ARM Processor – Programming

Lecture on ARM7 – LPC2148 - Interfacing programs in C

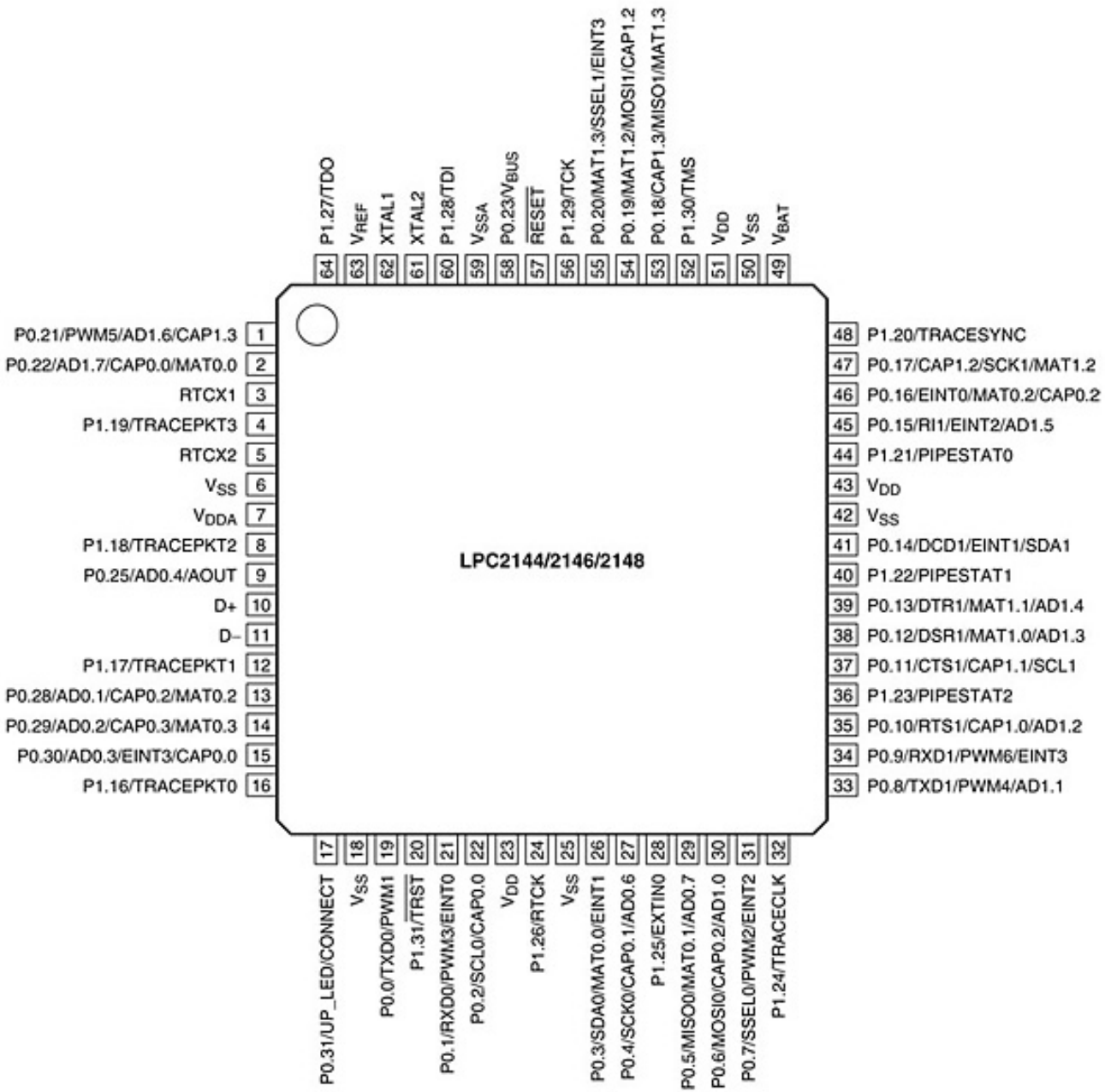
By
Harish V. Mekali

Features of LPC2148 – LQFP64

- **ARM7TDMI processor**
- **40KB on RAM And 512KB Flash memory**
- **Up to 45 GPIO pins from two ports => P0[0:25] + P0[28:31] + P1[16:31]**
- **Operating voltage : 3.3v**
- **Crystal frequency : upto 60MHz**
- **14 analog inputs – with two 10 bit ADCs**
- **1 ten bit DAC**
- **2 timers / counters**
- **6 PWM**
- **RTC and watchdog timers**
- **9 level / edge sensitive external interrupt pins**
- **Protocols : 2 UART, 2 I2C, SPI, SSP, USB 2.0**
- **Power saving modes : Ideal and Power down**

Applications :

- Point of Sale
- Ticketing machine
- Access control
- Industry control
- Medical instruments
- Communication gateway / router
- Mobile phones / tablets
- Automobile applications
- General purpose applications


LPC2144/2146/2148
1 P0.21/PWM5/AD1.6/CAP1.3
2 P0.22/AD1.7/CAP0.0/MAT0.0
3 RTCX1
4 P1.19/TRACEPKT3
5 RTCX2
6 VSS
7 VDDA
8 P1.18/TRACEPKT2
9 P0.25/AD0.4/AOUT
10 D+
11 D−
12 P1.17/TRACEPKT1
13 P0.28/AD0.1/CAP0.2/MAT0.2
14 P0.29/AD0.2/CAP0.3/MAT0.3
15 P0.30/AD0.3/EINT3/CAP0.0
16 P1.16/TRACEPKT0
17 P0.31/UP_LED/CONNECT
18 VSS
19 P0.0/TXD0/PWM1
20 P1.31/TRST
21 P0.1/RXD0/PWM3/EINT0
22 P0.2/SCL0/CAP0.0
23 VDD
24 P1.26/RTCK
25 VSS
26 P0.3/SDA0/MAT0.0/EINT1
27 P0.4/SCK0/CAP0.1/AD0.6
28 P1.25/EXTIN0
29 P0.5/MISO0/MAT0.1/AD0.7
30 P0.6/MOSI0/CAP0.2/AD1.0
31 P0.7/SSEL0/PWM2/EINT2
32 P1.24/TRACECLK
33 P0.8/TXD1/PWM4/AD1.1
34 P0.9/RXD1/PWM6/EINT3
35 P0.10/RTS1/CAP1.0/AD1.2
36 P1.23/PIPESTAT2
37 P0.11/CTS1/CAP1.1/SCL1
38 P0.12/DSR1/MAT1.0/AD1.3
39 P0.13/DTR1/MAT1.1/AD1.4
40 P1.22/PIPESTAT1
41 P0.14/DCD1/EINT1/SDA1
42 VSS
43 VDD
44 P1.21/PIPESTAT0
45 P0.15/RI1/EINT2/AD1.5
46 P0.16/EINT0/MAT0.2/CAP0.2
47 P0.17/CAP1.2/SCK1/MAT1.2
48 P1.20/TRACESYNC
49 VBAT
50 VSS
51 VDD
52 P1.30/TMS
53 P0.18/CAP1.3/MISO1/MAT1.3
54 P0.19/MAT1.2/MOSI1/CAP1.2
55 P0.20/MAT1.3/SSEL1/EINT3
56 P1.29/TCK
57 RESET
58 P0.23/VBUS
59 VSSA
60 P1.28/TDI
61 XTAL2
62 XTAL1
63 VREF
64 P1.27/TDO

LPC2148 pin mapping on ALS board

SL No.	PROGRAM NAME	PORT LINE
1	LCD	P0.2 - P0.7
2	7SEG DISPLAY	P0.16 - P0.23 & P0.28 - P0.31
3	GP LED'S	P0.16 – P0.23
4	SWITCHES	P1.16 – P1.23
5	STEPPER MOTOR	P0.12 – P0.15
6	DC MOTOR	P0.8 & P0.11
7	RELAY	P0.10
8	BUZZER	P0.9
9	EXT-INTERRUPT0	P1.25
10	DAC0800	P0.16 – P0.23
11	PWM	P0.8
12	Internal ADC	P0.16
13	4X4 KEY MATRIX	P1.16 – P1.23
14	EXT-INTERRUPT1	P0.3

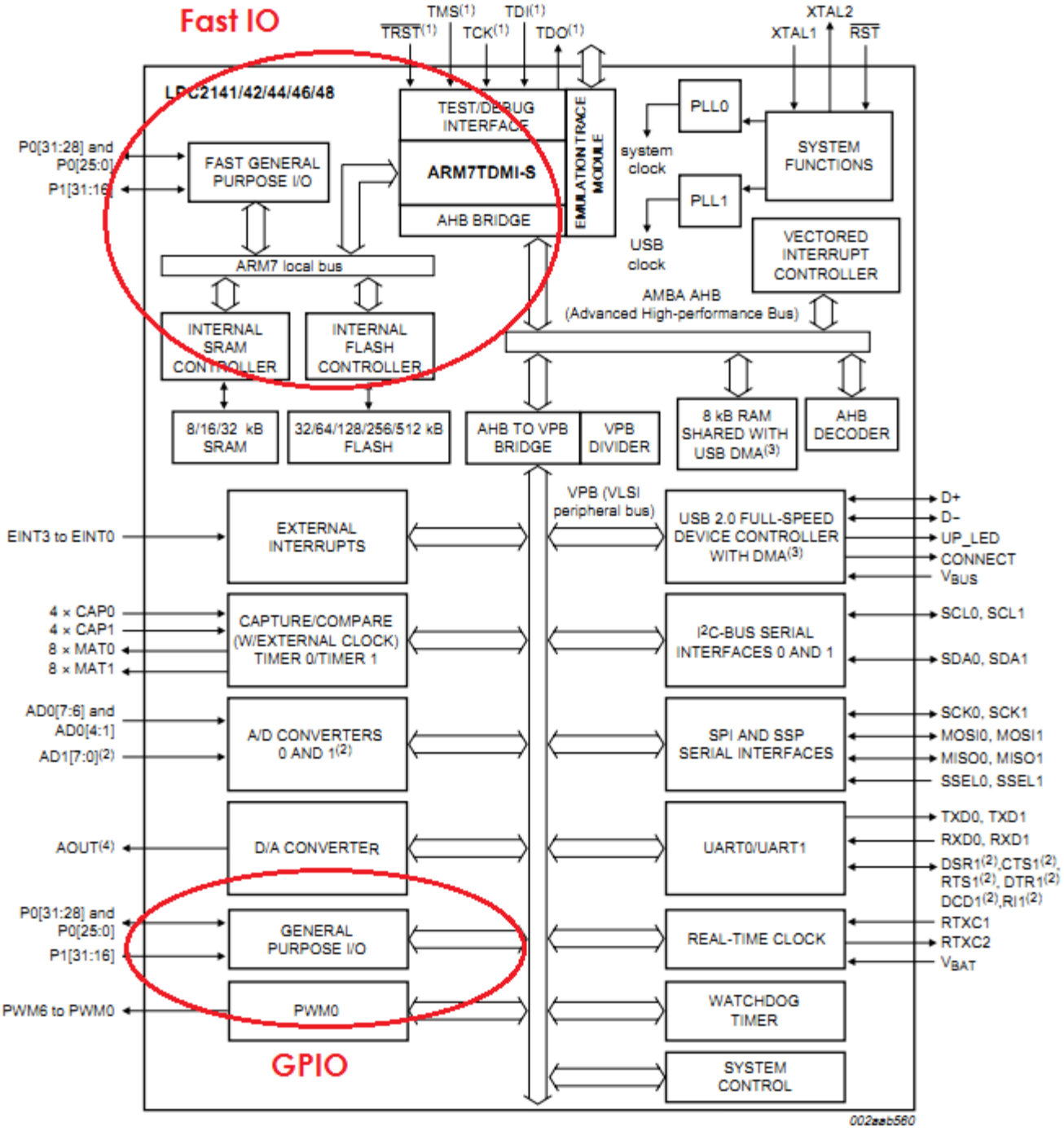
GPIO – General Purpose Input Output

LED

Buzzer

Rely with DC motor

Switch


Fast IO
TMS(1)
TDI(1)
TRST(1)
TCK(1)
TDO(1)
XTAL2
XTAL1
RST
LPC2141/42/44/46/48
TEST/DEBUG INTERFACE
ARM7TDMI-S
AHB BRIDGE
EMULATION TRACE MODULE
PLL0
PLL1
system clock
USB clock
SYSTEM FUNCTIONS
VECTORED INTERRUPT CONTROLLER
P0[31:28] and P0[25:0]
P1[31:16]
FAST GENERAL PURPOSE I/O
ARM7 local bus
INTERNAL SRAM CONTROLLER
INTERNAL FLASH CONTROLLER
AMBA AHB (Advanced High-performance Bus)
8/16/32 kB SRAM
32/64/128/256/512 kB FLASH
AHB TO VPB BRIDGE
VPB DIVIDER
8 kB RAM SHARED WITH USB DMA(3)
AHB DECODER
VPB (VLSI peripheral bus)
EINT3 to EINT0
EXTERNAL INTERRUPTS
USB 2.0 FULL-SPEED DEVICE CONTROLLER WITH DMA(3)
D+
D−
UP_LED
CONNECT
VBUS
4 × CAP0
4 × CAP1
8 × MAT0
8 × MAT1
CAPTURE/COMPARE (W/EXTERNAL CLOCK) TIMER 0/TIMER 1
I2C-BUS SERIAL INTERFACES 0 AND 1
SCL0, SCL1
SDA0, SDA1
AD0[7:6] and AD0[4:1]
AD1[7:0](2)
A/D CONVERTERS 0 AND 1(2)
SPI AND SSP SERIAL INTERFACES
SCK0, SCK1
MOSI0, MOSI1
MISO0, MISO1
SSEL0, SSEL1
AOUT(4)
D/A CONVERTER
UART0/UART1
TXD0, TXD1
RXD0, RXD1
DSR1(2), CTS1(2), RTS1(2), DTR1(2) DCD1(2), RI1(2)
P0[31:28] and P0[25:0]
P1[31:16]
GENERAL PURPOSE I/O
REAL-TIME CLOCK
RTXC1
RTXC2
VBAT
PWM6 to PWM0
PWM0
WATCHDOG TIMER
GPIO
SYSTEM CONTROL
002aab560

GPIO(General Purpose Input Output) Pins

Each port is associated with 4 register

1. Pin function selection as every pin in the controller is multiplexed

- **PINSEL register => PINSEL0 for P0[0:15]**
  PINSEL1 for P0[16:31]
  PINSEL2 for P1[16:31]

i.e., for each pin in port, two PINSEL bits are dedicated as there can be 4 functions of each pin

2. Pin direction selection (Input or Output)

- **IODIR register => IO0DIR for PO & IO1DIR for P1**

3. Pin value selection (Low or High)

- **SET register (High value) => IO0SET for P0 & IO1SET for P1**
- **Clear register (Low value) => IO0CLR for P0 & IO1CLR for P1**

Ex :

```
IO0DIR = 0x00000080;        // Pin PORT0.7 configured as output
IO0SET = 0x00000080;        //Pin PORT0.7 goes HIGH
IO0CLR = 0x00000080;        //Pin PORT0.7 goes LOW
```

GPIO - Blink an LED

```
#include <LPC21XX.h>

unsigned int delay;

int main ()
{
        PINSEL0 = 0x00000000 ;          // Configure P0.0 to P0.15 as GPIO
        PINSEL1 = 0x00000000 ;          // Configure P0.16 to P0.31 as GPIO
        IO0DIR   = 0x00010000 ;         // Configure P0.16 as Output

        while(1)
        {
        IO0CLR = 0x00010000;            // CLEAR (0) P0.16, LED OFF
        for(delay=0; delay<500000; delay++);        // delay
        for(delay=0; delay<500000; delay++);        // delay

        IO0SET =  0x00010000;           // SET (1) P0.16, LED ON
        for(delay=0; delay<500000; delay++);        // delay
        for(delay=0; delay<500000; delay++);        // delay
        }

}
```

GPIO - Buzzer

```
#include <LPC21XX.h>

unsigned int delay;

int main ()
{
        PINSEL0 = 0x00000000 ;          // Configure P0.0 to P0.15 as GPIO
        PINSEL1 = 0x00000000 ;          // Configure P0.16 to P0.31 as GPIO
        IO0DIR  = 0x00000200 ;          // Configure P0.9 as Output

        while(1)
        {
        IO0CLR = 0x00000200;            // CLEAR (0) P0.9, Buzzer OFF
        for(delay=0; delay<500000; delay++);        // delay
        for(delay=0; delay<500000; delay++);        // delay

        IO0SET =  0x00000200;           // SET (1) P0.9, Buzzer ON
        for(delay=0; delay<500000; delay++);        // delay
        for(delay=0; delay<500000; delay++);        // delay
        }

}
```

GPIO - Relay

```
#include <LPC21XX.h>

unsigned int delay;

int main ()
{
        PINSEL0 = 0x00000000 ;          // Configure P0.0 to P0.15 as GPIO
        PINSEL1 = 0x00000000 ;          // Configure P0.16 to P0.31 as GPIO
        IO0DIR   = 0x00000400 ;         // Configure P0.10 as Output

        while(1)
        {
        IO0CLR = 0x00000400;            // CLEAR (0) P0.10, Relay OFF
        for(delay=0; delay<500000; delay++);        // delay
        for(delay=0; delay<500000; delay++);        // delay

        IO0SET =  0x00000400;           // SET (1) P0.10, Relay ON
        for(delay=0; delay<500000; delay++);        // delay
        for(delay=0; delay<500000; delay++);        // delay
        }

}
```

GPIO - Switch

```
#include <LPC21XX.h>

unsigned int delay;

#define SW1                 0x00800000                                  // P1.23

int main ()
{
            PINSEL1 = 0x00000000 ;              // Configure P0.16 to P0.31 as GPIO
            PINSEL2 = 0x00000000 ;
            IO1DIR  =  0x00000000 ;             // Configure P1.0 to P1.31 as input
            IO0DIR  =  0x00FF0000 ;              // Configure P0.16 to P0.31 as output

            while(1)
            {
            if(!(IOPIN1 & SW1))
            {IO0CLR = 0x00000200;               // CLEAR (0) P0.10 to P0.13 and P0.18 to P0.21, LEDs ON
            for(delay=0; delay<500000; delay++);            // delay
            for(delay=0; delay<500000; delay++);            // delay

            IO0SET =  0x00000200;               // SET (1) P0.10 to P0.13 and P0.18 to P0.21, LEDs OFF
            for(delay=0; delay<500000; delay++);            // delay
            for(delay=0; delay<500000; delay++);            // delay
            }}

}
```

GPIO – Relay with DC motor

```
#include<lpc214x.h>
void clock_wise(void);
void anti_clock_wise(void);
unsigned int j=0;
int main()
{
 IO0DIR= 0X00000900;
 IO0SET= 0X00000100; // P0.8 should always high.
 while(1)
 {
  clock_wise();
  for(j=0;j<400000;j++); // delay
  anti_clock_wise();
  for(j=0;j<400000;j++); // delay
 } // End of while(1)
} // End of Main

void clock_wise(void)
{
 IO0SET = 0X00000900;//Selecting the P0.11 line for clockwise
}
void anti_clock_wise(void)
{
 IO0CLR = 0X00000900;
 IO0SET = 0X00000100;//Selecting the P0.11 line for Anti clockwise
}
```

GPIO – Stepper motor

```
#include <LPC21xx.H>
 void clock_wise(void);
 void anti_clock_wise(void);
 unsigned long int var1,var2;
 unsigned int i=0,j=0,k=0;
 int main(void)
 {
            PINSEL0 = 0x00FFFFFF;              //P0.12 to P0.15 GPIo
            IO0DIR |= 0x0000F000;              //P0.12 to P0.15 output
            while(1)
            {
                  for(j=0;j<50;j++)      // 20 times in Clock wise Rotation
                        clock_wise();

                  for(k=0;k<65000;k++);   // Delay to show  anti_clock Rotation

                  for(j=0;j<50;j++)      // 20 times in  Anti Clock wise Rotation
                        anti_clock_wise();

                  for(k=0;k<65000;k++);   // Delay to show clock Rotation

            }                                                          // End of while(1)

 }                                                                     // End of main
```

GPIO – Stepper motor

```
void clock_wise(void)
{
        var1 = 0x00000800;              //For Clockwise
        for(i=0;i<=3;i++)               // for A B C D Stepping
        {
                var1 = var1<<1;         //For Clockwise
                IO0PIN = ~var1;
                for(k=0;k<3000;k++);    //for step speed variation
        }
}

void anti_clock_wise(void)
{
        var1 = 0x00010000;      //For Anticlockwise
        for(i=0;i<=3;i++)       // for A B C D Stepping
        {
                var1 = var1>>1;         //For Anticlockwise
                IO0PIN = ~var1;
                for(k=0;k<3000;k++);    //for step speed variation
        }
}
```