Philips TinyARM LPC2129 DIP50 Module

1. Introduction

The purpose of this document is to specify mechanical dimensions and electrical behaviour of a new TinyARM DIP50 module.

2. Reason for development

There are various ARM derivates on the market. Philips recently introduced ARM processor housed in QFP64 pin package. It features 128kbyte or 256kbyte of Flash memory and 16kbyte 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.
C. The chip requires 2 different voltages.

3. Solution

TinyARM DIP50 module intends to solve all problems:
A. TinyARM DIP50 module is housed in DIP50 package.
B. TinyARM DIP50 requires just one convenient 5V supply. All the voltages required for Philips processor are generated by TinyArm module.
C. TinyARM DIP50 module features built-in 10 MHz oscillator.
D. TinyARM DIP50 module includes Reset generator.

4. Development version of TinyARM module

To be defined later.

5. List of required signals for development version

All the signals required for standard ARM Test/Debug JTAG interface are generated on JtagTinyARM board. This includes standard JTAG TRSTn,TCLK, TMS,TDI and TDO signals and handling of RTCK and DBGSEL signals.
Reset signal for all circuitry is generated on board.

6. Voltage levels and electrical behaviour

TinyARM 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 100mA from 5V supply.

TinyARM module can supply 3.3V voltage from its “virtual” pin 52.
Pin 52 is positioned on the right side of DIP50 pin 1 and is marked as “3V”. Do not draw more than 20mA from this supply !

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

7. Method of programming internal Flash

There are 2 methods to program Flash – either JTAG or ISP (using UART0 of Philips LPC212x chip). Please, refer to the relevant Philips LPC212x 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.

8. 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.

9. 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.

10. Preliminary drawing

None.

11. 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 "virtua" pins:
RST_OUTN – output of TinyARM internal reset generator, active low, open drain output. This signal is connected to "virtual" pin 54 which is positioned to the left from pin 26.

PWRENN – input controlling TinyARM voltage regulators. Default is low when the voltage regulators are enabled and generate 3.3V and 1.8V respectively. This signal is pulled down on TinyARM board. Normally leave this pin unconnected (N.C.). Driving this pin high will disable voltage regulators. This signal is connected to "virtual" pin 53 which is positioned to the right from pin 25.

12. 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.

13. 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
- connect logic low (pull down 3k9 to ground) to DIP50 pin 15 which corresponds to TinyARM 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 TinyARM 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 TinyARM 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 TinyARM DIP50 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 TinyARM 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:
NOTE: This was not tested YET !

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
52(virtual)	1	TVcc

Some additional resistors are strongly recommended to apply correct default voltage levels on JTAG pins.
Notes:
a. DBGSEL pin should be driven high, i.e. pullup resistor 1k to Vcc for the part to enter debug mode when debugging is possible.
b. TinyARM module can supply 3.3V voltage from its "virtual" pin 52. Pin 52 is positioned on the right side of DIP50 pin 1 and is

marked as “3V”.

15. TinyARM DIP50 board pinout

PinNr	Pin Name	Altern.		PinNr	Pin Name	Altern.
1	P0.0	TxD0/PWM1		50	VCC	
2	P0.1	RxD0/PWM3/EINT0		49	P0.30	AIN3/EINT3/CAP0.0
3	P0.2	SCL/CAP0.0		48	P0.29	AIN2/CAP0.3/MAT0.3
4	P0.3	SDA/MAT0.0/EINT1		47	P0.28	AIN1/CAP0.2/MAT0.2
5	P0.4	SCK/CAP0.1		46	P0.27	AIN0/CAP0.1/MAT0.1
6	P0.5	MISO/MAT0.1		45	TD1/N.C.	
7	P0.6	MOSI/CAP0.2		44	P0.25	RD1
8	P0.7	SSEL/PWM2/EINT2		43	P0.24	TD2
9	P0.8	TxD1/PWM4		42	P0.23	RD2
10	P0.9	RxD1/PWM6/EINT3		41	P0.22	CAP0.0/MAT0.0
11	P0.10	RTS1/CAP1.0		40	P0.21	PWM5/CAP1.3
12	P0.11	CTS1/CAP1.1		39	P0.20	MAT1.3/SSEL1/EINT3
13	P0.12	DSR1/MAT1.0		38	P0.19	MAT1.2/MOSI1/CAP12
14	P0.13	DTR1/MAT1.1		37	P0.18	CAP1.3/MISO1/MAT13
15	P0.14	DCD1/EINT1		36	P0.17	CAP1.2/SCK1/MAT1.2
16	P0.15	RI1/EINT2		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	

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.