TinyARM PHIL50 LPC21x8 Module in DIP50 package

1. Introduction

The purpose of this document is to specify mechanical dimensions and electrical behavior of PHIL50 LPC21x8 DIP50 modules.

2. Reason for development

Philips introduced ARM LPC2138 and LPC2148 processors housed in QFP64 pin package. They feature 256kbyte of Flash memory and 32kbyte SRAM.
This chip can be used as a replacement for 8/16 microcontrollers.
There are following problems when using this chip:
A. Price of development tools, both software and hardware.
B. Unusual package which is not convenienent for development.

3. Solution

TinyARM's PHIL50 module intends to solve all problems:
A. The module is housed in DIP50 package.
B. It requires just one convenient 5V supply. All the voltages required for Philips processor are generated by TinyArm module.
C. It features built-in 10 MHz oscillator.
D. It includes Reset generator.
E. The module includes 32768Hz quartz and backup voltage generation for real time clock section of the processor.
F. LPC2148 module includes full speed USB interface.
G. The module is re-usable. It can be used in several designs, the only
additional investment is DIL50 socket for each design.
F. Software tools are either IAR or free GNU .

4. List of signals required for debugging purposes

All the signals required for standard ARM Test/Debug JTAG interface are
generated by microcontroller in the module. This includes standard JTAG TRSTN,TCLK, TMS,TDI, TDO and RTCK signals.
Reset signal for all circuitry is generated on board.

5. Voltage levels and electrical behavior

The module runs from 5V supply applied to DIP50 pin 50.
The supply voltage for this version can be from 4.5V to 5.5V.
The estimated current consumption is 60mA from 5V supply.

The module can supply 3.3V and 1.8V voltages. There are 3 holes, each 0.8 mm in diameter, left one being the output of 1.8V regulator, the middle one is ground and the right one is the output of 3.3V regulator.
Do not draw more than 20mA from any regulator.

Voltage levels for ARM I/O pins are 3.3V compliant. The specs says that
I/O pins are 5V tolerant.

6. Method of programming internal Flash

There are 2 methods to program Flash – either JTAG or ISP (using UART0 of Philips LPC21x8 chip). Please, refer to the relevant Philips LPC21x8
manual.
Note: P0.14 must be logic low for TinyARM to enter bootloader mode.
P0.14 is pulled up to 3.3V supply on TinyARM PCB so that TinyARM

does not enter bootloader mode.

7. Mechanical dimensions

The module should fit into the standard 600 mils DIP50 socket. Each module is supplied with 1 pc of DIL50 socket to help development.

8. Tips and tricks for development

P0.14 must be logic low for TinyARM to enter bootloader mode.
P0.14 is pulled up to 3.3V supply on TinyARM PCB so that TinyARM does not enter bootloader mode.

9. Preliminary drawing

None.

10. TinyARM's board control signals

RSTN – input for external reset signal, active low. Leave unconnected if not used. TinyARM board generates proper internal Reset signal upon power-up. An external push button can be connected to RSTN to generate reset signal. No debounce is needed for the push button.
Note: If driving RSTN pin by external circuitry, then it must be open collector (open drain) with pull-up resistor. Do NOT use push-pull to drive RSTN !

RTCK – Multiplexed with P1.26. leave this pin unconnected for normal operation.

Signals available on "virtual" pins:
RSTOUTN – output of the board internal reset generator, active low, open drain output. This signal is connected to "virtual" pin 52 which is positioned to the left from pin 26.

VBAT – battery voltage input which powers RTC section of LPC21x8 microcontrollers. This signal is connected to "virtual" pin 51 which is positioned to the right from pin 25.

11. Interfacing TinyARM board to various voltage levels

TinyARM board runs from 5V supply and generates 3.3V voltage for I/O pins logic levels reference. So all TinyARM I/Os are 3.3V compliant, meaning they can directly interface with 3.3V logic. They can also interface directly 5V LV TTL and HCT MOS logic.
TinyARM I/Os are 5V resistant so they can receive 5V TTL/HC/HCT logic levels. However, I/O cannot drive 5V HC MOS inputs directly, some level conversion is necessary. The simplest method for slow signals is using pullup resistor to 5V supply.

12. Simple test / First use of TinyARM

It is easy to start with TinyARM. All what you need is an experimental board, DIP50 socket, RS232 interface circuit like MAX232 and 5V power supply. Philips supplies LPC21xx ISP utility for Windows which can talk to TinyArm's UART0 serial interface.

Make the connections like the following:

- connect ground to DIP50 pin 25
- connect +5V to DIP50 pin 50
- apply logic low (pull down 3k9 to ground) to DIP50 pin 15 which corresponds to LPC21x8 Port0.14. This pin is samples during reset or power-up to indicate whether to start ISP utility.
- connect DIP50 pin 1 which corresponds to LPC21x8 Port00/TxD0 to the input of MAX232 driver, e.g. to T1in (MAX232 pin 11). The output of MAX232 driver, e.g. T1out (MAX232 pin 14) is connected to DB9 or DB25 connector as TxD signal.
- connect DIP50 pin 1 which corresponds to LPC21x8 Port01/RxD0 to the output of MAX232 receiver, e.g. to R1out (MAX232 pin 12). The input of MAX232 driver, e.g. R1in (MAX232 pin 13) is connected to DB9 or DB25 connector as RxD signal.
- Connect a push-button between DIP50 pin 26 RSTN and ground.
- Insert PHIL50 board into DIP50 socket.
- Connect RS232 cable to the PC on one side and to your board on the other side.
- Apply power from 5V supply.
- Start Philips LPC21xx utility
- Press push button to connect module to the PC, if necessary.
- ENJOY !

14. Connecting Macraigor Systems WIGGLER to TinyARM board

The easiest way is to use extra DIP50 socket and connect DIP50 pins to the male dual-row 20 pin header as follows:

DIP50 pin	Header pin	Signal name
25	4,6,8,10,12,14 16,18	GND
34	3	TRSTn
33	7	TMS
32	9	TCK
31	5	TDI
30	13	TDO
26	15	RESETn
24	11	RTCK (P1.26 – see notes)
3.3V	1	TVcc

Some additional resistors are strongly recommended to apply correct default voltage levels on JTAG pins.

Notes:

a.LOW on P1.26/RTCK pin while RESETn is LOW enables pins P1.31 to P1.26 to operate as a Debug port after reset. Use 1k pull-down resistor on that pin for the part to enter JTAG debug mode.

b.LOW on P1.20 pin while RESETn is LOW enables pins P1.25 to P1.16

to operate as a Debug port after reset. Use 1k pull-down resistor on
that pin for the part to enter extended debug mode.
c.The module can supply 3.3V voltage. See section 5. for details.

15. TinyARM PHIL50 LPC2138 board pinout

Pin	Pin Name	Altern.		Pin	Pin Name	Altern.
1	P0.0	TxD0/PWM1		50	VCC	
2	P0.1	RxD0/PWM3/EINT0		49	P0.30	AD0.3/EINT3/CAP0.0
3	P0.2 *1	SCL0/CAP0.0		48	P0.29	AD0.2/CAP0.3/MAT0.3
4	P0.3 *1	SDA0/MAT0.0/EINT1		47	P0.28	AD0.1/CAP0.2/MAT0.2
5	P0.4	SCK0/CAP0.1/AD0.6		46	P0.27	AD0.0/CAP0.1/MAT0.1
6	P0.5	MISO0/MAT0.1/AD0.7		45	P0.26	AD0.5
7	P0.6	MOSI0/CAP0.2/AD1.0		44	P0.25	AD0.4/AOUT
8	P0.7	SSEL0/PWM2/EINT2		43	P0.31	----
9	P0.8	TxD1/PWM4/AD1.1		42	P0.23	----
10	P0.9	RxD1/PWM6/EINT3		41	P0.22	CAP0.0/MAT0.0/AD1.7
11	P0.10	RTS1/CAP1.0/AD1.2		40	P0.21	PWM5/CAP1.3/AD1.6
12	P0.11 *1	CTS1/CAP1.1/SCL1		39	P0.20	MAT1.3/SSEL1/EINT3
13	P0.12	DSR1/MAT1.0/AD1.3		38	P0.19	MAT1.2/MOSI1/CAP12
14	P0.13	DTR1/MAT1.1/AD1.4		37	P0.18	CAP1.3/MISO1/MAT13
15	P0.14 *2	DCD1/EINT1/SDA1		36	P0.17	CAP1.2/SCK1/MAT1.2
16	P0.15	RI1/EINT2/AD1.5		35	P0.16	EINT0/MAT0.2/CAP02
17	P1.16	TRACEPKT0		34	P1.31	**TRSTn**
18	P1.17	TRACEPKT1		33	P1.30	**TMS**
19	P1.18	TRACEPKT2		32	P1.29	**TCK**
20	P1.19	TRACEPKT3		31	P1.28	**TDI**
21	P1.20	TRACESYNC		30	P1.27	**TDO**
22	P1.21	PIPESTAT0		29	P1.26	**RTCK**
23	P1.22	PIPESTAT1		28	P1.25	EXTIN0
24	P1.23	PIPESTAT2		27	P1.24	TRACECLK
25	GND			26	RSTN	

Notes:
*1 Open drain output – requires pullup
*2 P0.14 must be logic low for TinyARM to enter bootloader mode.
P0.14 is pulled up to 3.3V supply so that LPC2138 does not enter bootloader mode.
P0.14 is open drain output – requires pullup

16. TinyARM PHIL50 LPC2148 board pinout

Pin	Pin Name	Altern.		Pin	Pin Name	Altern.
1	P0.0	TxD0/PWM1		50	VCC	
2	P0.1	RxD0/PWM3/EINT0		49	P0.30	AD0.3/EINT3/CAP0.0
3	P0.2 *1	SCL0/CAP0.0		48	P0.29	AD0.2/CAP0.3/MAT0.3
4	P0.3 *1	SDA0/MAT0.0/EINT1		47	P0.28	AD0.1/CAP0.2/MAT0.2
5	P0.4	SCK0/CAP0.1/AD0.6		46	---	
6	P0.5	MISO0/MAT0.1/AD0.7		45	---	
7	P0.6	MOSI0/CAP0.2/AD1.0		44	P0.25	AD0.4/AOUT
8	P0.7	SSEL0/PWM2/EINT2		43	P0.31	----
9	P0.8	TxD1/PWM4/AD1.1		42	P0.23	VBUS
10	P0.9	RxD1/PWM6/EINT3		41	P0.22	CAP0.0/MAT0.0/AD1.7
11	P0.10	RTS1/CAP1.0/AD1.2		40	P0.21	PWM5/CAP1.3/AD1.6
12	P0.11 *1	CTS1/CAP1.1/SCL1		39	P0.20	MAT1.3/SSEL1/EINT3
13	P0.12	DSR1/MAT1.0/AD1.3		38	P0.19	MAT1.2/MOSI1/CAP12
14	P0.13	DTR1/MAT1.1/AD1.4		37	P0.18	CAP1.3/MISO1/MAT13
15	P0.14 *2	DCD1/EINT1/SDA1		36	P0.17	CAP1.2/SCK1/MAT1.2
16	P0.15	RI1/EINT2/AD1.5		35	P0.16	EINT0/MAT0.2/CAP02
17	P1.16	TRACEPKT0		34	P1.31	**TRSTn**
18	P1.17	TRACEPKT1		33	P1.30	**TMS**
19	P1.18	TRACEPKT2		32	P1.29	**TCK**
20	P1.19	TRACEPKT3		31	P1.28	**TDI**
21	P1.20	TRACESYNC		30	P1.27	**TDO**
22	P1.21	PIPESTAT0		29	P1.26	**RTCK**
23	P1.22	PIPESTAT1		28	P1.25	EXTIN0
24	P1.23	PIPESTAT2		27	P1.24	TRACECLK
25	GND			26	RSTN	

Notes:

*1 Open drain output - requires pullup

*2 P0.14 must be logic low for TinyARM to enter bootloader mode.
P0.14 is pulled up to 3.3V supply so that LPC2148 does not enter bootloader mode.
P0.14 is open drain output - requires pullup

17. Known problems of Philips LPC21x8 processors

a. SPI interface requires to use pullup resistor on SSEL pin even if it is set to Master mode. That results in pin wasting, pin SSEL cannot be used as GPIO P0.7 when SPI is used.