LPC-E2124

Get Started Guide

Revision 1.0 28/03/2005

LPC-E2124 Board

Introduction

The **LPC2124** are based on a 16/32 bit ARM7TDMI-S™ CPU with real-time emulation and embedded trace support, together with 128/256 kilobytes (kB) of embedded high speed flash memory. A 128-bit wide memory interface and a unique accelerator architecture enable 32-bit code execution at maximum clock rate. For critical code size applications, the alternative 16-bit Thumb Mode reduces code by more than 30% with minimal performance penalty.

With their compact 64 pin package, low power consumption, various 32-bit timers, 4-channel 10-bit ADC, PWM channels and 46 GPIO lines with up to 9 external interrupt pins these microcontrollers are particularly suitable for industrial control, medical systems, access control and point-of-sale. With a wide range of serial communications interfaces, they are also very well suited for communication gateways, protocol converters and embedded soft modems as well as many other general-purpose applications.

The **LPC-E2124** Development board is designed to evaluate LPC2124 processor. It has the following features:

- CS8900 Ethernet interface
- two general purpose buttons
- potentiometer connected to analog input 0
- Dallas i-button interface
- 24LC515 EEPROM for external web storage
- standard JTAG connector with ARM 2x10 pin layout for programming/debugging with ARM-JTAG
- two on board voltage regulators 1.8V and 3.3V with up to 800mA current
- single power supply: +5VDC required
- three LAN status LEDs, three general purpose status LEDs
- power supply filtering capacitor
- USB to RS232 interface
- RESET circuit with external control of Philips ISP utility via USB/RS232
- RESET button
- DBG jumper for JTAG enable
- BSL jumper for bootloader enable
- JRST jumper for enable/disable external RESET control by RS232
- 14.7456 Mhz crystal
- extension header for all uC ports
- PCB: FR-4, 1.5 mm (0,062"), green soldermask, white silkscreen component print
- Dimensions: 80x90 mm (3.15x3.55")

The purpose of this guide is to describe LPC-P2124 Development board.

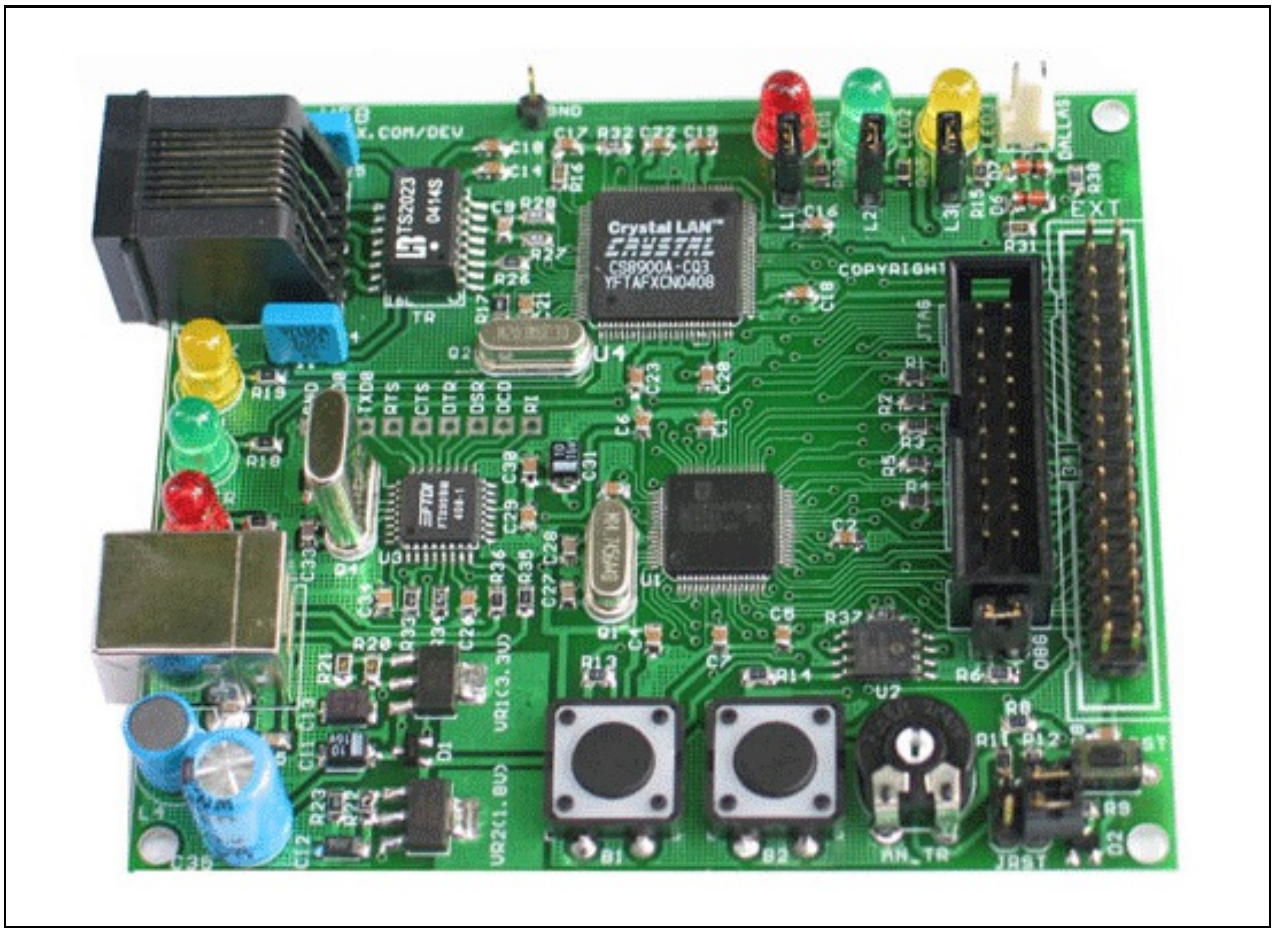
Board LPC-E2124
Hardware details Describes the hardware peripherials in detail
Programming describes how to write programs for the E2124 Board.

Revision 01.01.2005 Creating

LPC-E2124 Board

Picture

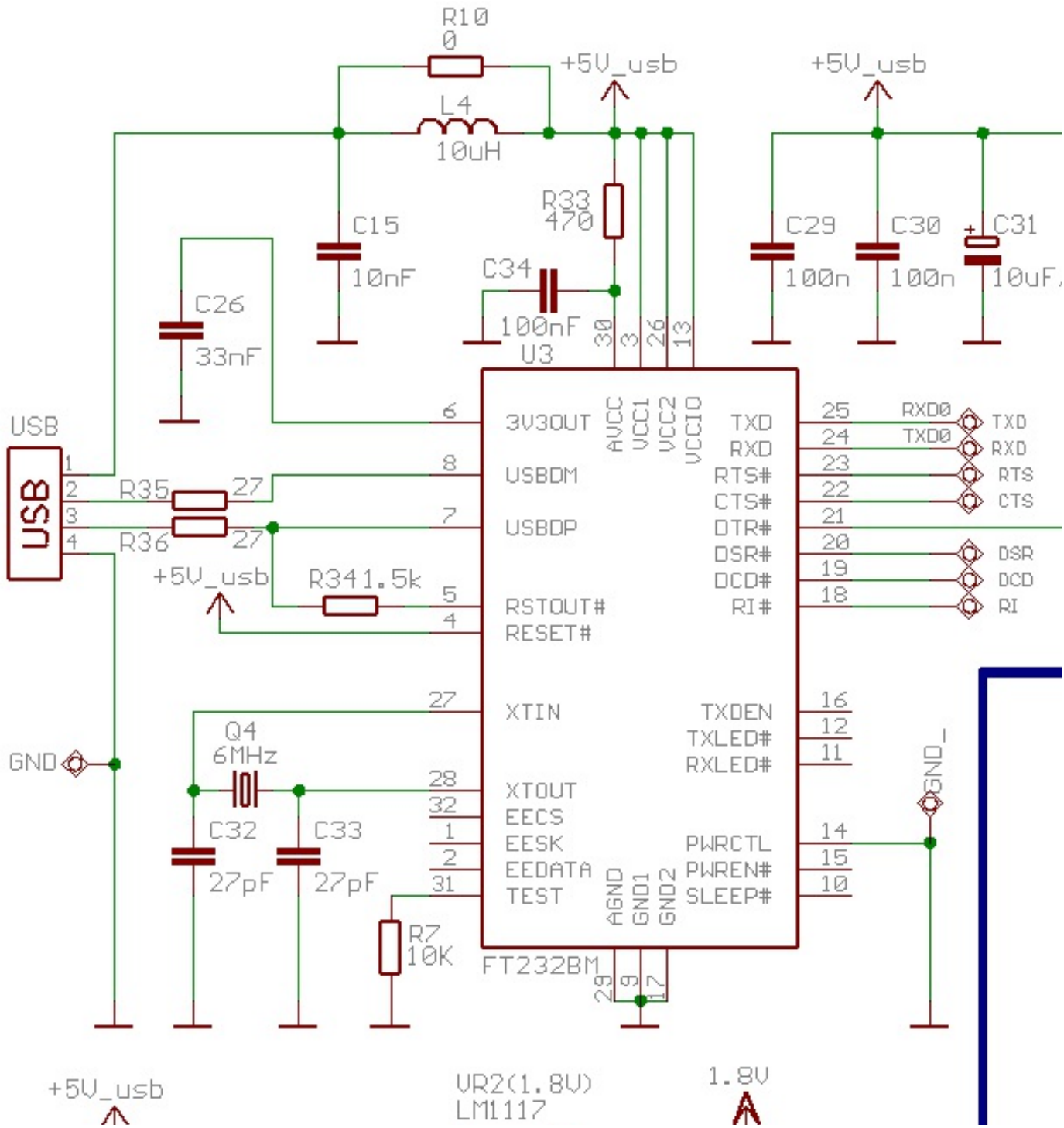
This is picture of LPC-E2124 Development board.

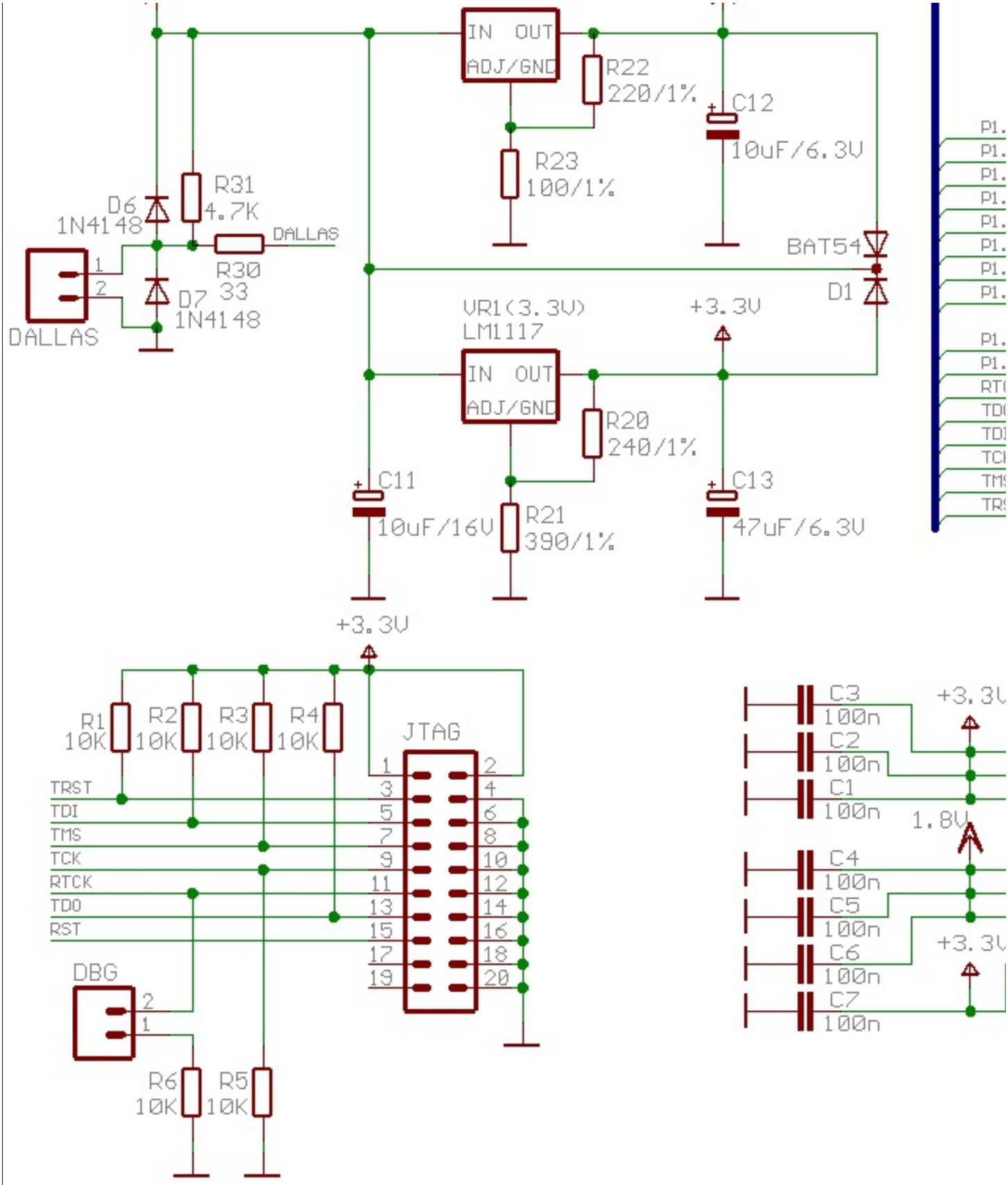


Board LPC-E2124

LPC-E2124 Board

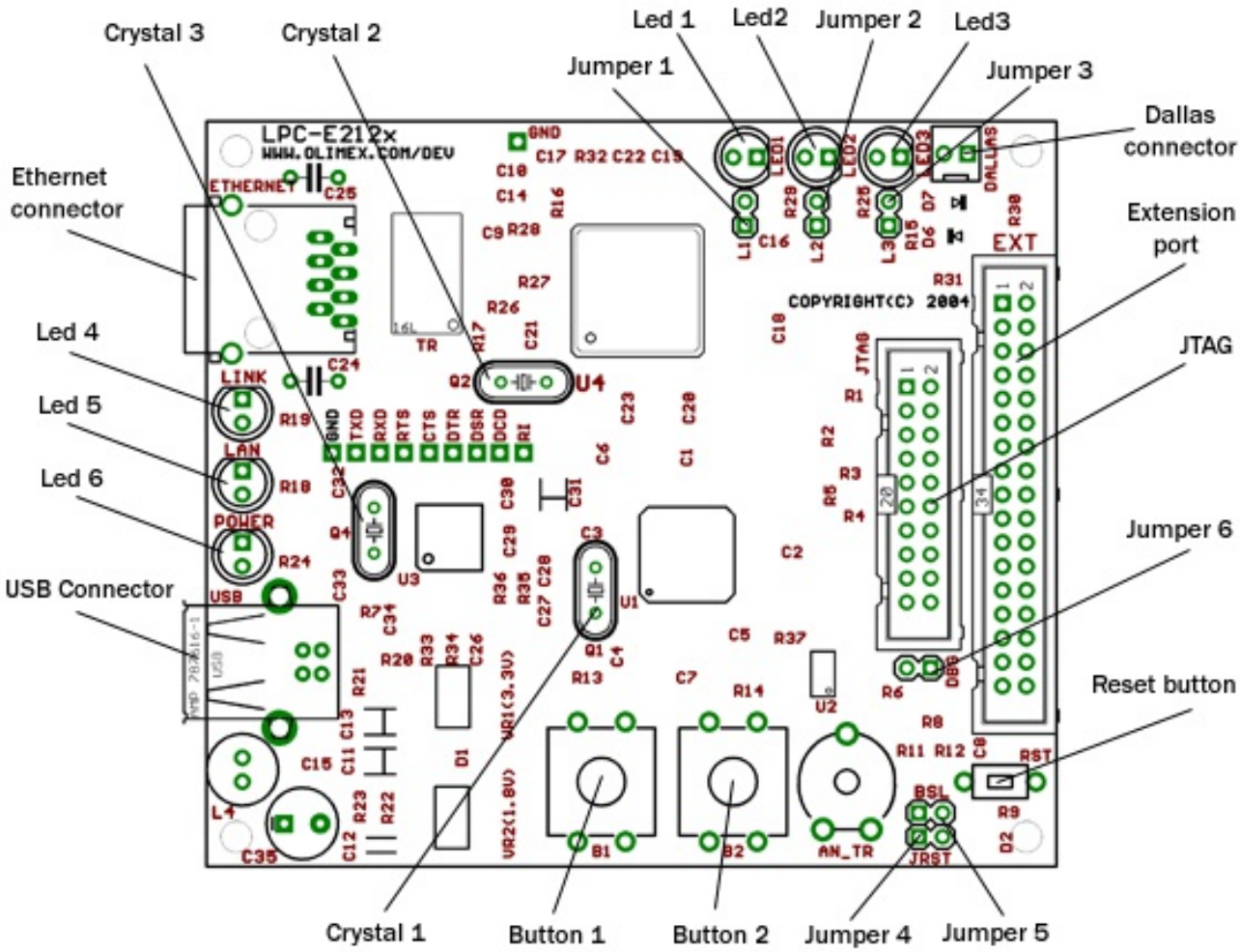
E2124




IN OUT
ADJ/GND
R22
220/1%
C12
10uF/6.3V
R23
100/1%
D6
1N4148
R31
4.7K
DALLAS
R30
33
D7
1N4148
DALLAS
BAT54
D1
VR1(3.3V)
LM1117
+3.3V
IN OUT
ADJ/GND
R20
240/1%
C11
10uF/16V
R21
390/1%
C13
47uF/6.3V
+3.3V
R1
10K
R2
10K
R3
10K
R4
10K
JTAG
TRST
TDI
TMS
TCK
RTCK
TDO
RST
DBG
R6
10K
R5
10K
C3
100n
C2
100n
C1
100n
+3.3V
1.8V
C4
100n
C5
100n
C6
100n
C7
100n
+3.3V

LPC-E2124 Board

E2124 Board



LPC-E2124 Board

LPC E2129 Hardware description

Peripherials

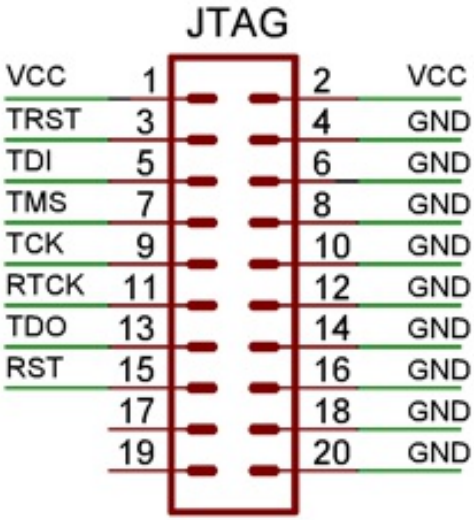
Unit	Description
Ethernet connector	RJ45 Ethernet connector
USB Connector	USB connector type B
JTAG Connector	2x10 0,1" step connector for programming with ARM-JTAG.
Dallas Connector	Interface to Dallas device connected to P0.24 / TD2 (PIN 5).
Buttons	Two buttons connected to interrupt ports Button 1 - P0.15 / RI1 / EINT2 (PIN 45) Button 2 - P0.9 / RXD1 / PWM6 / EINT3 (PIN 34)
Leds	Red status led (L1) connected to P0.8 / TXD1 / PWM4 (PIN 33) Green status led (L2) connected to P0.10 / RTS1 / CAP1.0 (PIN 35) Yellow status led (L3) connected to P0.11 / CTS1 / CAP1.1 (PIN 37) and three LAN status leds

Technical characteristics

Parameter	Description
Voltage Supply	+ 5.0V DC from USB
CPU	LPC2124
Crystals	Crystal 1 - Q1 - 14,745 MHz crystal Crystal 2 - Q2 - 20 MHz crystal Crystal 3 - Q4 - 6 MHz crystal
Board dimensions	80x90 mm (3.15x3.55")
PCB	FR-4, 1.5 mm (0,062"), green soldermask, white silkscreen component print
Operating Temperature	form 0ºC to 70ºC

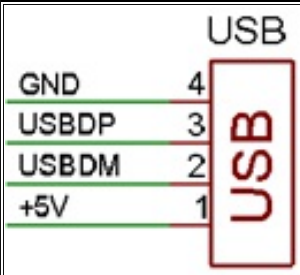
JTAG Connector

Pin / Name	Connected to:	Functionality



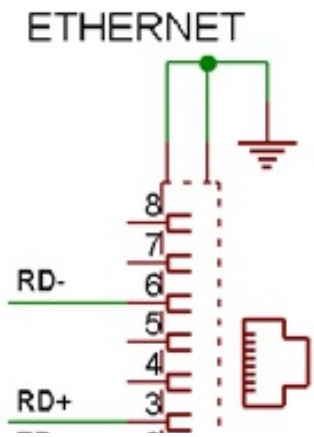
1 - VCC	VCC	-
2 - VCC	VCC	-
3 - TRST	PIN 20	P1.31 / TRST
4 - GND	GROUND	-
5 - TDI	PIN 60	P1.28 / TDI
6 - GND	GROUND	-
7 - TMS	PIN 52	P1.30 / TMS
8 - GND	GROUND	-
9 - TCK	PIN 56	P1.29 / TCK
10 - GND	GROUND	-
11 - RTCK	PIN 24	P1.26 / RTCK
12 - GND	GROUND	-
13 - TDO	PIN 64	P1.27 / TDO
14 - GND	GROUND	-
15 - RST	PIN 57	RST
16 - GND	GROUND	-
17 - -	no connected	-
18 - GND	GROUND	-
19 - -	no connected	-
20 - GND	GROUND	-

USB Connector



Pin / Name	Connected to:	Functionality
1 - +5V	+5V DC	-
2 - USBDM	FT232BM (PIN 8)	USBDM
3 - USBDP	FT232BM (PIN 7)	USBDP
4 - GND	GROUND	-

Ethernet Connector



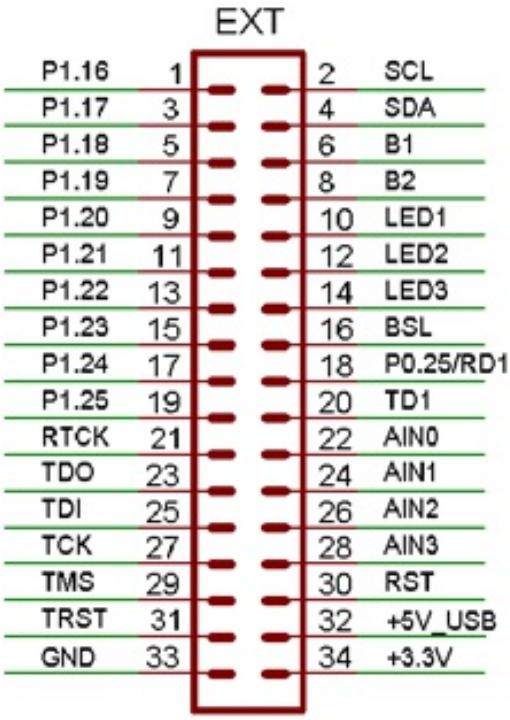
Pin / Name	Connected to:	Functionality
1 - TD+	CS8900A PIN 87	TXD+
2 - TD-	CS8900A PIN 88	TXD-
3 - RD+	CS8900A PIN 91	RXD+
4 -	not connected	-
5 -	not connected	-
6 - RD-	CS8900A PIN 92	RXD-
7 -	not connected	-

8 -	not connected	-

Dallas Connector

Pin / Name	Connected to:	Functionality
1 - DALLAS	PIN 5	P0.24 / TD2
2 - GND	GROUND	-

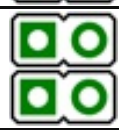
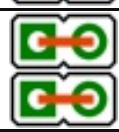
Extension ports



Pin / Name	Connected to:	Functionality
PIN 1	PIN 16	P1.16 / TRACEPTK0
PIN 2	PIN 22	P0.2 / SCL / CAP0.0
PIN 3	PIN 12	P1.17 / TRACEPTK1
PIN 4	PIN 26	P0.3 / SDA / MAT0.0 / EINT1
PIN 5	PIN 8	P1.18 / TRACEPTK2
PIN 6	PIN 45	P0.15 / RI1 / EINT2
PIN 7	PIN 4	P1.19 / TRACEPTK3
PIN 8	PIN 34	P0.09 / RXD1 / PWM6 / INT3
PIN 9	PIN 48	P1.20 / TRACESYNC
PIN 10	PIN 33	P0.8 / TXD1 / PWM4
PIN 11	PIN 44	P1.21 / PIPESTAT0
PIN 12	PIN 35	P0.10 / RTS1 / CAP1.0
PIN 13	PIN 40	P1.22 / PIPESTAT1
PIN 14	PIN 37	P0.11 / CTS1 / CAP1.1
PIN 15	PIN 36	P1.23 / PIPESTAT2
PIN 16	PIN 41	P0.14 / DCD1 / EINT1
PIN 17	PIN 32	P1.24 / TRACECLK
PIN 18	PIN 9	P0.25 / RD1
PIN 19	PIN 28	P1.25 / EXTIN0
PIN 20	PIN 10	TD1
PIN 21	PIN 24	P1.26 / RTCK
PIN 22	PIN 11	P0.27 / AINT0 / CAP0.1 / MAT0.1
PIN 23	PIN 64	P1.27 / TDO
PIN 24	PIN 13	P0.28 / AINT1 / CAP0.2 / MAT0.2
PIN 25	PIN 60	P1.28 / TDI
PIN 26	PIN 14	P0.29 / AINT2 / CAP0.3 / MAT0.3
PIN 27	PIN 56	P1.29 / TCK
PIN 28	PIN 15	P0.30 / AINT3 / EINT3 / CAP0.0
PIN 29	PIN 52	P1.30 / TMS
PIN 30	PIN 57	RST

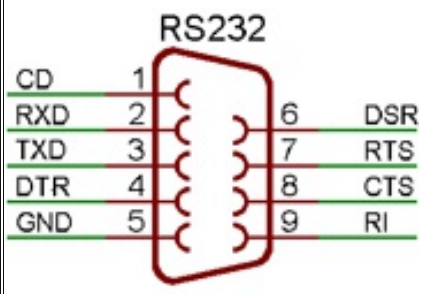
PIN 31	PIN 20	P1.31 / TRST
PIN 32	+5V USB	-
PIN 33	GROUND	-
PIN 34	+3.3V	-

Jumpers

Jumpers	Position	Description
Jumper 1 (L1)		Red led is not connected.
		Red led connected to P0.8 / TXD1 / PWM4 (PIN 33).
Jumper 2 (L2)		Green led is not connected.
		Green led connected to P0.10 / RTS1 / CAP1.0 (PIN 35).
Jumper 3 (L3)		Yellow led is not connected.
		Yellow led connected to P0.11 / CTS1 / CAP1.1 (PIN 37).
Jumper 4 (JRST) Jumper 5 (BSL)		Disable ICSP programming.
		Enable ICSP programming - via USB Connector (virtual COM port).
Jumper 6 (DBG)		Disable JTAG programming.
		Enable JTAG programming.

Programming: RS232

RS232 Connector



Pin / Name	Description
1 - CD	**Carrier Detected.**
2 - RXD	**Received Data.**
3 - TXD	**Transmited Data.**
4 - DTR	**Data Terminal Ready.**
5 - GND	**Signal Ground.**
6 - DSR	**Data Set Ready.**
7 - RTS	**Request to Send.**
8 - CTS	**Clear to Send..**
9 - RI	**Ring Indicator.**

Register description

Register	Address	Function
U0RBR	0xE000C000 DLAB = 0	**Receiver Buffer Register.** Input data buffer.
U0THR	0xE000C000 DLAB = 0	**Transmit Holding Register.** Output data buffer.
U0DLL	0xE000C000 DLAB = 1	**Divisor Latch LSB.**
U0DLM	0xE000C000 DLAB = 1	**Divisor Latch MSB.**
U0IER	0xE000C004 DLAB = 0	**Interrupt Enable Register.**
U0IIR	0xE000C008	**Interrupt ID Register.**
U0FCR	0xE000C008	**FIFO Control Register.**

U0LCR	0xE000C00C	**Line Control Register.**
U0LSR	0xE000C014	**Line Status Register.**
U0SCR	0xE000C01C	**Scratch Pad Register.**
U0TER	0xE000C030	**Transmit Enable.**

1.Initialization

1.1. Set Line Control Register

U0LCR	Function	Description	Reset Value
1:0	Word Length Select	00: 5 bit character length 01: 6 bit character length 10: 7 bit character length 11: 8 bit character length	0
2	Stop Bit Select	0: 1 stop bit 1: 2 stop bits (1.5 if U0LCR[1:0]=00)	0
3	Parity Enable	0: Disable parity generation and checking 1: Enable parity generation and checking	0
5:4	Parity Select	00: Odd parity 01: Even parity 10: Forced "1" stick parity 11: Forced "0" stick parity	0
6	Break Control	0: Disable break transmission 1: Enable break transmission. Output pin UART0 TxD is forced to logic 0 when U0LCR6 is active high.	0

7	Divisor Latch Access Bit	0: Disable access to Divisor Latches 1: Enable access to Divisor Latches	0

1.2. UART0 Baudrate Calculation

The U0DLL and U0DLM registers together form a 16 bit divisor where U0DLL contains the lower 8 bits of the divisor and U0DLM contains the higher 8 bits of the divisor.

```
devisor = pclk / (16 * baud);
```

1.3. Set Functionality to pins

Set functionality to P0.0 -> TX0 and P0.1 -> RXD0

2. RS232 Communication

2.1. Write to RS232

Use follow algorithm to send data:
- fill U0THR register with data to write
- wait shift all data
- clear interrupt flag

2.2. Read from RS232

Use follow algorithm to receive data:
- wait read all data
- clear interrupt flag
- get data from U0RBR

3. Example

Initialize:

```
//set Line Control Register (8 bit, 1 stop bit, no parity, enable DLAB)
U0LCR_bit.WLS = 0x3;     //8 bit
U0LCR_bit.SBS = 0x0;     //1 stop bit
U0LCR_bit.PE = 0x0;      //no parity
U0LCR_bit.DLAB = 0x1;    //enable DLAB

//devisor
```

```
U0DLL = Pclk / (16 * baud);       //low bite
U0DLM = Pclk / (16 * baud)>>8;    //high bite
U0LCR &= ~0x80;

//set functionality to pins: port0.0 -> TX0, port0.1 -> RXD0
PINSEL0_bit.P0_0 = 0x1;
PINSEL0_bit.P0_1 = 0x1;
```

Read Data:

```
//when U0LSR_bit.DR is 1 - U0RBR contains valid data
while (U0LSR_bit.DR == 0);
return U0RBR;
```

Write Data:

```
//when U0LSR_bit.THRE is 1 - U0THR contains valid data.
while (U0LSR_bit.THRE == 0);
U0THR = ch0;
```

Programming: Real Time Clock

Register description

Register	Address	Function
ILR	0xE0024000	**Interrupt Location.** Reading this location indicates the source of an interrupt. Writing a one to the appropriate bit at this location clears the associated interrupt.
CTC	0xE0024004	**Clock Tick Counter.** Value from the clock divider.
CCR	0xE0024008	**Clock Control Register.** Controls the function of the clock divider.
CIIR	0xE002400C	**Counter Increment Interrupt.** Selects which counters will generate an interrupt when they are incremented.
AMR	0xE0024010	**Alarm Mask Register.** Controls which of the alarm registers are masked. RW
CTIME0	0xE0024014	**Consolidated Time Register 0**
CTIME1	0xE0024018	**Consolidated Time Register 1**
CTIME2	0xE002401C	**Consolidated Time Register 2**

1.Initialization

1.1. Turn on the 32KHz external clock

CLKSRC (bit 4 from CCR Register) 0 - Disable 32kHz external clock
1 - Enable 32kHz external clock

1.2. Enable Interrupt

CIIR	Function	Description
0	IMSEC	When one, an increment of the Second value generates an interrupt

0	IMSEC	When one, an increment of the Second value generates an interrupt.
1	IMMIN	When one, an increment of the Minute value generates an interrupt.
2	IMHOUR	When one, an increment of the Hour value generates an interrupt.
3	IMDOM	When one, an increment of the Day of Month value generates an interrupt.
4	IMDOW	When one, an increment of the Day of Week value generates an interrupt.
5	IMDOY	When one, an increment of the Day of Year value generates an interrupt.
6	IMMON	When one, an increment of the Month value generates an interrupt.
7	IMYEAR	When one, an increment of the Year value generates an interrupt.

1.3. Start the Real Time Clock

CLKEN (bit 0 from CCR Register) Enable/Disable Real Time Clock
0 - Disable Real Time Clock
1 - Enable Real Time Clock

2. Example

Initialize:

```
CCR_bit.CLKEN  = 0;     //rtc disable
CCR_bit.CLKSRC = 1;     //set external 32kHz oscillator
CCR_bit.CTCRST = 0;     //disable reset
CCR_bit.CTTEST = 0;     //disable test
AMR            = 0;     //initialize interrupt mask register of RTC
CIIR_bit.IMSEC = 1;     //enable interupt every seconds
ILR            = 3;     //clear all interrupt of RTC
CCR_bit.CLKEN  = 1;     //rtc enable
```

LPC-E2124 Board

Programming: Blinking LED

GPIO Register map

Generic name	Description
IOPIN	**GPIO Port Pin value register**. The current state of the GPIO configured port pins can always be read from this register, regardless of pin direction and mode. Activity on non-GPIO configured pins will not be reflected in this register.
IOSET	**GPIO Port Output set register.** This register controls the state of output pins in conjunction with the IOCLR register. Writing ones produces highs at the corresponding port pins. Writing zeroes has no effect.
IODIR	**GPIO Port Direction control register.** This register individually controls the direction of each port pin.
IOCLR	**GPIO Port Output clear register.** This register controls the state of output pins. Writing ones produces lows at the corresponding port pins and clears the corresponding bits in the IOSET register. Writing zeroes has no effect.

Pin Connect Block Register Map

Register name	Description
PINSEL0	PINSEL0 Pin function select register 0 (from P0.0 to P0.15)
PINSEL1	PINSEL1 Pin function select register 1 (from P0.16 to P0.31)
PINSEL2	PINSEL2 Pin function select register 2

1. Initialization (general case)

1.1. Set first functionality to port

```
PINSEL1 = 0x00;     //set first functionality to port (from P0.0 to P0.15)
```

1.2. Set port which is connected to LED as output

```
IO0DIR = 0xFF;    //set P0.0 to P0.15 port as output
```

2. Led blink (general case)

```
IO0CLR = 0xFF;   // set P0.0 - P0.15 to low
IO0SET = 0xFF;   // set P0.0 - P0.15 to high
```

3. Example - blink led, which is connected to P0.8

//Initialization

```
  PINSEL0_bit.P0_8 = 0x0;  // set first functionality to port
  IO0DIR_bit.P0_8  = 0x1;  // set P0.8 port to output
  IO0SET_bit.P0_8  = 0x1;  // set P0.8 port to high
```

//loop forever

```
while(1)
{
   Delay(1000);            // Simple delay
   IO0SET_bit.P0_8 = 0x1;  // set P0.8 port to high
   Delay(1000);            // Simple delay
   IO0CLR_bit.P0_8 = 0x1;  // set P0.8 port to high
}
```

LPC-E2124 Board

Links

1. Philips web site

LPC2124 product datasheets, application notes, etc info:
http://www.semiconductors.philips.com/

2. LPC microcontrollers discussion forum

http://groups.yahoo.com/group/lpc2000/ - forum for discussions on LPC2000 ARM microcontrollers

http://groups.yahoo.com/group/arm-olimex/ - forum for discussions on Olimex ARM boards

3. IAR Systems EW-ARM C compiler and debugger

http://www.iar.com/Products/?name=EWARM

4. Rowley associates CrossWorks for ARM C compiler and debugger

http://www.rowley.co.uk