

Programming and Control

Blog Archive

- 2012 (2)
- 2011 (1)
- ▼ 2010 (4)
 - June (1)
 - May (1)
 - ▼ April (2)
 - [Simple Keyboard](#)
 - [Simulation Using](#)
 - [SendInput functio...](#)
 - [LPC17xx GPIO basic](#)

MONDAY, APRIL 5, 2010

LPC17xx GPIO basic

Depend on LPC17xx version the pinouts may be different. I am using 100-pin LPC1768 as an example.

Reference document: [LPC17xx User manual](#)

Require #include header [lpc17xx.h](#)

Pins on LPC1768 are divided into 5 groups (PORT) starting from 0 to 4.

Pin naming convention: Po.o (group o, pin o) or (Port o, Pin o)

Note: if I refer to a page number, that page number is from LPC17xx User manual

Each pin has 4 operating modes: GPIO(default), 1st alternate function, 2nd alternate function, 3rd alternate function.

All GPIO pins are powered automatically so we do not need to turn on the pin power supply.

1. Pin function setting

The register LPC_PINCON control operating mode of these pins.

LPC_PINCON->PINSEL0[1:0] control PIN 0.0 operating mode (page 106/835)

...

LPC_PINCON->PINSEL0[31:30] control PIN 0.15 operating mode

LPC_PINCON->PINSEL1[1:0] control PIN 0.16 operating mode

...

LPC_PINCON->PINSEL1[29:28] control PIN 0.30 operating mode

LPC_PINCON->PINSEL2[1:0] control PIN 1.0 operating mode

...

LPC_PINCON->PINSEL2[31:30] control PIN 1.15 operating mode

LPC_PINCON->PINSEL3[1:0] control PIN 1.16 operating mode

...

LPC_PINCON->PINSEL3[31:30] control PIN 1.31 operating mode

...

LPC_PINCON->PINSEL9[25:24] control PIN 4.28 operating mode

LPC_PINCON->PINSEL9[27:26] control PIN 4.29 operating mode

Note: Some register bits are reserved and are not used to control a pin. For example,

LPC_PINCON->PINSEL9[23:0] are reserved

LPC_PINCON->PINSEL9[31:28] are reserved

Bit Values Function

00 GPIO function

01 1st alternate function

10 2nd alternate function

11 3rd alternate function

Example:

To set pin 0.3 as GPIO (set corresponding bit to 00)

LPC_PINCON->PINSEL0 &= ~(1 << 7) | (1 << 6);

To set pin 0.3 as ADC channel 0.6 (2nd alternate function, set corresponding bit to 10)

LPC_PINCON->PINSEL0 |= ((1 << 7) | (0 << 6)); // you may omit (0 << 6)

2. Pin direction setting

Register LPC_GPIOn->FIODIR[31:0] control the pin input/output where n is the pin group (0-4)

To set a pin as output, set the corresponding bit to 1. To set a pin as input, set the corresponding bit to 0. By default, all pins are set as input (all bits are 0).

Example:

To set pin 0.3 as output
`LPC_GPIO0->FIODIR |= (1<<3);`

3. A pin is set as output

Pin digital high/low setting

`LPC_GPIO0->FIOSET` is used to turn a pin to HIGH (page 122/835)

Register `LPC_GPIO0->FIOCLR` is used to turn a pin to LOW

To turn a pin to digital 1 (high), set the corresponding bit of `LPC_GPIO0->FIOSET` to 1. To turn a pin to digital 0 (low), set the corresponding bit of `LPC_GPIO0->FIOCLR` to 1.

Example:

Turn pin 0.3 to high

`LPC_GPIO0->FIOSET |= (1<<3);`

If we set `LPC_GPIO0->FIOSET` bit to 0, there is no effect

Turn pin 0.3 to low

`LPC_GPIO0->FIOCLR |= (1<<3);`

If we set `LPC_GPIO0->FIOCLR` bit to 0, there is no effect

4. A pin is set to input

a. Read a pin value

Register `LPC_GPIO0->FIOPIN` stores the current pin state. (page 125/835)

The corresponding bit is 1 indicates that the pin is driven high

Example:

To read current pin 0.3 state

`value = ((LPC_GPIO0->FIOPIN & (1 <<3))>> 3);`

Note:

Write a 1/0 to a corresponding bit in `LPC_GPIO0->FIOPIN` can change the output of the pin to 1/0 but it is not recommended. We should use `LPC_GPIO0->FIOSET` and `LPC_GPIO0->FIOCLR` instead.

b. Pin internal pull up setting

Register `LPC_PINCON->PINMODEn` is used to set up a pin internal pull up

`LPC_PINCON->PINMODE0[1:0]` control Po.0 internal pull up

...

`LPC_PINCON->PINMODE0[31:30]` control Po.15 internal pull up

Please see `LPC_PINCON->PINSELn` for the full list or page 110/835

Bit values Pin mode

00 on-chip pull-up resistor enabled

01 repeater mode, see page 103/835

10 tri-state mode, neither pull-up nor pull-down resistor enabled

11 on-chip pull-down resistor enabled

Example

By default all pins which are set as input has internal pull-up on (00)

To disable internal pull-up on pin 0.3

`LPC_PINCON->PINSEL0 |= (1 << 7);`

5. Using GPIO interrupt, only pin group 0 and 2 are used with pin interrupt

a. Falling edge interrupt

Register `LPC_GPIOINT->IOxIntEnF` is used to enable falling edge detected interrupt, in which x is the group number, either 0 or 2. To turn on falling edge interrupt of a pin, set the corresponding pin to 1.

Example

To enable pin 0.3 falling edge interrupt

`LPC_GPIOINT->IO0IntEnF |= (1 << 3);`

b. Raising edge interrupt

Register LPC_GPIOINT->IOxIntEnR is used to enable raising edge detected interrupt, in which x is the group number. To turn on raising edge interrupt of a pin, set the corresponding pin to 1.

Example

To enable pin 0.3 and 0.5 raising edge interrupt
 LPC_GPIOINT->IOoIntEnR |= ((1 << 3) | (1 << 5));

c. All GPIO interrupts are connected to EINT3 interrupt source. You need to turn EINT3_IRQn on in order to use GPIO interrupt.
 NVIC_EnableIRQ(EINT3_IRQn);

d. Interrupt handler subroutine

Add this subroutine to your code. Every time the GPIO interrupts are fired (regardless of which pin), this subroutine is called.

```
void EINT3_IRQHandler (void)
{
}
}
```

Register LPC_GPIOINT->IOxIntStatF[31:0] has the status of which pin falling edge interrupt was fired.
 Register LPC_GPIOINT->IOxIntStatR[31:0] has the status of which pin raising edge interrupt was fired.
 Register LPC_GPIOINT->IOxIntClr[31:0] is used to clear the status of a pin interrupt. To clear the status bit, write 1 to the corresponding bit.

Example

To check if pin 0.3 interrupt was fired or pin 0.5 interrupt was fired
 void EINT3_IRQHandler (void)

```
{
  if ((LPC_GPIOINT->IOoIntStatR & (1 << 3)) == (1 << 3))
  {
    //raising edge interrupt on pin 0.3 was fired
    LPC_GPIOINT->IOoIntClr |= (1 << 3); // clear the status
    //do your task

    return;
  }

  if ((LPC_GPIOINT->IOoIntStatR & (1 << 5)) == (1 << 5))
  {
    //raising edge interrupt on pin 0.5 was fired
    LPC_GPIOINT->IOoIntClr |= (1 << 5); // clear the status
    //do your task

    return;
  }
}
```

Posted by [Anh Ngo Viet Nguyen](#) at [5:10 PM](#)

7 comments:



Daniel [January 30, 2013 at 5:14 AM](#)

I'm having a little bit of trouble following your guide. Do you think you could take a look at what I have and briefly comment is there's something not quite right? The source is in the post here:
<http://mbed.org/questions/333/GPIO-Interrupt-Troubles/>
 Thanks so much for this guide!

[Reply](#)

PercySugden [May 17, 2013 at 6:30 AM](#)



Thanks very much ! I am new to the LPC17xx and I couldn't see how to handle multiple gpio inputs. They don't exactly make it obvious and I was going round in circles until I read your blog. Now I am flying - Thanks again - James.

[Reply](#)



shashank rinke [July 23, 2013 at 6:10 AM](#)

Thank you for explaining things especially interrupt handling it helped a lot!!!!

[Reply](#)



Artacart [September 11, 2013 at 11:09 AM](#)

nice...)

[Reply](#)



sprabubeece [July 8, 2014 at 12:44 AM](#)

i m new to lpc1768,so i cant understantwhat you told here,,can u explain briefly through email and any other eBooks,then sample program with explanation it is very useful for my future,,,,,,,,please help me,,,,,

[Reply](#)



Green Giant [September 4, 2014 at 11:05 PM](#)

Seriously good blog post. Explains the basics better than the user manual. Thanx!

[Reply](#)



Unknown [August 3, 2015 at 7:48 AM](#)

Hi,
I am trying to compile a program on LpcXpresso for LPC1769.
I have included "chip.h" but it seems that I am missing those definitions : LPC_PINCON, LPC_SC
file : chip_lpc175X_6x.h does not includes those definitions...
Could you please give me a suggestion ?

[Reply](#)

Enter your comment...

Comment as:

Fabio Rocha (▼)

[Sign out](#)

[Publish](#)

[Preview](#)

☐ Notify me

[Newer Post](#)

[Home](#)

Subscribe to: [Post Comments \(Atom\)](#)

About Me

[ANH NGO VIET NGUYEN](#)

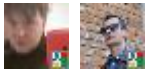
[View my complete profile](#)

Followers

Join this site

with Google Friend Connect

Members (2)



Already a member? [Sign in](#)

