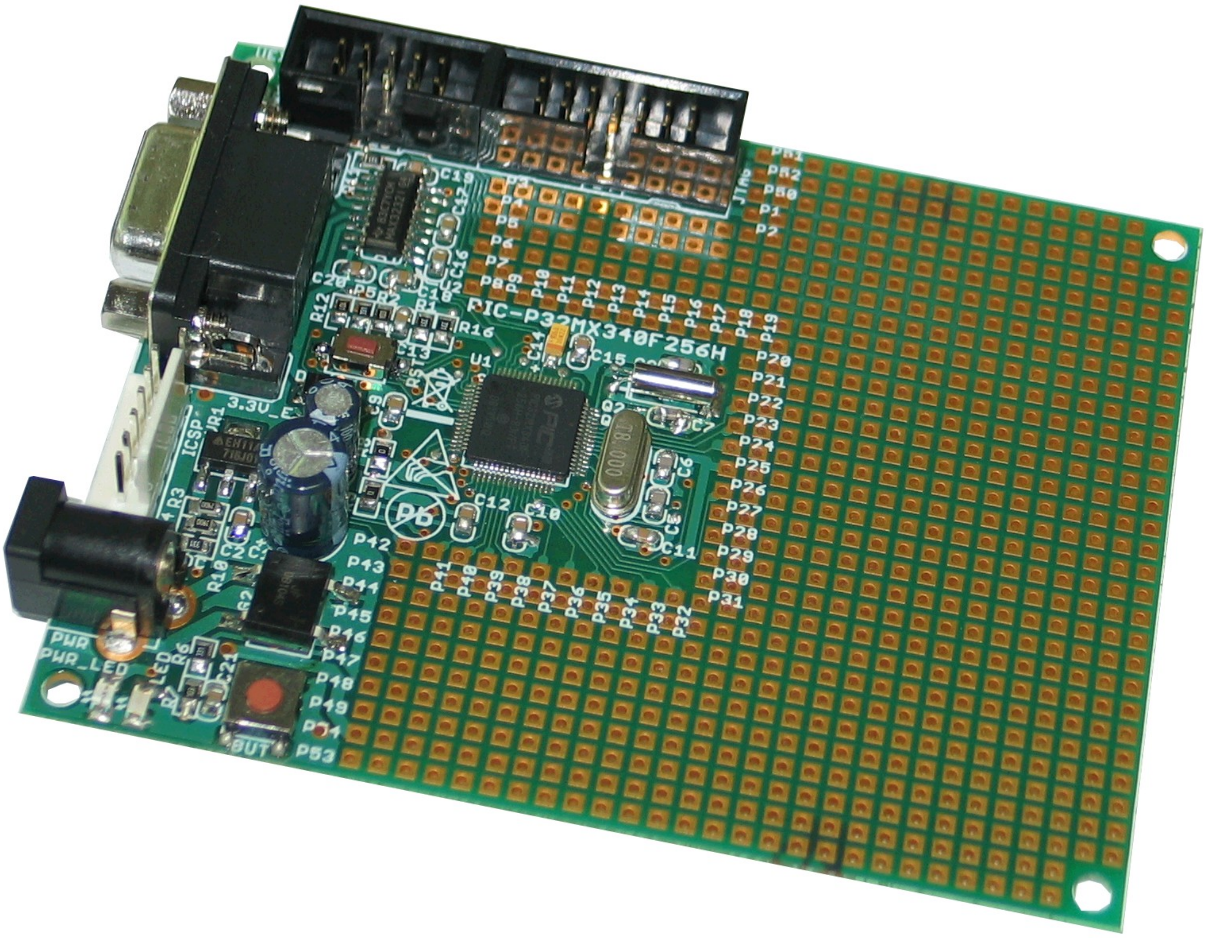


PIC-32MX development board

Users Manual

All boards produced by Olimex are ROHS compliant

Rev.A, June 2008

INTRODUCTION:

The **NEW PIC-32MX** board uses the new PIC32 32 bit MIPS 4K core processor from Microchip which offers speed and performance at low cost. This board has PIC32MX340F512 microcontroller on it with its 512 KB of Flash, 32KB RAM, 80MHz clock, UARTs, PWMs, DMAs. The board has both ICSP and JTAG connectors so it can be programmed with PIC-ICD2 or JTAG tool

BOARD FEATURES:

- PIC32MX340F512 microcontroller
- UEXT connector for other Olimex modules like MOD-MP3, MOD-NRF24Lx, MOD-SMB380, MOD-RFID125 etc.
- ICSP/ICD connector for programming with PIC-ICD2 or PIC-ICD2-POCKET.
- JTAG connector
- RS232 interface with driver
- Quartz crystalls 8 MHz and 32768 Hz
- Status LED
- User Button
- Reset button
- Power plug-in jack with diode bridge can be powered with AC or DC power supply
- 3.3V voltage regulator
- Extension slot on every uC pin
- Gird 100 mils
- GND bus
- Vcc bus
- Four mounting holes 3,3 mm (0,13")
- FR-4, 1.5 mm (0,062"), green soldermask, white silkscreen component print
- Dimensions 100x80 mm (3,9x3,15")

ELECTROSTATIC WARNING:

The PIC-32MX board is shipped in protective anti-static packaging. The board must not be subject to high electrostatic potentials. General practice for working with static sensitive devices should be applied when working with this board.

BOARD USE REQUIREMENTS:

Cables: 1.8 meter USB A-B cable to connect PIC-ICD2 or PIC-ICD2-POCKET to USB host on PC (if you use other programmer, you should read its specification in order to choose a cable).

Hardware: **PIC-ICD2, PIC-ID2-POCKET**
Or any compatible tool for programming and/or debugging

!!!Warning!!! When you want to program this microcontroller with PIC-ICD2, PIC-ICD2-POCKET or PIC-ICD2-TINY, before connecting the programmer to your target board, you should first connect the programmer to your computer and open MPLAB. There, first from menu Configure – Select Device – choose the microcontroller you are about to program, then from menu Programmer – Select Programmer – choose MPLAB ICD 2, wait while MPLAB is downloading operation system, and after ICD2 is connected – check in menu Programmer – Settings – Power – there is option – Power target circuit from MPLAB ICD 2 – this option should be forbidden, you could not select it. Now it is safe to connect the programmer to your target board.

Software: **MPLAB IDE v8.14 + MPLAB C32** for developing your own applications.
The demo software shows basic functionality and how to blink LED (C source and HEX), how to read a button (C source and HEX), the use of Timer1 (C source and HEX) and UART functions (C source and HEX). The sources are compiled with MPLAB C32 C compiler.

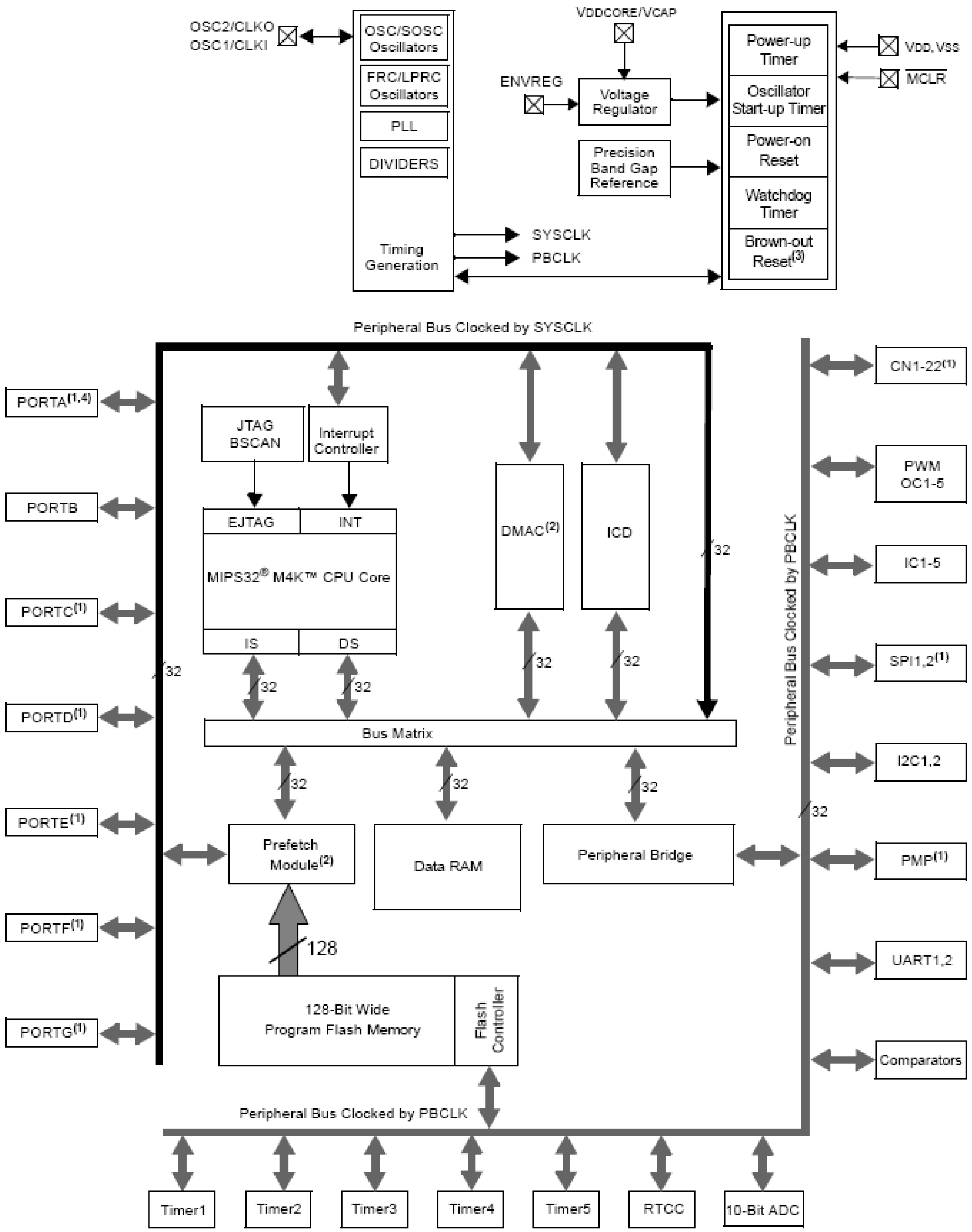
PROCESSOR FEATURES:

- High-performance RISC CPU
  - MIPS32® M4K™ 32-Bit Core with 5-Stage Pipeline
  - Single-Cycle Multiply and High-Performance Divide Unit
  - MIPS 16e™ Mode for Up to 40% Smaller Code Size
  - User and Kernel Modes to Enable Robust Embedded System
  - Two 32-Bit Core Register Files to Reduce Interrupt Latency
  - Prefetch Cache Module to Speed Execution from Flash
- Special Microcontroller Features
  - Operating Voltage Range of 2.3V to 3.6V
  - 512K Flash and 32K Data Memory
  - Additional 12 KB of Boot Flash Memory
  - Mutiple Interrupt Vectors with Individually Programmable Priority
  - Fail-Safe Clock Monitor Mode
  - Configurable Watchdog Timer with On-Chip, Low-Power RC Oscillator for Reliable Operation
- Analog Features
  - Up to 16-Channel 10-bit Analog-toDigital Converter:
    - 500 ksps conversion rate
    - Conversion available during Sleep, Idle
  - Two Analog Comparators
- Peripheral Features
  - Atomic SET, CLEAR and INVERT Operation on Select Peripheral Registers
  - Up to 4-Channel Hardware DMA Controller with Automatic Data Size Detection
  - Two I²C™ Modules
  - Two UART Modules with:
    - RS-232, RS-485 and LIN 1.2 support
    - IrDA® with on-chip hardware encoder and decoder
  - Parallel Master and Slave Port (PMP/PSP) with 8-bit and 16-bit Data and Up to 16 Addres Lines
  - Hardware Real-Time Clock/Calendar (RTCC)
  - Five 16-bit Timers/Counters (two 16-bit pairs combine to create two 32-bit timers)
  - Five Capture Inputs
  - Five Compare PWM Outputs
  - Five External Interrupts pins
  - High-Current Sink/Source (18 mA/18 mA) on All I/O Pins

- Configurable Open-Drain Output on Digital I/O

BLOCK DIAGRAM:

PIC32MX3XX/4XX BLOCK DIAGRAM (GENERAL PURPOSE)



Note 1: Not all pins or features are implemented on all device pinout configurations. See Table 1-3 for I/O port pin descriptions.

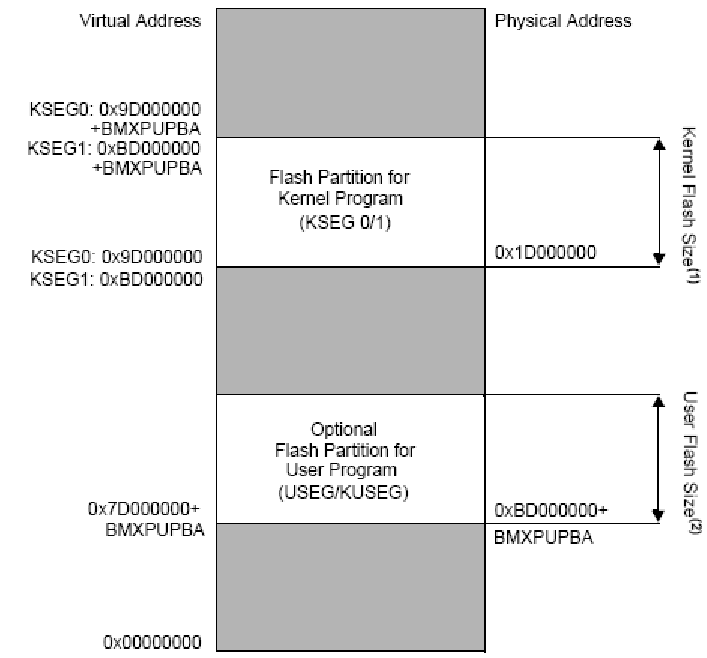
2: Some features are not available on certain devices.

3: BOR functionality is provided when the on-board voltage regulator is enabled.

4: PORTA is not present on 64-pin devices

MEMORY MAP:

FLASH PARTITIONING

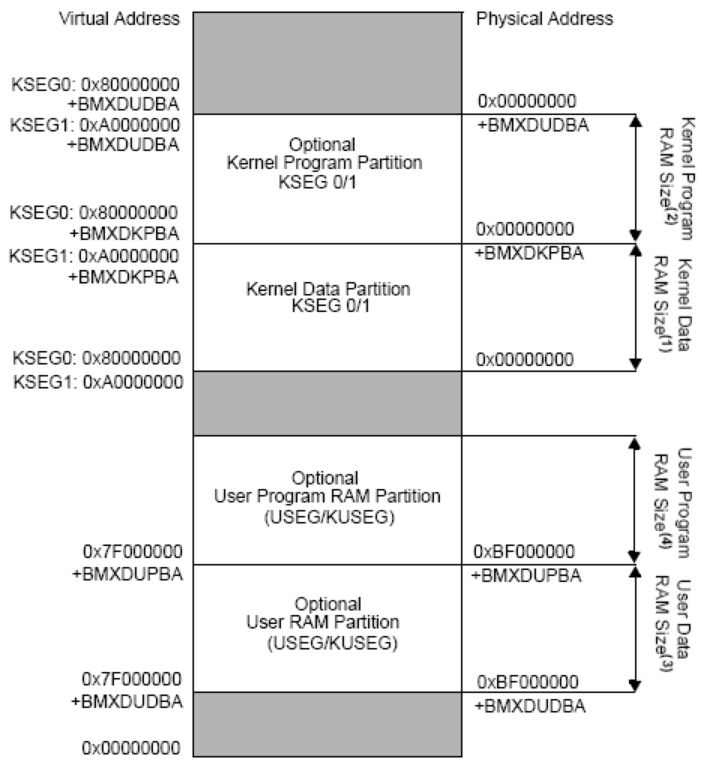


Note 1: Kernel Flash Size = BMXPUPBA

2: User Flash Size = BMXPFMSZ-BMXPUPBA

3: If BMXPUPBA is '0', then:
K Flash Size = BMXPFMSZ (i.e., all the Flash)
Usr Flash Size = 0

RAM PARTITIONING



Note 1: Kernel Data RAM Size = BMXDKPBA

2: Kernel Program RAM Size = BMXDUDBA – BMXDKPBA

3: User Data RAM Size = BMXDUPBA – BMXDUDBA

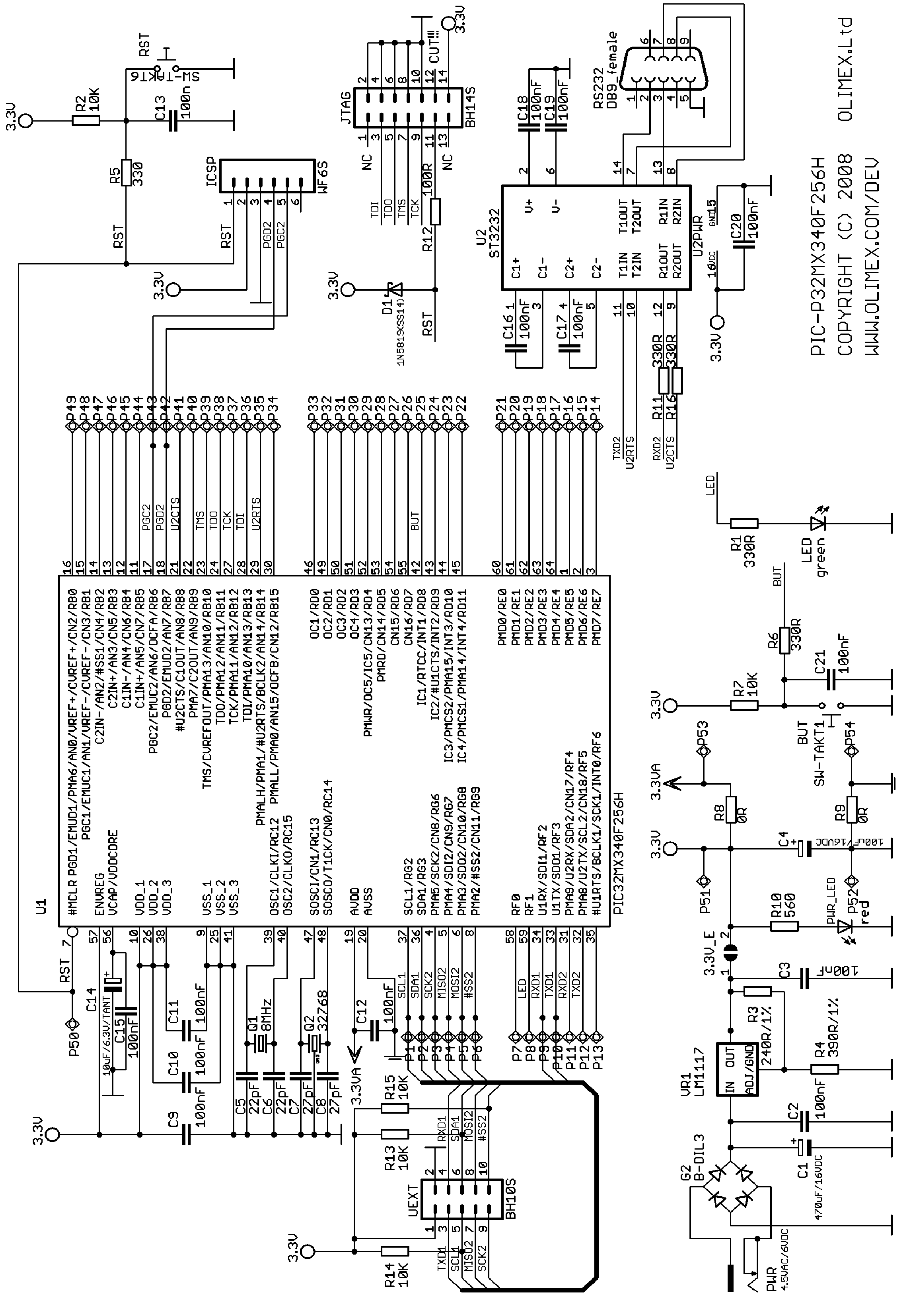
4: User Program RAM Size = DRM Size – BMXDUPBA

5: If BMXDKPBA, BMXDUDBA or BMXDUPBA is ‘0’, then:
Kernel Data RAM Size = BMXDRMSZ (i.e., all RAM)
Kernel Program RAM Size = 0
User Data RAM Size = 0
User Program RAM Size = 0

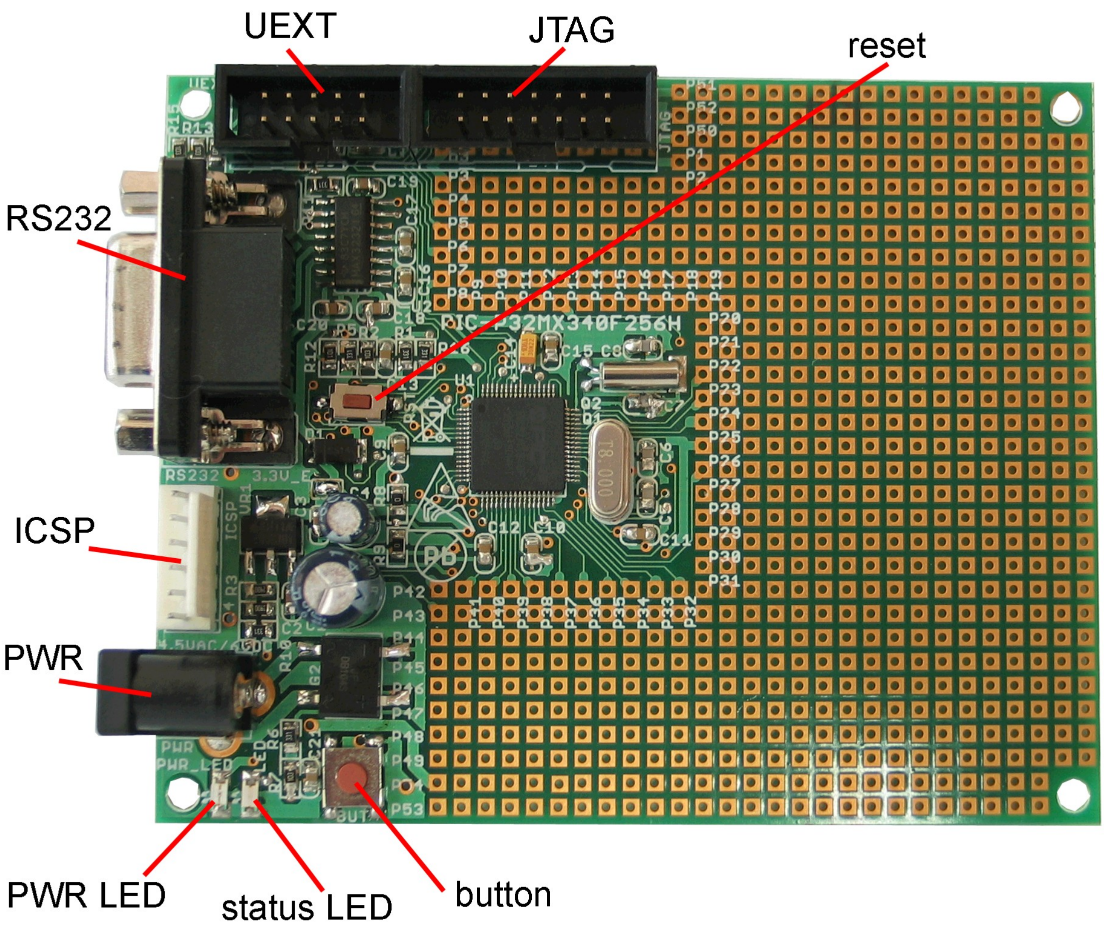
PERIPHERAL ADDRESS TABLE

Peripheral	Virtual Address		Physical Address	
	Start	End	Start	End
WDT	BF80_0000	BF80_01FF	1F80_0000	1F80_01FF
RTCC	BF80_0200	BF80_03FF	1F80_0200	1F80_03FF
TMR1	BF80_0600	BF80_07FF	1F80_0600	1F80_07FF
TMR2	BF80_0800	BF80_09FF	1F80_0800	1F80_09FF
TMR3	BF80_0A00	BF80_0BFF	1F80_0A00	1F80_0BFF
TMR4	BF80_0C00	BF80_0DFF	1F80_0C00	1F80_0DFF
TMR5	BF80_0E00	BF80_0FFF	1F80_0E00	1F80_0FFF
Input Capture1	BF80_2000	BF80_21FF	1F80_2000	1F80_21FF
Input Capture2	BF80_2200	BF80_23FF	1F80_2200	1F80_23FF
Input Capture3	BF80_2400	BF80_25FF	1F80_2400	1F80_25FF
Input Capture4	BF80_2600	BF80_27FF	1F80_2600	1F80_27FF
Input Capture5	BF80_2800	BF80_29FF	1F80_2800	1F80_29FF
Output Compare1	BF80_3000	BF80_31FF	1F80_3000	1F80_31FF
Output Compare2	BF80_3200	BF80_33FF	1F80_3200	1F80_33FF
Output Compare3	BF80_3400	BF80_35FF	1F80_3400	1F80_35FF
Output Compare4	BF80_3600	BF80_37FF	1F80_3600	1F80_37FF
Output Compare5	BF80_3800	BF80_39FF	1F80_3800	1F80_39FF
I2C1	BF80_5000	BF80_51FF	1F80_5000	1F80_51FF
I2C2	BF80_5200	BF80_53FF	1F80_5200	1F80_53FF
SPI1	BF80_5800	BF80_59FF	1F80_5800	1F80_59FF
SPI2	BF80_5A00	BF80_5BFF	1F80_5A00	1F80_5BFF
UART1	BF80_6000	BF80_61FF	1F80_6000	1F80_61FF
UART2	BF80_6200	BF80_63FF	1F80_6200	1F80_63FF
Parallel Master Port	BF80_7000	BF80_71FF	1F80_7000	1F80_71FF
GPIO	BF80_8000	BF80_81FF	1F80_8000	1F80_81FF
ADC	BF80_9000	BF80_91FF	1F80_9000	1F80_91FF
Comparator Voltage REF	BF80_9800	BF80_99FF	1F80_9800	1F80_99FF
Comparator	BF80_A000	BF80_A1FF	1F80_A000	1F80_A1FF
Oscillator	BF80_F000	BF80_F1FF	1F80_F000	1F80_F1FF
Configuration	BF80_F200	BF80_F3FF	1F80_F200	1F80_F3FF
Flash (NVM)	BF80_F400	BF80_F5FF	1F80_F400	1F80_F5FF
Reset	BF80_F600	BF80_F7FF	1F80_F600	1F80_F7FF
Interrupts	BF88_1000	BF88_1FFF	1F88_1000	1F88_1FFF
Bus Matrix	BF88_2000	BF88_2FFF	1F88_2000	1F88_2FFF
DMA	BF88_3000	BF88_3FFF	1F88_3000	1F88_3FFF
Prefetch Cache	BF88_4000	BF88_4FFF	1F88_4000	1F88_4FFF
GPIO	BF88_6000	BF88_61FF	1F88_6000	1F88_61FF

SCHEMATIC:


PIC-P32MX340F256H
COPYRIGHT (C) 2008
OLIMEX.Ltd
WWW.OLIMEX.COM/DEV

BOARD LAYOUT:



POWER SUPPLY CIRCUIT:

The power supply of PIC-32MX could be either 4.5-6.0VAC or 6VDC-9VDC. The power consumption is round 65mA.

RESET CIRCUIT:

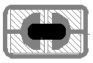
PIC-32MX reset circuit is made with a capacitor C13 (100nF) and a resistor R2 (10KΩ).

CLOCK CIRCUIT:

Quartz crystals at 8MHz and 32.768 KHz are connected to PIC-32MX.

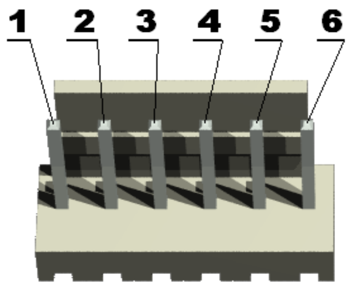
JUMPERS DESCRIPTION:

3.3V_E Enables 3.3V supply for **PIC32MX** and all other devices.

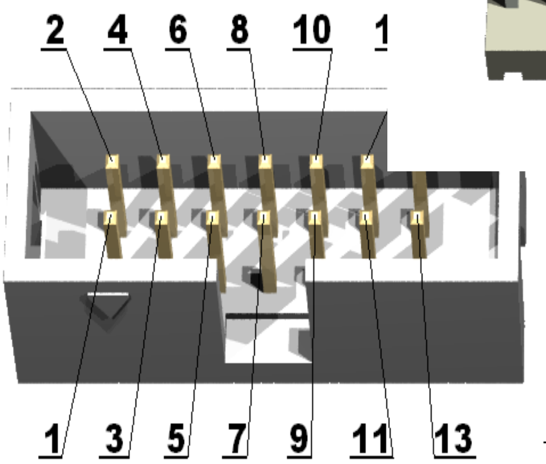


Default state closed (shorted).

CONNECTOR DESCRIPTIONS:

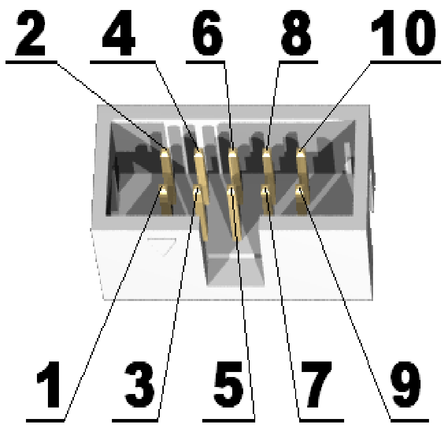


JTAG:



Pin #	Signal Name	Pin #	Signal Name
1	NC	2	GND
3	TDI	4	GND
5	TDO	6	GND
7	TMS	8	GND
9	TCK	10	GND
11	RST	12	(*)
13	NC	14	3.3V

(*) This pin isn't mounted.



TDI Input **Test Data In**. This is the serial data input for the shift register.
TDO Output **Test Data Out**. This is the serial data output for the shift register. Data is shifted out of the device on the negative edge of the TCK signal.
TMS Input **Test Mode Select**. The TMS pin selects the next state in the TAP state machine.
TCK Input **Test Clock**. This allows shifting of the data in, on the TMS and TDI pins. It is a positive edge triggered clock with the TMS and TCK signals that define the internal state of the device.

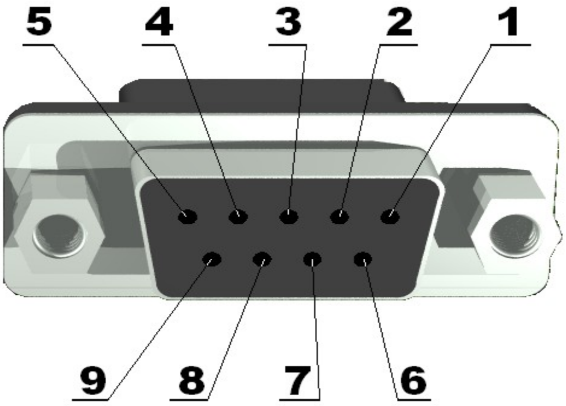
ISCP:

Pin #	Signal Name
1	RST
2	3.3V
3	GND
4	PGD2
5	PGC2
6	-

This connector allows programming and debugging via PIC-IDC2, PIC-IDC2-POCKET or compatible tool.

PWR-CON:

Pin #	Signal Name
1	+6VDC/4.5VAC
2	GND



This connector is used to power the PIC-32MX. External (4.5VAC/6VDC) power source have to be applied to these pins.

UEXT:

Pin #	Signal Name
1	3.3V
2	GND
3	TXD1
4	RXD1
5	SCL1
6	SDA1
7	MISO2
8	MOSI2
9	SCK2
10	#SS2

UEXT is connector for external plug-in modules.

TXD1 Output **Transmit Data 1**. This is the asynchronous serial data output (RS232) for the shift register.
RXD1 Input **Receive Data 1**. This is the asynchronous serial data input (RS232) for the shift register.
SCL1 I/O **Serial Clock 1**. This is the synchronization clock for the I2C 1 interface. It is output from the master and input for the slave.
SDA1 I/O **Serial Data 1**. Data register for the I2C interface.
MISO2 I/O **Master In Slave Out 2**. When processor is master this is input SPI 2 data register. When processor is slave this is output SPI data register.
MOSI2 I/O **Maser Out Slave In 2**. When pocessor is master this is output SPI 2 data register. When processor is slave this is input SPI data register.
SCK2 I/O **Serial Clock 2**. This is the synchronization clock for the SPI 2 interface. It is ouput from the master and input for the slave.
#SS2 I/O **Slave Select 2**. Save select signal dor the SPI 2. It is output from the master and input for the slave.

RS232:

Pin #	Signal Name
1	-
2	TXD2
3	RXD2
4	-
5	GND
6	-
7	U2CTS
8	U2RTS
9	-

TXD2 Output **Transmit Data 2**. This is the asynchronous serial data output (RS232) for the shift register on the UART2 controller.
RXD2 Input **Receive Data 2**. This is the asynchronous serial data input (RS232) for the shift register on the UART2 controller.
U2CTS Input **UART2 Clear To Send**. The DCE device is ready to accept data.
U2RTS Otput **UART2 Request To Send**. The DTE device (PIC-32MX) requests to send data.

INPUT/OUTPUT:

Button BUT - user button connected to PIC-32MX PORTD.RD8 (INT1).

RESET button – button connected to the RST pin of PIC-32MX

Power on LED (red) - its name is **PWR_LED** and indicates that power is on.

Status LED (green) – user LED connected to PIC-32MX PORTF.RF1 pin.

GETTING STARTED

In order to get started you need:

1. PIC-32MX board
2. Power supply (6VDC/4.5VAC)
3. Programmer
4. Cable to connect the programmer to the PC
5. Cable to connect the programmer to the board
6. Compilator/Assembler

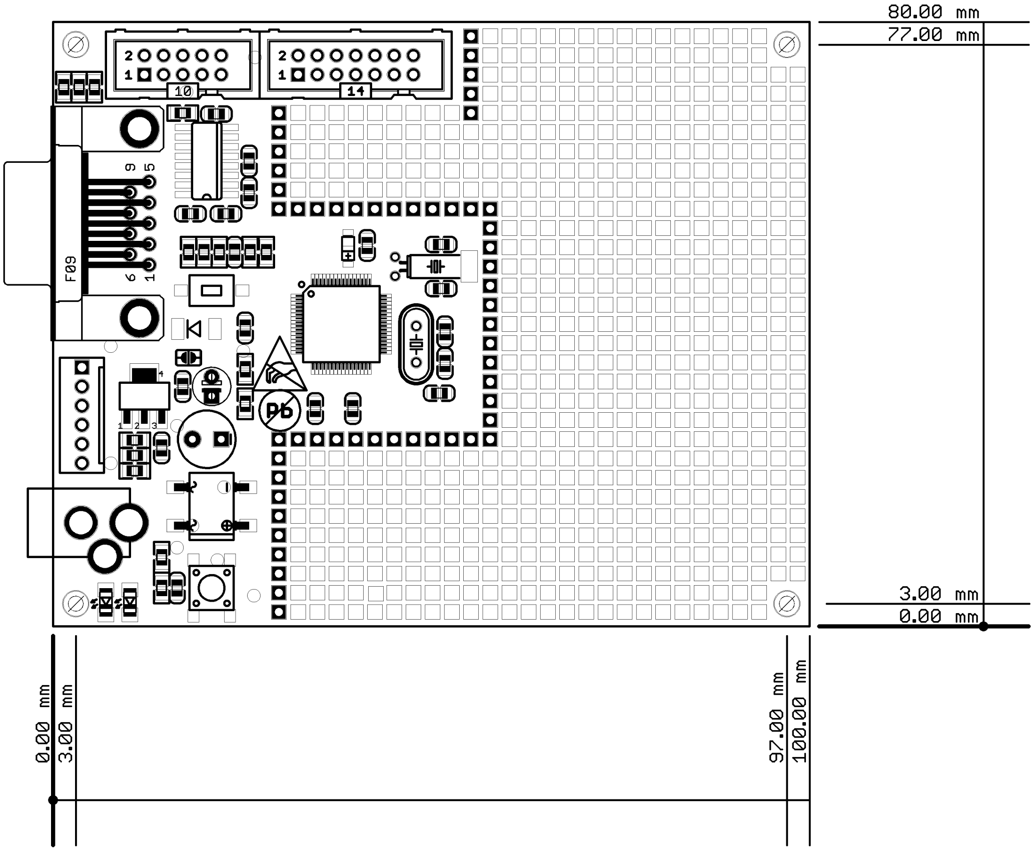
We provide here an example configuration but you could use any compatible programmer/compilator/assembler and the corresponding cables.

Example configuration:

1. PIC-32MX board
2. Power supply (6VDC/4.5VAC)
3. Programer – PIC-ICD2
4. Cable to connect the programmer to the PC – USB cable A-B
5. Cable to connect the programmer to the board – ICD2 cable (ISCP)
6. Compilator/Assembler – MPLAB v8.14 IDE + MPLAB C32

To get started you first should apply power to the board, connect the programmer to the board and to the PC, open MPLAB and write your application. After that you should compile and build your project, then you should debug it and if the previous stages are successfully finished, to program PIC-32MX. To get your application running you should disconnect the programmer from the board and reset the board.

MECHANICAL DIMENSIONS:



All measures are in mm.

AVAILABLE DEMO SOFTWARE:

All of the demo software is written under MPLAB v8.14 IDE + MPLAB C32.

1. **Blink LED demo software (C source and HEX)**
2. **Button read demo software (C source and HEX)**
3. **Timer1 demo software (C source and HEX)**
4. **UART basic demo software (C source and HEX)**
5. **UART interrupt demo software (C source and HEX)**

All of the demo software could be found on Olimex website www.olimex.com/dev.

ORDER CODE:

How to order?
You can order to us directly or by any of our distributors.
Check our web www.olimex.com/dev for more info.

Pb-free, Green All boards produced by Olimex are ROHS compliant

Revision history:

REV.A - created June 2008

Disclaimer: