Save and Exit.

Đề hiển thị được một số bất kỳ lên LCD, trong thư viện hàm không có và ta phải tự viết hàm. Đầu vào là một biến unsigned char, ta phải tách lấy hàng trăm, hàng chục, hàng đơn vị và đưa lần lượt lên LCD.

Code như sau:

```
void lcd_putnum(unsigned char so,unsigned char x,unsigned
char y)

{

unsigned char a,b,c;

a=so/100;

// lay fan tram

b=(so-100*a)/10;

// lay fan chuc

c=(so-100*a-10*b);

// lay hang don vi

lcd_gotoxy(x,y);

// ve vi tri x,y

lcd_putchar(a+48);

// day ra hang tram, ma ascii

lcd_putchar(b+48);

// day ra hang chuc, ma ascii

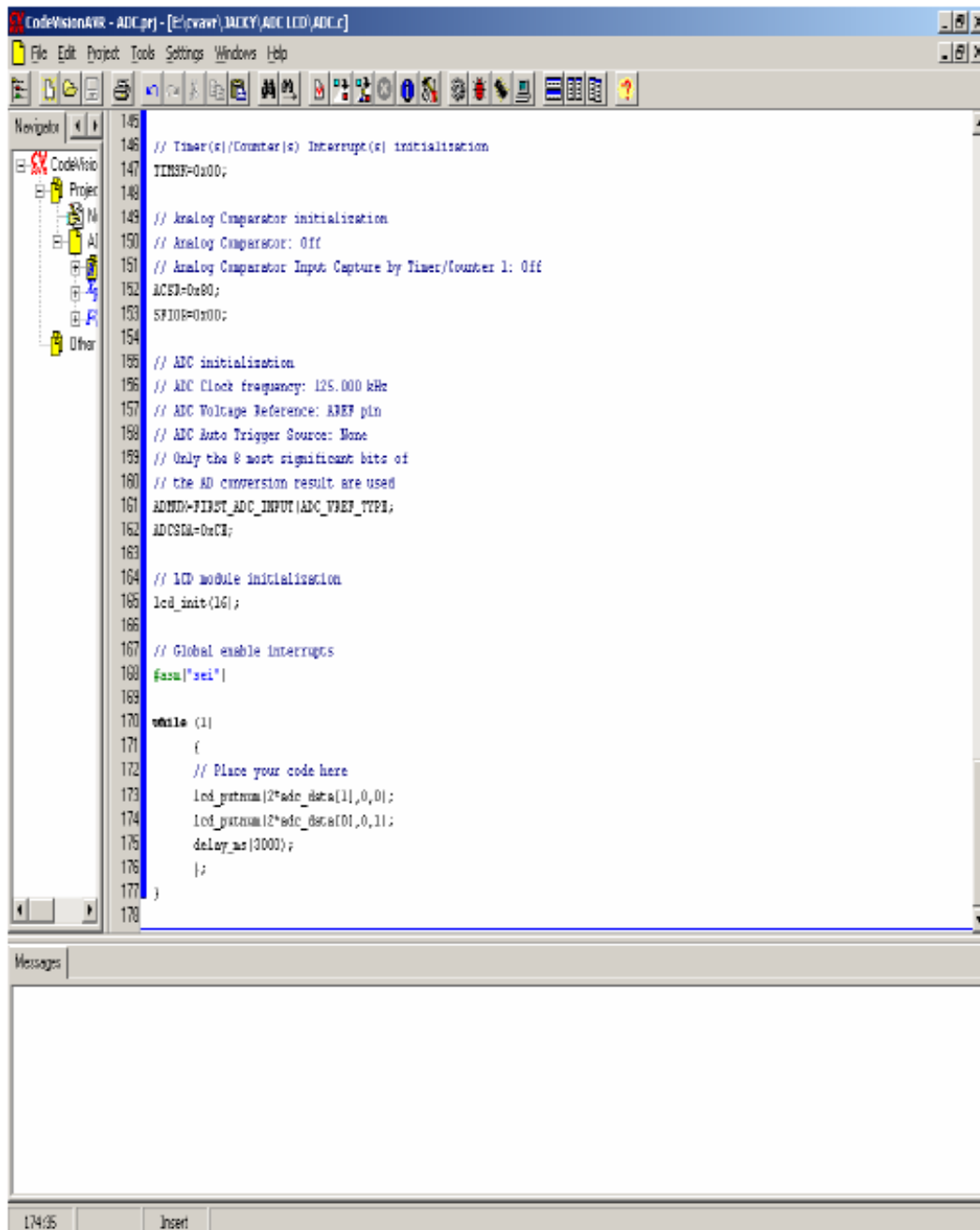
lcd_putchar(c+48);

// day ra hang don vi, ma ascii

}
```

Trong vòng while(1) trong hàm main ta viết như sau:

```
while (1) { // Place your code here
  lcd_putnum(2*adc_data[1],0,0); // đưa giá trị ADC từ
  LM35*2= nhiệt độ lcd_putnum(adc_data[0],0,1); // đưa giá
  trị ADC từ biến trở delay_ms(3000); // trễ 3 s, cập nhật
  dữ liệu một lần };
```



Đo nhiệt độ bằng LM35 qua ADC thường có sai số và độ trôi, do đó ta cần hiệu chỉnh nhiệt độ bằng cách so sánh với nhiệt kế.