MINI51 HC-SR04 Sensor Control

Example Code Introduction for 32-bit NuMicro®Family

Information

Application	This sample code uses Mini51 to control the HC-SR04 sensor
BSP Version	Mini51DE_Series_BSP_CMSIS_v3.02.000
Hardware	NuTiny-EVB-Mini51

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

1 Function Description

1.1 Introduction

This procedure is the ranging procedure of HC-SR04 ultrasonic module.

1.2 Principle

Processing:

a. Power on for system initialization and timer initialization.

b. When the timer external capture port has rising edge and falling edge change, MCU will generate external capture interrupt, save the timer value during interrupt, and set a detection flag after detecting the falling edge.

c. The main program judges the mark and calculates the time difference between the two captures.According to the HC-SR04 module, this time difference is the time of the back and forth transmission from ultrasonic wave to object.

d. distance calculation.Distance = transmission time *340/2.

e. Print the calculated distance through the serial port.If the detection range of the module is beyond 2cm~4m, the serial port will print the over-distance information.

1.3 Demo Result

```
Mini51 + HC_SR04 Sample Code...
Now distance is 97 mm
Now distance is 99 mm
Now distance is 95 mm
Now distance is 99 mm
Now distance is 104 mm
Now distance is 104 mm
Now distance is 109 mm
Now distance is 117 mm
Now distance is 113 mm
Now distance is 113 mm
Now distance is 118 mm
Now distance is 118 mm
Now distance is 118 mm
```

2 Code Description

The timer initialization：

```
/* Give a dummy target frequency here. Will over write capture resolution with macro */
TIMER_Open(TIMER0, TIMER_PERIODIC_MODE, 1000000);

/* Update prescale to set proper resolution.
   Timer 0 clock source is 12MHz, to set resolution to 1ms, we need to
   set clock divider to 12. e.g. set prescale to 12 - 1 = 11 */
TIMER_SET_PRESCALE_VALUE(TIMER0, 11);

/* Set compare value as large as possible, so don't need to worry about counter overrun too frequently */
TIMER_SET_CMP_VALUE(TIMER0, 0xFFFFFF);

/* Configure Timer 0 free counting mode, capture TDR value on rising edge */
TIMER_EnableCapture(TIMER0, TIMER_CAPTURE_FREE_COUNTING_MODE, TIMER_CAPTURE_FALLING_THEN_RISING_EDGE);

/* Start Timer 0 */
TIMER_Start(TIMER0);

/* Enable timer interrupt */
TIMER_EnableCaptureInt(TIMER0);
NVIC_EnableIRQ(TMR0_IRQn);
```

The MCU generates an external capture interrupt, during which the timer value is saved:

```
void TMR0_IRQHandler(void)
{
     /* clear timer over flag */
     TIMER_ClearIntFlag(TIMER0);
     TIMER_ClearCaptureIntFlag(TIMER0);
     /* high level */
     if((P3->PIN)&(1<<2))
     {
          t0 = TIMER_GetCaptureData(TIMER0);
     }
     else
     {
          t1 = TIMER_GetCaptureData(TIMER0);
          g_u8DetFlag=1;
```

```
        }
    }
```

The main program calculates the time difference between the two catches.According to the description of HC-SR04 module, this time difference is the time of the back and forth transmission from ultrasonic wave to the object, and the calculated distance is printed through the serial port:

```
    while (1)
    {
        P1->DOUT |= (1 << 0);
        CLK_SysTickDelay(100000);

        if (g_u8DetFlag)
        {
            g_u8DetFlag = 0;

            if (t1 >= t0)
            {
                Detected_dis = (t1 - t0);

                if (Detected_dis < 120)
                {
                    printf("Your distance is too close, must over than 2cm\n");
                }
                else if (Detected_dis > 23600)
                {
                    printf("Your distance is too farway ,not in my reach. The distance must less
than 4m\n");
                }
                else
                {
                    Detected_dis *= 17;
                    Detected_dis /= 100;
                    printf(" Now distance is %d mm\n", Detected_dis);
                }
            }

            P1->DOUT &= ~(1 << 0);
            CLK_SysTickDelay(100);
        }
    }
```

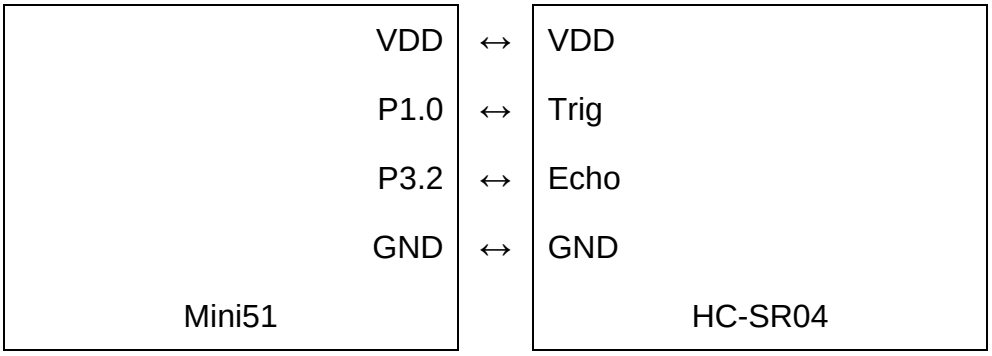
3 Software and Hardware Environment

Software Environment

- BSP version
  - Mini51DE_Series_BSP_CMSIS_v3.02.000
- IDE version
  - Keil uVersion5.25

Hardware Environment

- Circuit components
  - NuTiny-EVB-Mini51
  - HC-SR04
- Diagram



4 Directory Information

- EC_MINI51_Control_HC-SR04_Sensor_V1.00
  - Library — Sample code header and source files
    - CMSIS — Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
    - Device — CMSIS compliant device header file
    - StdDriver — All peripheral driver header and source files
  - SampleCode
    - ExampleCode — Source file of example code

5 How to Execute Example Code

1. Browsing into sample code folder by Directory Information(section 4) and double click Control_HC-SR04_Sensor.uvproj.
2. Enter Keil compile mode
   a. Build
   b. Download
   c. Start/Stop debug session
3. Enter debug mode
   a. Run

6 Revision History

Date	Revision	Description
Jul.24, 2019	1.00	1. Initially issued.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.